Search and Data Mining: Techniques

Association Rules

Anya Yarygina

Boris Novikov

Introduction

- Basic concepts and road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining

Reference

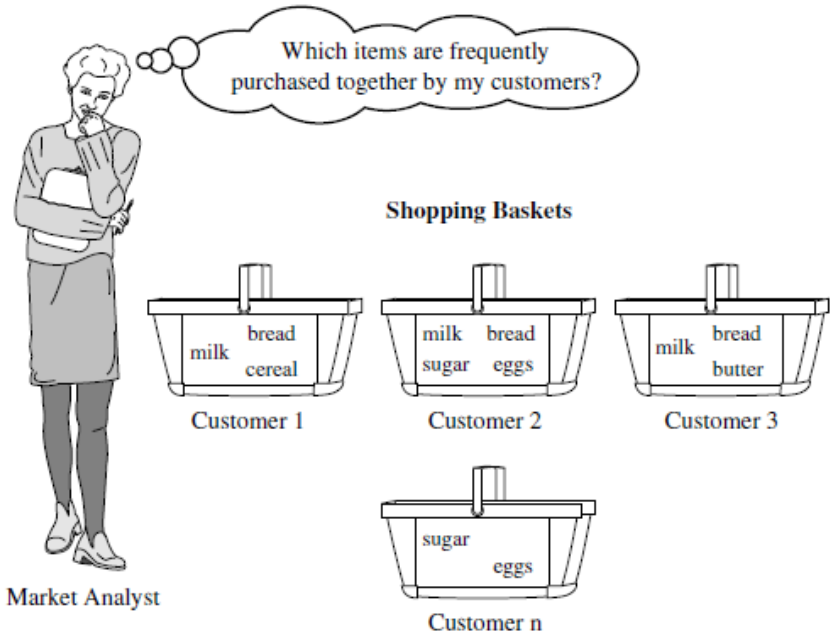
Jiawei Han, Micheline Kamber. Data Mining: Concepts and Techniques, 2nd Edition. The Morgan Kaufmann Series in Data Management Systems, 2006

Frequent patterns

- Frequent patterns
  - Frequent itemset
  - Frequent sequential pattern
  - Frequent structure pattern

Market basket analysis: a motivating example

- Analyze customer buying habits by finding associations between the different items that customers place in their "shopping baskets"



- Basket as boolean vector
- Buying patterns that reflect items that are frequently associated or purchased together
- Association rules
  - computer=>antivirus_sw [support=2%, confidence=60%]
- Measures of rule interestingness
  - Support (usefulness)
  - Confidence (certainty)
- Interestingness criteria
  - Minimum support threshold
  - Minimum confidence threshold

Frequent itemsets, closed itemsets, association rules

- I={I1,I2,…,Im}
- D={T}
- A=>B
  - Support(A=>B) = P(AB)
  - Confidence(A=>B) = P(B|A)

- Frequent itemset
- Strong rules
- Closed itemset
- Closed frequent itemset
- Maximal frequent itemset

Association rule mining

- Find all frequent itemsets
- Generate strong association rules from frequent itemsets

Frequent pattern mining: a road map

- Completeness of patterns to be mined
  - Complete set of frequent itemsets
  - Closed frequent itemsets
  - Maximal frequent itemsets
  - Constrained frequent itemsets
  - Approximate frequent itemsets
  - Near-match frequent itemsets
  - Top-k frequent itemsets
- Levels of abstraction involved in rule set
  - Single-level association rules
  - Multilevel association rules
- Number of data dimensions involved in the rule
  - Single-dimensional association rules
  - Multidimensional association rules
- Types of values handled in the rule
  - Boolean association rules
  - Quantitative association rules
- Kinds of rules to be mined
  - Association rules
  - Correlation rules
  - Strong gradient relationships
- Kinds of patterns to be mined
  - Frequent itemset mining
  - Sequential pattern mining
  - Structured pattern mining

Efficient and scalable frequent itemset mining methods

- Finding frequent itemsets
  - With candidate generation
  - Without candidate generation
  - From data in vertical format
  - Closed frequent itemsets
- Generating association rules

