CS 412 Intro. to Data Mining

การทำ Mining เพื่อหา pattern ที่เกิดขึ้นซ้ำๆ

Chapter 6. Mining Frequent Patterns, Association and Correlations: Basic Concepts and Methods

Jiawei Han, Computer Science, Univ. Illinois at Urbana-Champaign, 2017

What Is Pattern Discovery?

- What are patterns?
  - เซตของ items
  - Patterns: A set of items, subsequences, or substructures that occur frequently together (or strongly correlated) in a data set
  - Patterns จะแสดงสิ่งที่ซ่อนอยู่ใน Data ของเรา
  - Patterns represent intrinsic and important properties of datasets
- Pattern discovery: Uncovering patterns from massive data sets
- Motivation examples:
  - What products were often purchased together? สินค้าอะไรที่คนมักจะซื้อด้วยกันเสมอ ?
  - หลังจากที่ซื้อ Ipod ไปแล้วครั้งหน้าเค้าจะซื้ออะไร
  - What are the subsequent purchases after buying an iPad?
  - What code segments likely contain copy-and-paste bugs? Ex. ลืมเปลี่ยนชื่อตัวแปรให้เข้ากับ Code เรา
  - What word sequences likely form phrases in this corpus?

Pattern Discovery: Why Is It Important?

- Finding inherent regularities in a data set
- Foundation for many essential data mining tasks
  - Association, correlation, and causality analysis
  - Mining sequential, structural (e.g., sub-graph) patterns
  - Pattern analysis in spatiotemporal, multimedia, time-series, and stream data
  - Classification: Discriminative pattern-based analysis
  - Cluster analysis: Pattern-based subspace clustering
- Broad applications
  - Market basket analysis, cross-marketing, catalog design, sale campaign analysis, Web log analysis, biological sequence analysis

สำคัญเพราะทำให้ Data mining สันทำงานหลายอย่างได้

support = จำนวน transaction ที่มาสนับสนุนแนวคิดเรา < absolute / relative

Basic Concepts: k-Itemsets and Their Supports

เซตของ Item

- Itemset: A set of one or more items

Item เซตที่มีจำนวน item = k ตัว

- k-itemset: X = {x1, …, xk}

Itemset ที่มีสมาชิก 3 ตัว

  - Ex. {Beer, Nuts, Diaper} is a 3-itemset
- (absolute) support (count) of X, sup{X}: Frequency or the number of occurrences of an itemset X

transaction ที่ transactionที่มี beer อยู่

  - Ex. sup{Beer} = 3
  - Ex. sup{Diaper} = 4
  - Ex. sup{Beer, Diaper} = 3
  - Ex. sup{Beer, Eggs} = 1

TransactionID

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

- (relative) support, s{X}: The fraction of transactions that contains X (i.e., the probability that a transaction contains X)
  - Ex. s{Beer} = 3/5 = 60% → $\frac{\text{เลขที่นับเจอ}}{\text{transactionทั้งหมด}}$
  - Ex. s{Diaper} = 4/5 = 80%
  - Ex. s{Beer, Eggs} = 1/5 = 20%

relation support = สัดส่วน transaction ที่ support item set นั้น?

อัลกอริทึม = ขั้นตอนการทำงาน

Basic Concepts: Frequent Itemsets (Patterns)

- An itemset (or a pattern) X is *frequent* if the support of X is no less than a *minsup* threshold σ
- Let σ = *50%* (σ: *minsup* threshold)

ค่าตัดแบ่ง { เอา/ไม่เอา
ใช่/ไม่ใช่

For the given 5-transaction dataset

- All the frequent 1-itemsets:
  - Beer: 3/5 (60%); Nuts: 3/5 (60%)
  - Diaper: 4/5 (80%); Eggs: 3/5 (60%)
- All the frequent 2-itemsets:
  - {Beer, Diaper}: 3/5 (60%)
- All the frequent 3-itemsets?
  - None

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

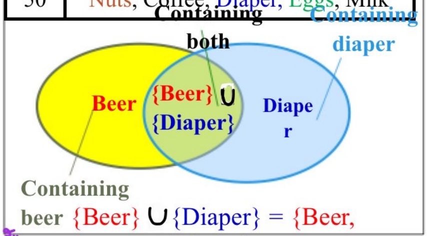
- Why do these itemsets (shown on the left) form the complete set of frequent *k*-itemsets (patterns) for any *k*?
- **Observation**: We may need an efficient method to mine a complete set of frequent patterns

From Frequent Itemsets to Association Rules

- Comparing with itemsets, rules can be more telling
  - Ex. *Diaper* → *Beer* คนซื้อ Diaper ก็จะไปซื้อ Beer ด้วย
    - *Buying diapers may likely lead to buying beers*
- How strong is this rule? (support, confidence)
  - Measuring association rules: $X \rightarrow Y$ (s, c)
    - Both X and Y are itemsets
  - Support, s: The probability that a transaction contains X ∪ Y
    - Ex. s{Diaper, Beer} = 3/5 = 0.6 (i.e., 60%)
  - Confidence, c: *The conditional probability* that a transaction containing X also contains Y
    - Calculation: c = sup(X ∪ Y) / sup(X)
    - Ex. c = sup{Diaper, Beer}/sup{Diaper} = ¾ = 0.75

Support ของ 2 อัน หารของตัวหน้า

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk



{Beer} ∪ {Diaper} = {Beer, Diaper}

Note: X ∪ Y: the union of two itemsets

- The set contains both X and Y

Mining Frequent Itemsets and Association Rules

- **Association rule mining**
  - Given two thresholds: *minsup, minconf*
  - Find all of the rules, $X \rightarrow Y$ (s, c)
    - such that, $s \geq minsup$ and $c \geq minconf$
- Let *minsup = 50%* minsup ไหนที่มี itemset เกิน 50% บ้าง
  - Freq. 1-itemsets: Beer: 3, Nuts: 3, Diaper: 4, Eggs: 3
  - Freq. 2-itemsets: {Beer, Diaper}: 3
- Let *minconf = 50%*
  - *Beer* → *Diaper* (60%, 100%)
  - *Di* (Q: Are these all rules?)) → Diaper → Beer (60%, 75%)

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

- **Observations:**
  - Mining association rules and mining frequent patterns are very close problems
  - Scalable methods are needed for mining large datasets

Efficient Pattern Mining Methods

- The Downward Closure Property of Frequent Patterns
- The Apriori Algorithm
- Extensions or Improvements of Apriori
- Mining Frequent Patterns by Exploring Vertical Data Format
- FPGrowth: A Frequent Pattern-Growth Approach
- Mining Closed Patterns

The Apriori Algorithm—An Example

เค้ากำหนด minsupport

minsup = 2

Database TDB

Tid	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

1st scan

C1

Itemset	sup
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

ตัดอันที่ support ไม่ผ่านออกไป

F1

Itemset	sup
{A}	2
{B}	3
{C}	3
{E}	3

มาสร้าง
๒ Itemset

C2

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

2nd scan

Support ไม่ถึงก็ตัด
ทิ้งไป

C2

Itemset	sup
{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

F2

Itemset	sup
{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2

C3

Itemset
{B, C, E}

3rd scan

F3

Itemset	sup
{B, C, E}	2