C2: Mining Association Rules:
Apriori and Its Related Issues

Ming-Syan Chen

September 25, 2018

Agenda for Classes 2018
(Might be revised as we progress)

- Class 1 – (9/11) Overview of data mining
- Class 2 – (9/18) R (I), Wush Wu
- Class 3 – (9/25) Association, Apriori and its related issues
- Class 4 – (10/2) Data stream mining, FP Tree, Vertical mining
- Class 5 – (10/9) Classification: decision tree, GPGPU
- Class 6 – (10/16) Description of Data, Project announcement
- Class 7 – (10/23) R (II), Wush Wu

Tentative Class Agenda (cont'd)

- Class 8 – (10/30) Data exploration, more on decision trees, rule-based classifiers
- Class 9 – (11/6) Scikit learn, LibSVM, Preparation for HW3 and HW4
- Class 10 – (11/13) KNN, Bays, Neural network, Concept of SVM
- Class 11 – (11/20) Abstract presentation, SVM, Clustering, K-means, PAM
- Class 12 – (11/27) More on clustering; Sequential pattern mining;
- Class 13 – (12/4) Web mining, PageRank, etc.

Tentative Class Agenda (cont'd)

- Class 14– (12/11) When Database and Data Mining Meet, Prof. Mingling Lo
- Class 15 – (12/18) Project presentation I
- Class 16 – (12/25) Project presentation II

  (Final Exam according to Univ. Schedule)

  (Project due 1/24/2019)
- Happy New Year!

Procedure of Data Mining

- Obtain and look over the data
- Decide your goal (usually a stretched and reachable one)
- Data cleaning/cleansing
- Choose data granularity, feature selection
- Apply mining methods
- Decide what to output and in what form
- Interpret your results (may have iterative refinements); convince your receiver/boss

Mining Capabilities

- Association
- Classification
- Clustering
- Sequential Pattern
- and more

Mining Association Rules

- Transaction data analysis: Mining association rules
  - Given: (1) a database of transactions
    (2) each tx has a list of items purchased
- Find all asso. rules: the presence of one set of items implies the presence of another set of items
  - people who purchased hammers also purchased nails

Two Parameters

- Confidence (how true)
  - the rule X&Y => Z has 90% conf. means 90% of customers who bought X and Y also bought Z
- Support (how useful is the rule)
  - useful rules should have some minimum tx support

Mining Association Rules in Transaction DBs

- Measurement of rule strength in a transaction DB.

$$A \rightarrow B \quad [support, confidence]$$

$$support = Prob(A \cup B) = \frac{\#\_of\_trans\_that\_contain\_both\ A\ and\ B}{total\_\#\_of\_trans}$$

$$confidence = Prob(B|A) = \frac{\#\_of\_trans\_that\_contain\_both\ A\ and\ B}{\#\_of\_trans\_containing\ A}$$

- We are often interested in only strong associations, i.e.

  $support \geq min\_sup$ and $confidence \geq min\_conf$.

- Examples.

  milk $\rightarrow$ bread [5%, 60%].

  tire $\wedge$ auto_accessories $\rightarrow$ auto_services [2%, 80%].

Two Steps for Mining Asso.

- Determining "large itemsets"
  - the main factor for overall performance
- Generating rules

Two approaches for Large Itemset Counting

- Apriori-Based
  - R. Agrawal and R. Srikant
- FP-Tree-Based
  - J. Han and J. Pei, etc (SIGMOD 2000)

Methods for Mining Association Rules

- Apriori (Agrawal & Srikant'94).
- Itemset generation
  - derivation of large 1-itemsets L1: At the first iteration, scan all the transactions and count the number of occurrences for each item.
  - level-wise derivation: At the *k*-th iteration, the candidate set *Ck* are those whose every (*k* - 1)-item subset is in *Lk-1*. Scan DB and count the # of occurrences for each candidate itemset.
  - the cardinality of C2 is huge
  - the exe time for the first 2 iterations is the dominating factor to overall performance

