Lecture 4

AI503: Advanced Machine Learning

Summary – last week

- Last week:
  - Academic Writing
    - Assessment 01 out

- This week:
  - Data Mining basics

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Data Mining (DM)

- What is data mining (knowledge discovery in databases)?
  - Extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) information or patterns from data in large databases
- What is not data mining?
  - (Deductive) query processing
  - Large or small statistical programs

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Principles of DM

- Data Mining applications
  - Database analysis and decision support
    - Market analysis and management
    - Risk analysis and management e.g., forecasting, customer retention, improved insurance policies, quality control, competitive analysis
    - Fraud detection and management
  - Other Applications
    - Text mining (news group, email, documents) and Web analysis (Google Analytics)
    - Intelligent query answering

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Principles of DM

- Market analysis
  - Targeted marketing/ Customer profiling
    - Find clusters of "model" customers who share the same characteristics: interest, income level, spending habits, etc.
  - Determine customer purchasing patterns over time
  - Cross-market analysis
    - Associations/co-relations between product sales
    - Prediction based on the association of information
  - Provide summary information
    - Various multidimensional summary reports
    - Statistical summary information (data central tendency and variation)

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

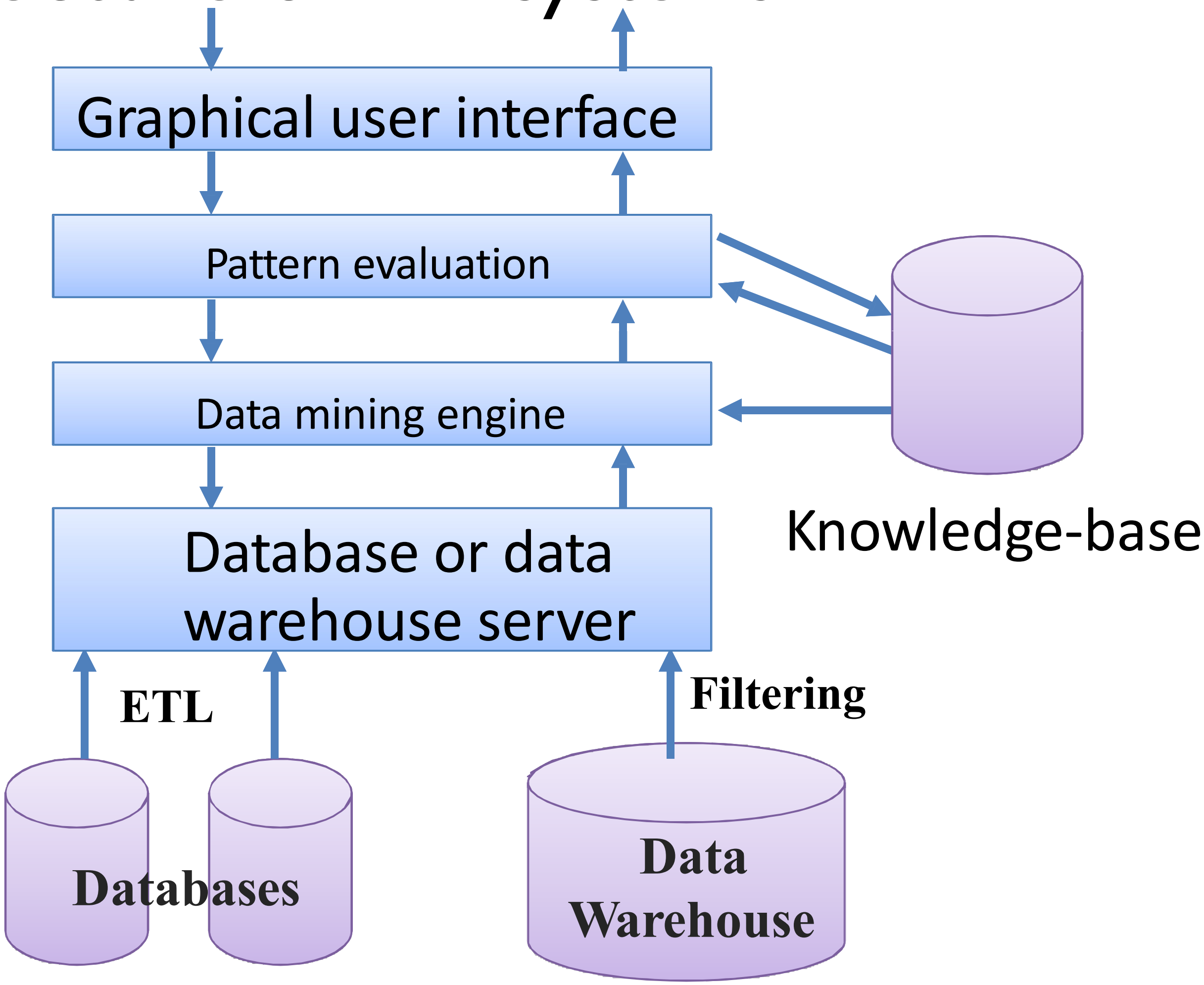
Principles of DM

- Corporate analysis and risk management
  - Finance planning and asset evaluation
    - Cash flow analysis and prediction
    - Trend analysis, time series, etc.
  - Resource planning
    - Summarize and compare the resource
      and spending
  - Competition
    - **Monitor** competitors and market directions
    - **Group** customers into classes and a class-based pricing procedure
    - Set pricing strategy in a highly competitive market

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Data Mining

- Architecture of DM systems



Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Data Mining

- DM functionalities
  - Association (correlation and causality)
    - Multi-dimensional vs. single-dimensional association
    - age(X, "20..29") , income(X, "20..29K") ⟶ buys(X, "PC") [support = 2%, confidence = 60%]
    - contains(T, "computer") ⟶ contains(x, "software") [1%, 75%]
  - Classification and Prediction
    - Finding models (functions) that describe and distinguish classes or concepts for future predictions
    - Presentation: decision-tree, classification rule, neural network
    - Prediction: predict some unknown or missing numerical values