Apriori algorithm: finding frequent itemsets using candidate generation

- Level-wise search
- Apriori property
  - All nonempty subsets of a frequent itemset must also be frequent
- Construct L_k from L_{k-1}
  - Join step
    - Join L_{k-1} with itself to construct C_k
    - Members of L_{k-1} are joinable if their first (k-2) items are in common
  - Prune step
    - Prune infrequent itemsets
    - Use apriori property
      - Any subset of candidate should be frequent
      - Subset testing using hash tree of all frequent itemsets

Aproiri algorithm: pseudo-code

Input:

- D, a database of transactions;
- *min_sup*, the minimum support count threshold.

Output: L, frequent itemsets in D.

Method:

```
(1)   L1 = find_frequent_1-itemsets(D);
(2)   for (k = 2; Lk−1 ≠ ϕ; k++) {
(3)      Ck = apriori_gen(Lk−1);
(4)      for each transaction t ∈ D { // scan D for counts
(5)         Ct = subset(Ck, t); // get the subsets of t that are candidates
(6)         for each candidate c ∈ Ct
(7)            c.count++;
(8)      }
(9)      Lk = {c ∈ Ck | c.count ≥ min_sup}
(10)  }
(11)  return L = ∪k Lk;

procedure apriori_gen(Lk−1:frequent (k − 1)-itemsets)
(1)   for each itemset l1 ∈ Lk−1
(2)      for each itemset l2 ∈ Lk−1
(3)         if (l1[1] = l2[1]) ∧ (l1[2] = l2[2]) ∧ ... ∧ (l1[k − 2] = l2[k − 2]) ∧ (l1[k − 1] < l2[k − 1]) then {
(4)            c = l1 ⋈ l2; // join step: generate candidates
(5)            if has_infrequent_subset(c, Lk−1) then
(6)               delete c; // prune step: remove unfruitful candidate
(7)            else add c to Ck;
(8)         }
(9)   return Ck;

procedure has_infrequent_subset(c: candidate k-itemset;
          Lk−1: frequent (k − 1)-itemsets); // use prior knowledge
(1)   for each (k − 1)-subset s of c
(2)      if s ∉ Lk−1 then
(3)         return TRUE;
(4)   return FALSE;
```

Generating association rules from frequent itemsets

- Strong association rules
  - Minimum support
  - Minimum confidence
    - Confidence(A=>B)=P(B|A)
- Generation of association rules
  - For each frequent itemset l generate all nonempty subsets of l
  - For every nonempty subset s of l output the rule “s=>(l-s)” if confidence is “high”

Improving efficiency of Apriori

- Hash-based technique
  - Hashing items into corresponding buckets
- Transaction reduction
  - Remove transactions from further consideration when it does not contain any frequent k-items
- Partitioning
  - Subdivide the transactions into nonoverlapping partitions
  - Find (local) frequent itemsets within partition
  - Calculate actual support to select global frequent itemsets
- Sampling
  - Mining on subset of given data
- Dynamic itemset counting
  - Adding candidate itemsets at different points during a scan

Mining frequent itemsets without candidate generation

- Frequent-pattern growth
  - Divide-and-conquer strategy
    - Compress the database representing frequent items into a frequent-pattern tree
    - Divide the compressed database into a set of conditional databases, each associated with one frequent item or “pattern fragment”
    - Mine each conditional database separately
- Set of frequent items sorted in the order of descending support count
- FP-tree construction
  - Branch created for each transaction
  - Count of each node in the common prefix is incremented
- Mining FP-tree
  - Each frequent length-1 pattern (an initial suffix pattern)
  - Conditional pattern base (set of prefix paths)
  - Conditional FP-tree
  - Frequent patterns generation

Mining frequent itemsets using vertical data format

- Horizontal data format {TID-itemset}
- Vertical data format {item-TIDset}
- ECLAT algorithm
  - Intersecting the TIDsets of every pair of frequent items
  - Apriori property
  - Support calculation
  - Diffset technique (track only the differences of TIDsets)