Support=2 tx's (i.e., 50%)

Database D

TID	Items
100	A C D
200	B C E
300	A B C E
400	B E

Scan D →

C_1

Itemset	Sup.
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

L_1

Itemset	Sup.
{A}	2
{B}	3
{C}	3
{E}	3

C_2

Itemset
{A B}
{A C}
{A E}
{B C}
{B E}
{C E}

Scan D →

C_2

Itemset	Sup.
{A B}	1
{A C}	2
{A E}	1
{B C}	2
{B E}	3
{C E}	2

L_2

Itemset	Sup.
{A C}	2
{B C}	2
{B E}	3
{C E}	2

C_3

Itemset
{B C E}

Scan D →

C_3

Itemset	Sup.
{B C E}	2

L_3

Itemset	Sup.
{B C E}	2

BE=>C conf:66%

Two Steps for Mining Asso. (cont'd)

- for each large itemset m do
  for each subset p of m do
  if (sup(m)/sup(m-p)>= minconf) then
  output the rule (m-p)=>p
  with conf= sup(m)/sup(m-p) and
  support=sup(m)
- m={a,c,d,e,f,g} 2000 tx's
  p={a,d} 5000 tx's
  {a,d}=>{c,e,f,g} conference: 40%, support: 2000 tx's

Properties of Apriori

- Downward closure for large (also called frequent) itemset generation
- The bottleneck is usually in C2
- Database scan is expensive
- The setting of "support" and "confidence"
- Using "top-k" itemsets instead of support
  - How to do itemset generation

Follow-ups of Apriori

- Data Stream mining
  - W.-G. Teng, M.-S. Chen and P. S. Yu, ``A Regression-Based Temporal Pattern Mining Scheme for Data Streams," *Proc. of the 29th Intern'l Conf. on Very Large Data Bases (VLDB-2003)*, September 9-12, 2003.
- Upper bound on the number of large itemsets
  - F. Geerts and B. Goethals and J. V. D. Bussche, "Tight upper bounds on the number of candidate patterns", TODS 2005)
- "closed large itemset"
- Spawned many works to improve its efficiency and also to explore its variations

Closed Itemsets and Maximal Itemsets

- An itemset X is called closed if there does not exist an itemset Y, s.t. Y contains X and s(Y)=s(X)
- A large itemset X is called maximal if there does not exist a large itemset Y, s.t., Y contains X

- Q1: if a large itemset X is closed, is X always maximal?
- Q2: if a large itemset X is maximal large itemset, is X always closed?

Closed Itemsets and Maximal Itemsets (cont'd)

- DB1 is said to be "equivalent" to DB2 if DB1 can be obtained from tx permutation from DB1
- Q: Prove or disapprove with an example.
  Q1. If DB1 and DB2 are equivalent, do they lead to the same set of colsed itemsets?
  Q2. If DB1 and DB2 lead to the same set of colsed itemsets, are DB1 and DB2 always equivalent to each other?

Redundant Rules

- For the same support and confidence, if we have a rule {a,d}=>{c,e,f,g}, do we have
  - {a,d}=>{c,e,f}
  - {a}=>{c,e,f,g}
  - {a,d,c}=>{e,f,g}
  - {a}=>{d,c,e,f,g}