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Data Mining

- Cluster analysis
  - Class label is unknown: group data to form new classes, e.g., cluster houses to find distribution patterns
  - Clustering based on the principle: maximizing the intra-class similarity and minimizing the interclass similarity
- Outlier analysis
  - Outlier: a data object that does not comply with the general behavior of the data
  - Can be considered as noise or exception, but is quite useful in fraud detection, rare events analysis

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Association Rule Mining

- Association rule mining has the objective of finding all co-occurrence relationships (called associations), among data items
  - Classical application: market basket data analysis, which aims to discover how items are purchased by customers in a supermarket
    - E.g., Cheese $\longrightarrow$ Bread [support = 10%, confidence = 80%] meaning that 10% of the customers buy cheese and 80% of customers buying cheese also buy bread.

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Association Rule Mining

- Basic concepts of association rules
  - Let $I = \{i_1, i_2, \ldots, i_m\}$ be a set of items. Let $T = \{t_1, t_2, \ldots, t_n\}$ be a set of transactions where each transaction t_i is a set of items such that $t_i \subseteq I$.
  - An association rule is an implication of the form:
    $X \longrightarrow Y$, where $X \subset I$, $Y \subset I$ and $X \cap Y = \emptyset$
    Bread $\longrightarrow$ Butter but not Bread $\longrightarrow$ Bread

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Association Rule Mining

- Association rule mining market basket analysis example
  - I – set of all items sold in a store
    - E.g., i_1 = Beef, i_2 = Chicken, i_3 = Cheese, …
  - T – set of transactions
    - The content of a customers basket
    - E.g., t_1: Beef, Chicken, Milk; t_2: Beef, Cheese; t_3: Cheese, Bread; t_4: …
  - An association rule might be
    - Beef, Chicken $\longrightarrow$ Milk, where {Beef, Chicken} is X and {Milk} is Y

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Association Rule Mining

- Rules can be weak or strong
  - The strength of a rule is measured by its **support** and **confidence**
  - The support of a rule $X \longrightarrow Y$, is the percentage of transactions in T that contains X and Y
    - Can be seen as an estimate of the probability $Pr(\{X,Y\} \subseteq t_i)$
    - With n as number of transactions in T, the support of the rule $X \longrightarrow Y$ is:

$$support = |\{i \mid \{X, Y\} \subseteq t_i\}| / n$$

    - Support deals with Data while the Confidence deals with semantic/bond

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Association Rule Mining

– The confidence of a rule $X \longrightarrow Y$, is the percentage of transactions in T containing X, that contain X U Y

- Can be seen as estimate of the probability $Pr(Y \subseteq t_i \mid X \subseteq t_i)$

$$\text{confidence} = |\{i \mid \{X, Y\} \subseteq t_i\}| \,/\, |\{j \mid X \subseteq t_j\}|$$

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Association Rule Mining

- Lift(l)
  The lift of the rule X=>Y is the confidence of the rule divided by the expected confidence, assuming that the itemsets X and Y are independent of each other. The expected confidence is the confidence divided by the frequency of {Y}.
- Lift(X=>Y) = Conf(X=>Y) / Supp(Y)
  Lift value near 1 indicates X and Y almost often appear together as expected, greater than 1 means they appear together more than expected and less than 1 means they appear less than expected. Greater lift values indicate stronger association

Association Rule Mining

- How do we interpret support and confidence?
  - If support is too low, the rule may just occur due to chance
    - Acting on a rule with low support may not be profitable since it covers too few cases
  - If confidence is too low, we cannot reliably predict Y from X
- Objective of mining association rules is to discover all associated rules in T that have support and confidence greater than a minimum threshold (minsup, minconf)!

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Association Rule Mining

- Finding rules based on support and confidence thresholds
  - Let minsup = 30% and minconf = 80%
  - Chicken, Clothes ⟶ Milk is valid, [sup = 3/7 (42.84%), conf = 3/3 (100%)]
  - Clothes ⟶ Milk, Chicken is also valid, and there are more…

Transactions	
T1	Beef, Chicken, Milk
T2	Beef, Cheese
T3	Cheese, Boots
T4	Beef, Chicken, Cheese
T5	Beef, Chicken, Clothes, Cheese, Milk
T6	Clothes, Chicken, Milk
T7	Chicken, Milk, Clothes

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Association Rule Mining

- This is rather a simplistic view of shopping baskets
  - Some important information is not considered, e.g., the quantity of each item purchased, the price paid,…
- There are a large number of rule mining algorithms
  - They use different strategies and data structures
  - Their resulting sets of rules are all the same

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

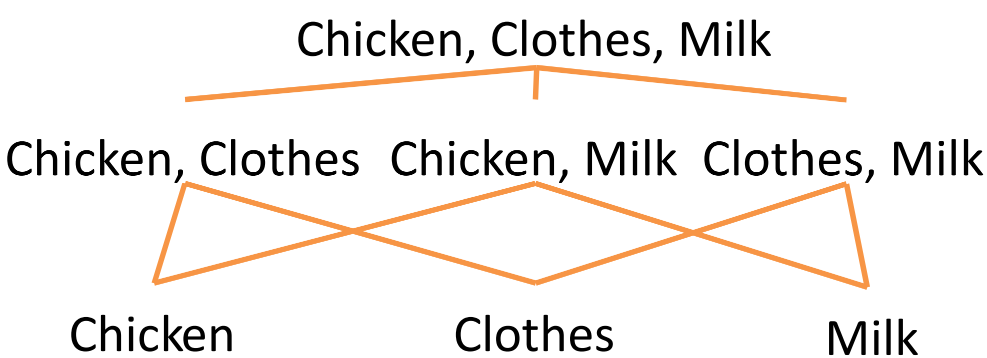
Association Rule Mining

- Approaches in association rule mining
  - Apriori algorithm
  - Mining with multiple minimum supports
  - Mining class association rules
- The best known mining algorithm is the Apriori algorithm
  - Step 1: find all frequent itemsets (set of items with support ≥ minsup)
  - Step 2: use frequent itemsets to generate rules

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm : Step 1

- Step 1: frequent itemset generation
  - The key is the apriori property (downward closure property): any subset of a frequent itemset is also a frequent itemset
    - E.g., for minsup = 30%



Transactions	
T1	Beef, Chicken, Milk
T2	Beef, Cheese
T3	Cheese, Boots
T4	Beef, Chicken, Cheese
T5	Beef, Chicken, Clothes, Cheese, Milk
T6	Clothes, Chicken, Milk
T7	Chicken, Milk, Clothes

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm : Step 1

- Finding frequent items
  - Find all 1-item frequent itemsets; then all 2-item frequent itemsets, etc.
  - In each iteration k, only consider itemsets that contain a k-1 frequent itemset
  - Optimization: the algorithm assumes that items are sorted in lexicographic order
    - The order is used throughout the algorithm in each itemset
    - {w[1], w[2], …, w[k]} represents a k-itemset w consisting of items w[1], w[2], …, w[k], where w[1] < w[2] < … < w[k] according to the lexicographic order