Mining closed frequent itemsets

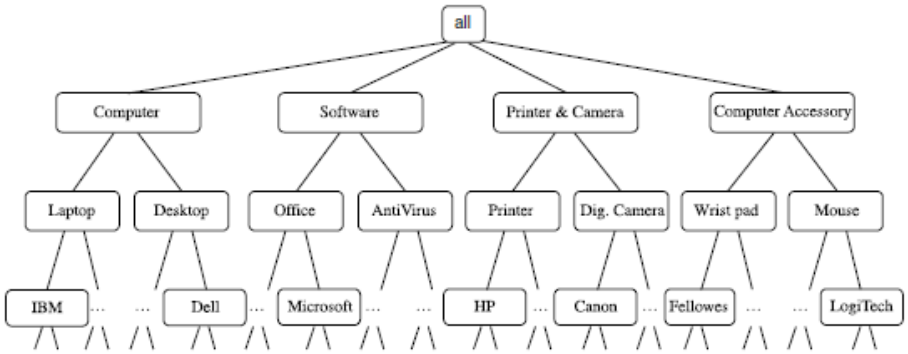
- Naïve approach
- Search for closed itemsets directly during the mining process (pruning techniques)
  - Item merging
  - Sub-itemset pruning
  - Item skipping
- Efficient checking of newly derived frequent itemset
  - Superset checking
  - Subset checking

Mining various kinds of association rules

- Multilevel association rules
- Multidimensional association rules

Mining multilevel association rules

- Transaction database
- Concept hierarchy



- Top-down strategy for support calculation
  - Using uniform minimum support for all levels
  - Using reduced minimum support at lower levels
  - Using item or group-based minimum support

Mining multidimensional association rules

- Database attributes
  - Categorical
  - Quantitative
    - Discretized using predefined concept hierarchies
    - Discretized or “clustered” into bins based on distribution of the data
- Search for frequent predicate sets

Mining multidimensional association rules using static discretization of quantitative attributes

- Quantitative attributes are discretized before mining using predefined concept hierarchies or data discretization techniques
- Categorical attributes may be generalized
- Relational database
  - Frequent itemset mining algorithm
- Data cube

Mining quantitative association rules

- Multidimensional association rules in which the numeric attributes are dynamically discretized during the mining process so as to satisfy some mining criteria (maximizing the confidence or compactness of the rules mined)
- Aquan1,Aquan2=>Acat
- Association rule clustering system
  - Map pairs of quantitative attributed into 2D grid for tuple satisfying a given categorical condition; search for clusters of point for which association rules are generated
  - Binning
    - Equal-width binning
    - Equal-frequency binning
    - Clustering-based binning
  - Finding frequent predicate sets
  - Clustering the association rules (rectangular clusters of rules)

From association mining to correlation analysis

- Strong rules are not necessarily interesting
- A=>B [support, confidence, correlation]
  - lift(A,B)=P(AB)/(P(A)P(B))
  - chi-squared $\chi^2 = \Sigma \frac{(observed - expected)^2}{expected}$
  - all_conf $all\_conf(X) = \frac{sup(X)}{max\_item\_sup(X)} = \frac{sup(X)}{max\{sup(i_j) | \forall i_j \in X\}}$
  - cosine $cosine(A, B) = \frac{P(A \cup B)}{\sqrt{P(A) \times P(B)}} = \frac{sup(A \cup B)}{\sqrt{sup(A) \times sup(B)}}$

Constraint-based association mining

- Knowledge type constraints
  - Association
  - Correlation
- Data constraints
  - Set of task-relevant data
- Dimension/level constraints
  - Desired dimensions
  - Desired levels in the concept hierarchy
- Interestingness constraints
  - Thresholds on statistical measures
- Rule constraints
  - Form of rules
- Metarule-guided mining association rules
- Constraint pushing: mining guided by rule constraints
  - Anti-monotonic
  - Monotonic
  - Succinct
  - Convertible
  - Inconvertible

Outline

- Basic concepts and road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining