15-826: Multimedia Databases and Data Mining

Lecture #30: Data Mining - assoc. rules

C. Faloutsos

Problem

• Given many market baskets
• Find which products sell together
Answer:

- Given many market baskets
- Find which products sell together
- Association rules ['large itemsets']
  - \{Milk, bread\} -> butter [milk,bread,butter: often]

Must-read Material

- Rakesh Agrawal, Tomasz Imielinski and Arun Swami Mining Association Rules Between Sets of Items in Large Databases Proc. ACM SIGMOD, Washington, DC, May 1993, pp. 207-216
Outline

Goal: ‘Find similar / interesting things’
- Intro to DB
- Indexing - similarity search
- Data Mining
  - …
  - Association Rules

Association rules - outline

- Main idea [Agrawal+SIGMOD93]
- performance improvements
- Variations / Applications
- Follow-up concepts
Association rules - idea

[Agrawal+SIGMOD93]
• Consider ‘market basket’ case:
  (milk, bread)
  (milk)
  (milk, chocolate)
  (milk, bread)
• Find ‘interesting things’, eg., rules of the form:
  milk, bread -> chocolate | 90%

Other settings?

• Market baskets - products
Other settings?

- Market baskets - products
- Documents - terms
- Patients - symptoms
- Proteins - proteins
- ...
- <any bi- or uni-partite graph>

Association rules - idea

[Agrawal+SIGMOD93]

- Consider ‘market basket’ case:
  - (milk, bread)
  - (milk)
  - (milk, chocolate)
  - (milk, bread)
- Find ‘interesting things’, eg., rules of the form:
  - milk, bread -> chocolate | 90%
Association rules - idea

In general, for a given rule

\[ I_j, I_k, \ldots, I_m \rightarrow I_x \mid c \]

‘c’ = ‘confidence’ (how often people by I_x, given that they have bought I_j, \ldots, I_m)

‘s’ = support: how often people buy I_j, \ldots, I_m, I_x

Problem definition:

• given
  – a set of ‘market baskets’ (=binary matrix, of N rows/baskets and M columns/products)
  – min-support ‘s’ and
  – min-confidence ‘c’

• find
  – all the rules with higher support and confidence
Association rules - idea

Closely related concept: “large itemset”

Ij, Ik, ... Im, Ix

is a ‘large itemset’, if it appears more than ‘min-support’ times

Observation: once we have a ‘large itemset’, we can find out the qualifying rules easily (how?)

Thus, let’s focus on how to find ‘large itemsets’

Association rules - idea

Naive solution: scan database once; keep $2^{*|I|}$ counters

Drawback?

Improvement?
Association rules - idea

Naive solution: scan database once; keep $2^{\vert I\vert}$ counters
Drawback? $2^{1000}$ is prohibitive...
Improvement? scan the db $\vert I\vert$ times, looking for 1-, 2-, etc itemsets

Eg., for $\vert I\vert=3$ items only (A, B, C), we have

\begin{tabular}{c|c|c}
A & B & C \\
100 & 200 & 2 \\
\end{tabular}

first pass

min-sup: 10
Anti-monotonicity property:
if an itemset fails to be ‘large’, so will every superset of it (hence all supersets can be pruned)

Sketch of the (famous!) ‘a-priori’ algorithm
Let \( L(i-1) \) be the set of large itemsets with \( i-1 \) elements
Let \( C(i) \) be the set of candidate itemsets (of size \( i \) )
Association rules - idea

Compute L(1), by scanning the database.
repeat, for i=2,3..., ‘join’ L(i-1) with itself, to generate C(i)
   two itemset can be joined, if they agree on their first i-2 elements
   prune the itemsets of C(i) (how?)
   scan the db, finding the counts of the C(i) itemsets - set
   this to be L(i)
   unless L(i) is empty, repeat the loop
(see example 6.1 in [Han+Kamber])

Association rules - outline

• Main idea [Agrawal+SIGMOD93]
• performance improvements
• Variations / Applications
• Follow-up concepts
Association rules - improvements

- Use the independence assumption, to second-guess large itemsets a few steps ahead
- Eliminate ‘market baskets’, that don’t contain any more large itemsets
- Partitioning (eg., for parallelism): find ‘local large itemsets’, and merge.
- Sampling
- Report only ‘maximal large itemsets’ (dfn?)
- FP-tree/FP-growth (seems to be the fastest)

Association rules - improvements

- FP-tree: no candidate itemset generation - only two passes over dataset
- Main idea: build a TRIE in main memory
  Specifically:
  - first pass, to find counts of each item - sort items in decreasing count order
  - second pass: build the TRIE, and update its counts

(eg., let A, B, C, D be the items in frequency order:)

Association rules - improvements

• eg., let A, B, C, D be the items in frequency order:

\[
\begin{align*}
&32 \text{ records} \\
&10 \text{ of them have A} \\
&4 \text{ have AB} \\
&2 \text{ have AC} \\
&1 \text{ has C}
\end{align*}
\]

Jiawei Han, Jian Pei, and Yiwen Yin. 2000. *Mining frequent patterns without candidate generation*. SIGMOD Rec. 29, 2 (May 2000), 1-12.
Association rules - outline

• Main idea [Agrawal+SIGMOD93]
• Performance improvements
• Variations / Applications
• Follow-up concepts

Association rules - variations

1) Multi-level rules: given concept hierarchy
• ‘bread’, ‘milk’, ‘butter’ -> foods;
• ‘aspirin’, ‘tylenol’ -> pharmacy
look for rules across any level of the hierarchy, eg
‘aspirin’ -> foods
(similarly, rules across dimensions, like ‘product’,
‘time’, ‘branch’:
‘bread’, ‘12noon’, ‘PGH-branch’ -> ‘milk’
Association rules - variations

2) Sequential patterns:
   ‘car’, ‘now’ -> ‘tires’, ‘2 months later’
   Also: given a stream of (time-stamped) events:
   A A B A C A B A C ......  
   find rules like
   B, A -> C
   [Mannila+KDD97]

Association rules - variations

3) Spatial rules, eg:
   ‘house close to lake’ -> ‘expensive’
Association rules - variations

4) Quantitative rules, eg:
   ‘age between 20 and 30’, ‘chol. level <150’ ->
   ‘weight > 150lb’

I.e., given numerical attributes, how to find rules?

Solution:
- bucketize the (numerical) attributes
- find (binary) rules
- stitch appropriate buckets together:
Association rules - outline

- Main idea [Agrawal+SIGMOD93]
- Performance improvements
- Variations / Applications
- Follow-up concepts

Association rules - follow-up concepts

Associations rules vs. correlation.
Motivation: if

milk, bread

is a ‘large itemset’, does this mean that there is a positive correlation between ‘milk’ and ‘bread’ sales?
Association rules - follow-up concepts

Motivation: if milk, bread is a ‘large itemset’, does this means that there is a positive correlation between ‘milk’ and ‘bread’ sales?

NO!!

‘milk’ and ‘bread’ ANTI-correlated, yet milk+bread: frequent

What to do, then?
Association rules - follow-up concepts

What to do, then?
A: report only pairs of items that are indeed correlated - ie, they pass the Chi-square test
The idea can be extended to 3-, 4- etc itemsets (but becomes more expensive to check)
See [Han+Kamber, § 6.5], or [Brin+,SIGMOD97]

Association rules - Conclusions

Association rules: a new tool to find patterns
• easy to understand its output
• fine-tuned algorithms exist (FP-growth)
Answer:

- Given many market baskets
- Find which products sell together
- Association rules ['large itemsets']
  - {Milk, bread} -> butter [milk,bread,butter: often]