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Finding frequent items

- Initial step
  - Find frequent itemsets of size 1: F_1
- Generalization, $k \geq 2$
  - C_k = candidates of size k: those itemsets of size k that could be frequent, given F_{k-1}
  - F_k = those itemsets that are actually frequent, $F_k \subseteq C_k$ (need to scan the database once)

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm : Step 1

– Generalization of candidates uses F_{k-1} as input and returns a superset (candidates) of the set of all frequent k-itemsets. It has two steps:

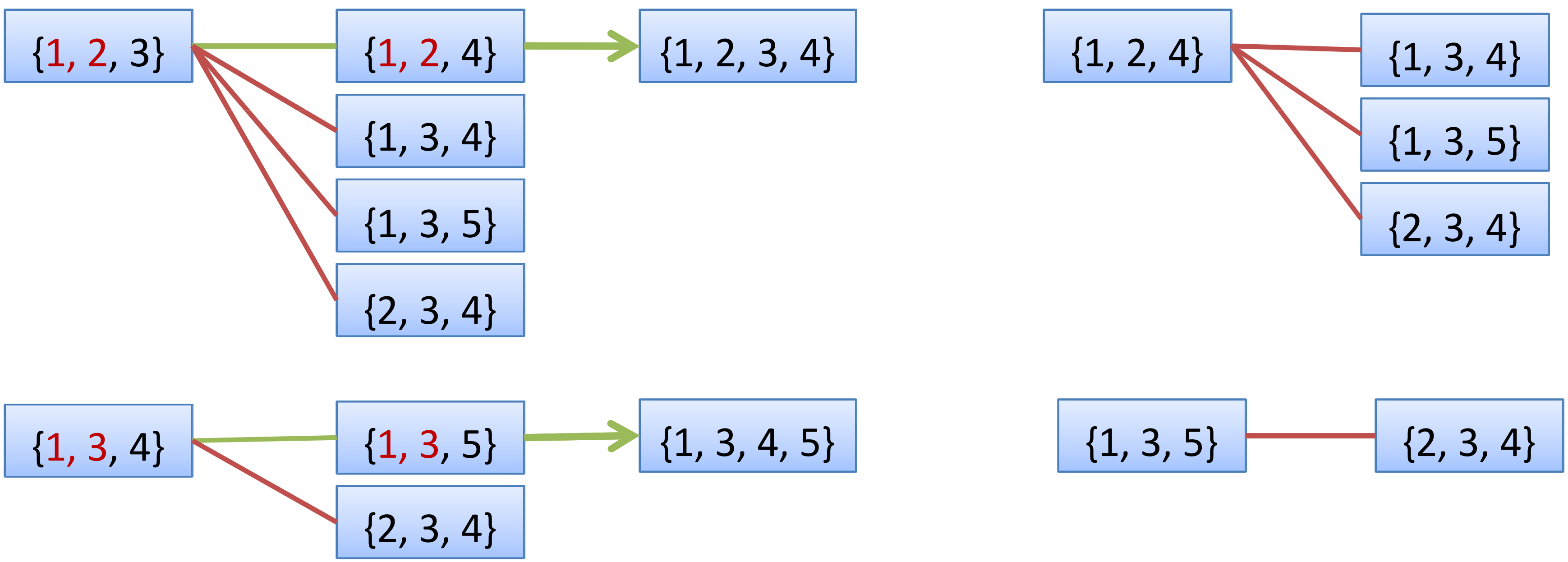
- Join step: generate all possible candidate itemsets C_k of length k, e.g., $I_k = join(A_{k-1}, B_{k-1}) \Leftrightarrow A_{k-1} = \{i_1, i_2, \ldots, i_{k-2}, i_{k-1}\}$ and $B_{k-1} = \{i_1, i_2, \ldots, i_{k-2}, i'_{k-1}\}$ and $i_{k-1} < i'_{k-1}$; Then $I_k = \{i_1, i_2, \ldots, i_{k-2}, i_{k-1}, i'_{k-1}\}$
- Prune step: remove those candidates in C_k that do not respect the downward closure property (include "k-1" non-frequent subsets)

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm : Step 1

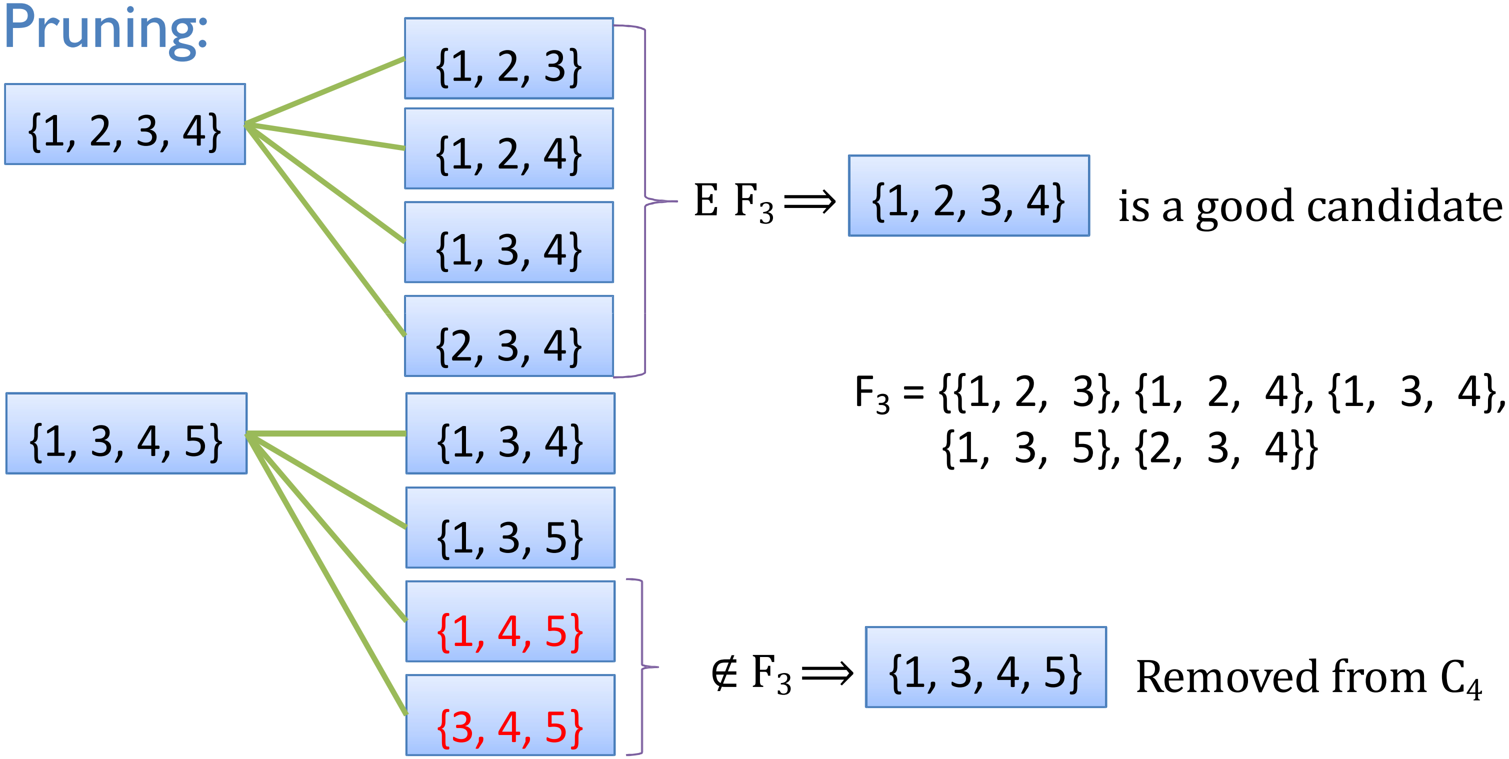
– Generalization e.g., F_3 = {{1,2,3}, {1,2,4}, {1,3,4}, {1,3,5}, {2,3,4}}

- Try joining each 2 candidates from F_3



Apriori Algorithm : Step 1

- After join C_4 = {{1, 2, 3, 4},{1, 3, 4, 5}}
- Pruning:



- After pruning C_4 = {{1, 2, 3, 4}}

Apriori Algorithm : Step 1

- Finding frequent items, example, minsup = 0.5
  - First T scan ({item}:count)
    - C_1: {1}:2, {2}:3, {3}:3, {4}:1, {5}:3
    - F_1: {1}:2, {2}:3, {3}:3, {5}:3;
      - {4} has a support of ¼ < 0.5 so it does not belong to the frequent items
    - C_2 = prune(join(F_l))
    - join : {1,2}, {1,3}, {1,5}, {2,3}, {2,5}, {3,5};
    - prune: C_2 : {1,2}, {1,3}, {1,5}, {2,3}, {2,5}, {3,5}; (all items
    - belong to F_1)

TID	Items
T100	1, 3, 4
T200	2, 3, 5
T300	1, 2, 3, 5
T400	2, 5

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm : Step 1

TID	Items
T100	1, 3, 4
T200	2, 3, 5
T300	1, 2, 3, 5
T400	2, 5

– SecondT scan

- C_2:{1,2}:1,{1,3}:2,{1,5}:1,{2,3}:2,{2,5}:3, {3,5}:2
- F_2:{1,3}:2,{2,3}:2,{2,5}:3,{3,5}:2
- Join: we could join {1,3} only with {1,4} or {1,5}, but they are not in F_2. The only possible join in F_2 is {2,3} with {2,5} resulting in {2,3,5};
- prune({2,3,5}):{2,3},{2,5},{3,5} all belong to F_2, hence, C_3:{2,3,5}

– ThirdT scan

- {2,3,5}:2, then sup({2,3,5}) = 50%, minsup condition is fulfilled. Then F_3:{2,3,5}

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm : Step 2

- Step 2: generating rules from frequent itemsets
  - Frequent itemsets are not the same as association rules
  - One more step is needed to generate association rules: for each frequent itemset I, for each proper nonempty subset X of I:
    - Let $Y = I \setminus X$; $X \longrightarrow Y$ is an association rule if:
      - Confidence($X \longrightarrow Y$) $\geq$ minconf,
      - Support($X \longrightarrow Y$) $:= |\{i \mid \{X, Y\} \subseteq t_i\}| / n$ = support(I)
      - Confidence($X \longrightarrow Y$) $:= |\{i \mid \{X, Y\} \subseteq t_i\}| / |\{j \mid X \subseteq t_j\}|$ = support(I) / support(X)

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm : Step 2

- Rule generation example, minconf = 50%
  - Suppose {2, 3, 5} is a frequent itemset, with sup=50%, as calculated in step 1
  - Proper nonempty subsets: {2, 3}, {2, 5}, {3, 5}, {2}, {3}, {5}, with sup=50%, 75%, 50%, 75%, 75%, 75% respectively
  - These generate the following association rules:
    - $2,3 \longrightarrow 5$, confidence= 100%; (sup(I)= 50%; sup{2,3}= 50%; 50/50= 1)
    - $2,5 \longrightarrow 3$, confidence= 67%; (50/75)
    - $3,5 \longrightarrow 2$, confidence= 100%; (…)
    - $2 \longrightarrow 3,5$, confidence= 67%
    - $3 \longrightarrow 2,5$, confidence= 67%
    - $5 \longrightarrow 2,3$, confidence= 67%
  - All rules have support = support(I) = 50%

TID	Items
T100	1, 3, 4
T200	2, 3, 5
T300	1, 2, 3, 5
T400	2, 5

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm : Step 2

- Rule generation, summary
  - In order to obtain $X \longrightarrow Y$, we need to know support(I) and support(X)
  - All the required information for confidence computation has already been recorded in itemset generation
    - No need to read the transactions data any more
    - This step is not as time-consuming as frequent itemsets generation

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Apriori Algorithm

- Apriori Algorithm, summary
  - If k is the size of the largest itemset, then it makes at most k passes over data (in practice, k is bounded e.g., 10)
  - The mining exploits sparseness of data, and high minsup and minconf thresholds
  - High minsup threshold makes it impossible to find rules involving **rare items** in the data.
- The solution is a mining with **multiple minimum supports** approach

Acknowledgment - Thanks to Wolf-Tilo Balke and Silviu Homoceanu - TU Braunschweig for the slides

Summary

- Some common uses of database systems.
- Characteristics of file-based systems.
- Problems with file-based approach.
- Meaning of the terms database, DBMS.
- Typical functions of a DBMS.
- Major components of the DBMS environment.
- Personnel involved in the DBMS environment.
- Advantages and disadvantages of DBMSs.

Next week

- Multiple Minimum Supports