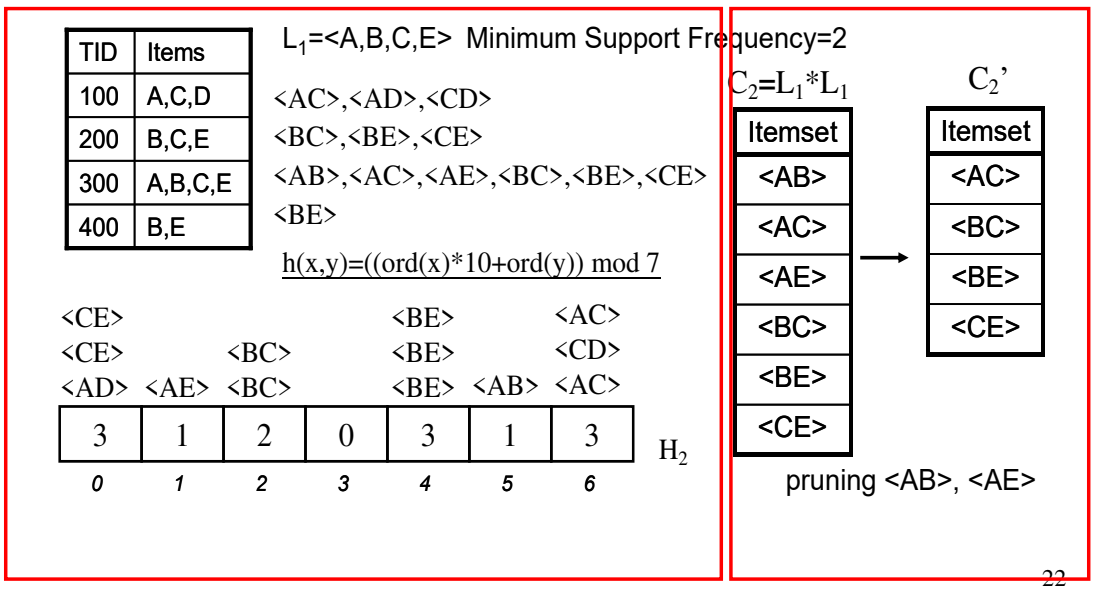
Scan Reduction

- Use candidate sets to generate candidate sets
  e.g., Instead of Ci→ Li→Ci+1 (dbscan)→Li+1
  We use Ci→Ci+1'→Ci+2' (dbscan)→Li+1,Li+2..
- Save the runs of database scans
- May get back to use large itemsets to generate candidate sets if so necessary

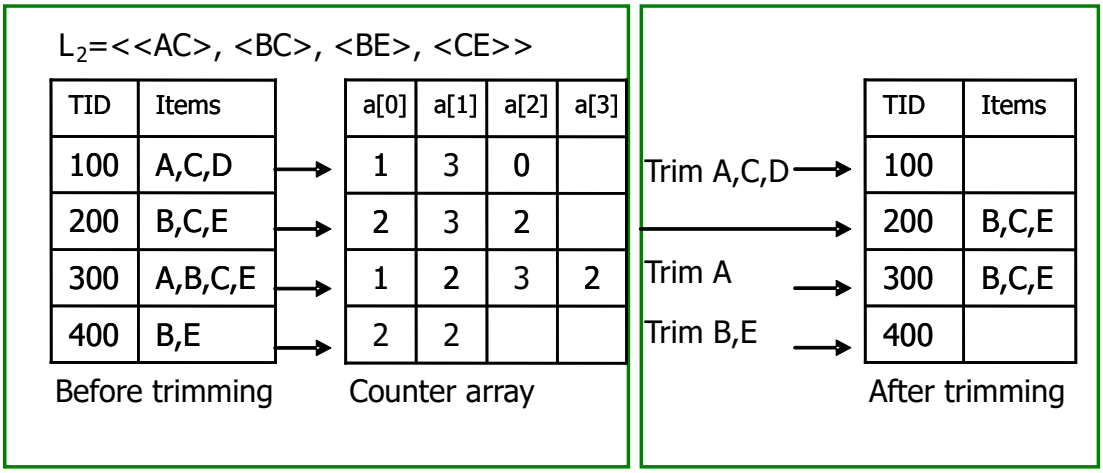
Improvement for Aprior

- DHP
  - J. Park, M.-S. Chen, and P. Yu. "*An effective hash based algorithm for mining association rules*. " *Proceedings of ACM SIGMOD*, May 1995. A complete version in Using A Hash-Based Method with Transaction Trimming for Mining Association Rules," IEEE Trans. on Knowledge and Data Eng., vol. 9, no. 5, pp. 813-825, Sept./Oct. 1997.
- Hash table scheme
  - Eliminate infrequent candidate itemsets in the early phase
- Transaction items pruning
  - Eliminate infrequent items from the database

Candidate Itemsets Pruning



Transaction Items Pruning (from L2 to L3)



Trimming information collecting (A appears in L2 once, C in L2 3 times, D not in L2)

Transaction trimming

A misleading "strong" association rule

- 10000 transactions
  - 6000 of them included computer games.
  - 7500 of them included video.
  - 4000 of them included computer games and video.
- Minimum support: 30%, minimum confidence: 60%

$$buys(\text{computer games}) \Rightarrow buys(\text{videos})$$
$$[support = 40\%, confidence = 66\%]$$

- However, P({video})=0.75

From Association Analysis to Correlation Analysis

- The support and confidence measures are insufficient at filtering out uninteresting association rules.

$$A \Rightarrow B[support, confidence, correlation]$$

Lift

- The **lift** between the occurrence of A and B can be measured by computing

  The probability of a transaction contains the *union* of sets A and B.
  ↖ It doesn't mean P(A or B).

$$lift(A,B) = \frac{P(A \cup B)}{P(A)P(B)} = \frac{P(B|A)}{P(B)} = \frac{conf(A \Rightarrow B)}{\sup(B)}$$

  < 1, negatively correlated
  > 1, positively correlated
  = 1, no correlation (A and B are independent)

- **Lift** assesses the degree to which the occurrence of one "lifts" the occurrence of the other.

Interestingness Measure: Correlations (Lift)

- *play basketball* $\Rightarrow$ *eat cereal* [40%, 66.7%] is misleading
  - The overall % of students eating cereal is 75% > 66.7%.
- *play basketball* $\Rightarrow$ *not eat cereal* [20%, 33.3%] is more accurate, although with lower support and confidence
- Measure of dependent/correlated events: lift

$$lift = \frac{P(A \cup B)}{P(A)P(B)}$$

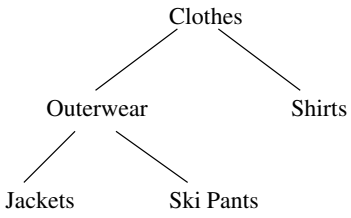
$$lift(B,C) = \frac{2000/5000}{3000/5000 * 3750/5000} = 0.89$$

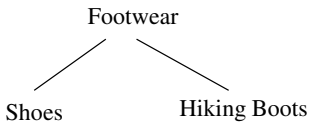
$$lift(B,\neg C) = \frac{1000/5000}{3000/5000 * 1250/5000} = 1.33$$

	Basketball	Not basketball	Sum (row)
Cereal	2000	1750	3750
Not cereal	1000	250	1250
Sum(col.)	3000	2000	5000

Generalized Association Rules

- Given the class hierarchy (taxonomy), one would like to choose proper data granularities for mining.
- Different confidence/support may be considered.





Freg. Itemset	Itemset support
Jacket	2
Outerwear	3
Clothes	4
Shoes	2
Hiking Boots	2
Footwear	4
Outerwear, Hiking Boots	2
Clothes, Hiking Boots	2
Outerwear, Footwear	2
Clothes, Footwear	2

Database

Tx	Items bought
100	Shirt
200	Jacket, Hiking Boots
300	Ski Pants, Hiking Boots
400	Shoes
500	Shoes
600	Jacket

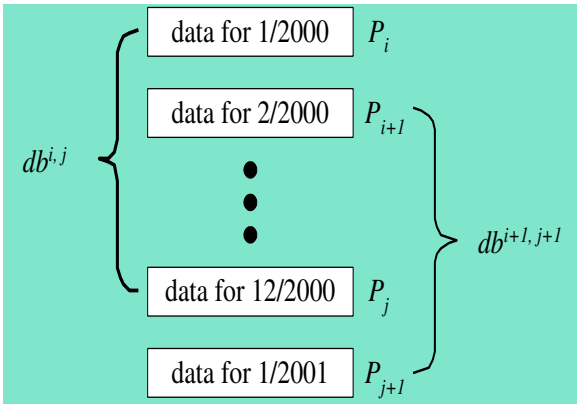
	sup(30%)	conf(60%)
Outerwear → Hiking Boots	33%	66%
Outerwear → Footwear	33%	66%
Hiking Boots → Outwear	33%	100%
Hiking Boots → Clothes	33%	100%
However,		
Jacket → Hiking Boots	16%	50%
Ski Pants → Hiking Boots	16%	100%

Incremental Mining

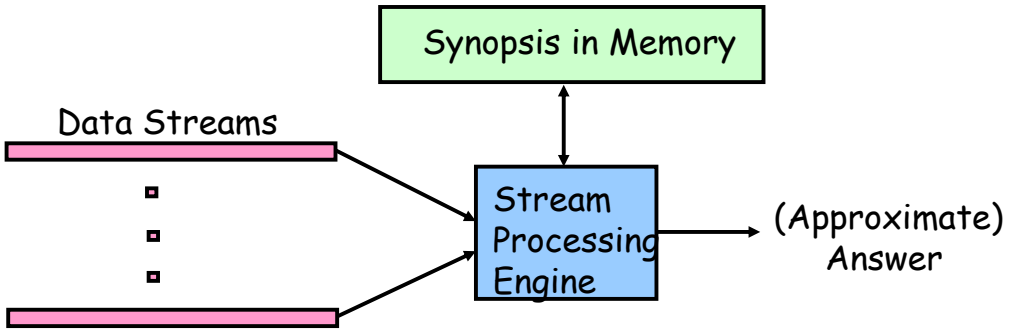
- Due to the increasing use of the record-based databases, recent important applications have called for the need of incremental mining
  - Such applications include Web log records, stock market data, grocery sales data, transactions in electronic commerce, and daily weather/traffic records, to name a few

Incremental Mining

- To mine the transaction database for a fixed amount of most recent data (say, data in the last 12 months)
- One has to not only include new data (i.e., data in the new month) into, but also remove the old data (i.e., data in the most obsolete month) from the mining process.
- Google for "Incremental mining"



Data Streams: Computation Model



- Stream processing requirements
  - **Single pass:** Each record is examined at most once
  - **Bounded storage**: Limited Memory for storing synopsis
  - **Real-time**: Per record processing time must be low

Big Data (Recalled)

- Big data refers to a huge amount of data which is fast accumulated from various sources.
  - ***Volume*** - Scale from terabytes to zettabytes
  - ***Variety*** - Relational and non-relational data types from an ever-expanding variety of sources (e.g., IOT, Social Networks, Multimedia applications)
  - ***Velocity*** - Streaming data and fast movement of large volume data
- Veracity
- Value

Related Papers (just some)

- The easiest way is to use Google
- R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," Proc. 20th Int'l Conf. Very Large Data Bases, pp. 478-499, Sept. 1994.
- M.-S. Chen, J. Han, and P. S. Yu. Data mining: An overview from a database perspective. IEEE Trans. Knowledge and Data Engineering, 8:866-883, 1996.
- J. Park, M.-S. Chen, and P. S. Yu. Using A Hash-Based Method with Transaction Trimming for Mining Association Rules," IEEE Trans. on Knowledge and Data Eng., vol. 9, no. 5, pp. 813-825, Sept./Oct. 1997.

About reading papers

- Not the same as reading textbook (important!)
- First, know who, why, where and when the paper was written
- What the problem is solved and how (in a high level)
- What is the primary contribution (method proposed, new finding, etc)
- How this paper was cited later?
- Anything that I should/could do with a stretched effort