นายเกียรติศักดิ์ แสนจันทร์ 623021041-7

CS 412 Intro. to Data Mining

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

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

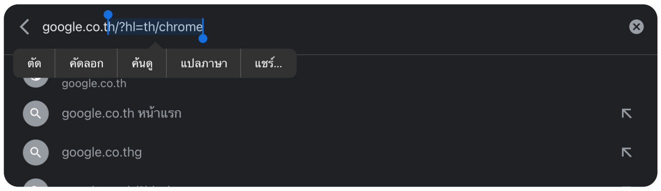
การทำ mining เพื่อหา Patterns ที่เกิดขึ้นซ้ำ

1

What Is Pattern Discovery?

- ❑ What are patterns? การค้นหา patterns ที่ซ่อนอยู่
  - ❑ Patterns: A set of items, subsequences, or substructures that occur frequently together (or strongly correlated) in a data set
    ลูกค้ามักซื้ออะไรคู่กันเสมอ (set of items)
  - ❑ Patterns represent intrinsic and important properties of datasets
- ❑ Pattern discovery: Uncovering patterns from massive data sets
  เอาไปใช้ประโยชน์อะไรได้บ้าง
- ❑ Motivation examples:
  - ❑ What products were often purchased together?
    สินค้าใดที่ลูกค้ามักจะซื้อคู่กันเสมอ ? เพราะถ้ารู้ ร้านจะได้เตรียมสินค้าทั้งสองไว้เท่าๆกัน
  - ❑ What are the subsequent purchases after buying an iPad?
    เมื่อลูกค้าซื้อไอแพดไปแล้ว จะกลับมาซื้อฟิล์ม/เคส เป็นต้น ดังนั้นร้าน อาจจัดโปรโมชั่นให้ลูกค้า
  - ❑ What code segments likely contain copy-and-paste bugs?
  - ❑ What word sequences likely form phrases in this corpus?

- ถ้าเราพิมพ์ค้นหาแบบยังไม่เต็มคำ เช่น



มันจะขึ้นคำที่เราต้องการค้นหา ซึ่งเรียนรู้จาก pattern เป็นต้น

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

สามารถเติมเลข
ต่อท้าย K ได้

Basic Concepts: k-Itemsets and Their Supports

- Itemset: A set of one or more items
- k-itemset: $X = \{x_1, \ldots, x_k\}$
  - Ex. {Beer, Nuts, Diaper} is a 3-itemset
- จำนวน transaction ที่มาสนับสนุน
- (*absolute*) *support* (*count*) of X, sup{X}: Frequency or the number of occurrences of an itemset X
  - Ex. sup{Beer} = 3
  - Ex. sup{Diaper} = 4
  - Ex. sup{Beer, Diaper} = 3
  - Ex. sup{Beer, Eggs} = 1

Absolute ไม่ดี เพราะไม่รู้
transaction ทั้งหมด

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

มักใช้ relative แทน absolute

- (*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%
  - Ex. s{Diaper} = 4/5 = 80%
  - Ex. s{Beer, Eggs} = 1/5 = 20%

จุดมุ่งหมายคือ เราต้องทำมือได้
เข้าใจการทำงานของ Data mining

Basic Concepts: Frequent Itemsets (Patterns)

ดูยังไงว่ามีความถี่ = เกิดขึ้นบ่อย

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

ค่าขีดแบ่งว่า จะเอาหรือไม่เอา

- Let σ = *50%* (σ: *minsup* threshold)

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: coffee 2/5 (40%)
  - {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

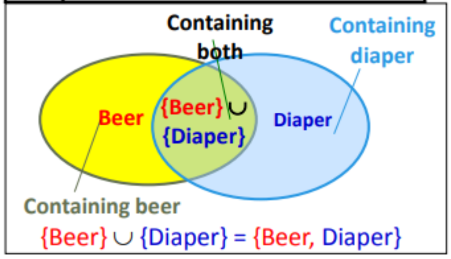
From Frequent Itemsets to Association Rules

- Comparing with itemsets, rules can be more telling
  - Ex. *Diaper → Beer* คนซื้อ Diaper จะนำไปสู่การซื้อ Beer
    - *Buying diapers may likely lead to buying beers*
- How strong is this rule? (support, confidence)
  - Measuring association rules: *X → 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*

D/B D
3/5 ÷ 4/5

    - Calculation: *c* = sup(X ∪ Y) / sup(X)
    - Ex. *c* = sup{Diaper, Beer}/sup{Diaper} = ¾ = 0.75

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



Note: X ∪ Y: the union of two itemsets

■ The set contains both X and Y

Mining Frequent Itemsets and Association Rules

ก่อนจะหาต้องกำหนด minsup, minconf

- **Association rule mining**
  - Given two thresholds: *minsup*, *minconf*
  - Find all of the rules, $X \rightarrow Y$ (s, c)
    - such that, s ≥ *minsup* and *c* ≥ *minconf*

อาจเลือกจากตัวที่มี support และ confident สูงสุด

- Let *minsup = 50%*
  - Freq. 1-itemsets: Beer: 3, Nuts: 3, Diaper: 4, Eggs: 3
  - Freq. 2-itemsets: {Beer, Diaper}: 3

สูตร

$C = Sup(X \cup Y) / sup(X)$

- Let *minconf = 50%*
  - *Beer* → *Diaper* (60%, 100%)
  - *Diaper* → *Beer* (60%, 75%)

(Q: Are these all rules?)

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 Downward Closure Property of Frequent Patterns

- ❑ Observation: From $TDB_{1:}$ T_1: {a_1, …, a_{50}}; T_2: {a_1, …, a_{100}}
  - ❑ We get a frequent itemset: {a_1, …, a_{50}}
  - ❑ Also, its subsets are all frequent: {a_1}, {a_2}, …, {a_{50}}, {a_1, a_2}, …, {a_1, …, a_{49}}, …
  - ❑ There must be some hidden relationships among frequent patterns!
- ❑ The downward closure (also called "Apriori") property of frequent patterns
  - ❑ If **{beer, diaper, nuts}** is frequent, so is **{beer, diaper}**
  - ❑ Every transaction containing {beer, diaper, nuts} also contains {beer, diaper}
  - ❑ Apriori: Any subset of a frequent itemset must be frequent
- ❑ Efficient mining methodology
  - ❑ If any subset of an itemset S is infrequent, then there is no chance for S to be frequent—why do we even have to consider S!?

A sharp knife for pruning!

Apriori Pruning and Scalable Mining Methods

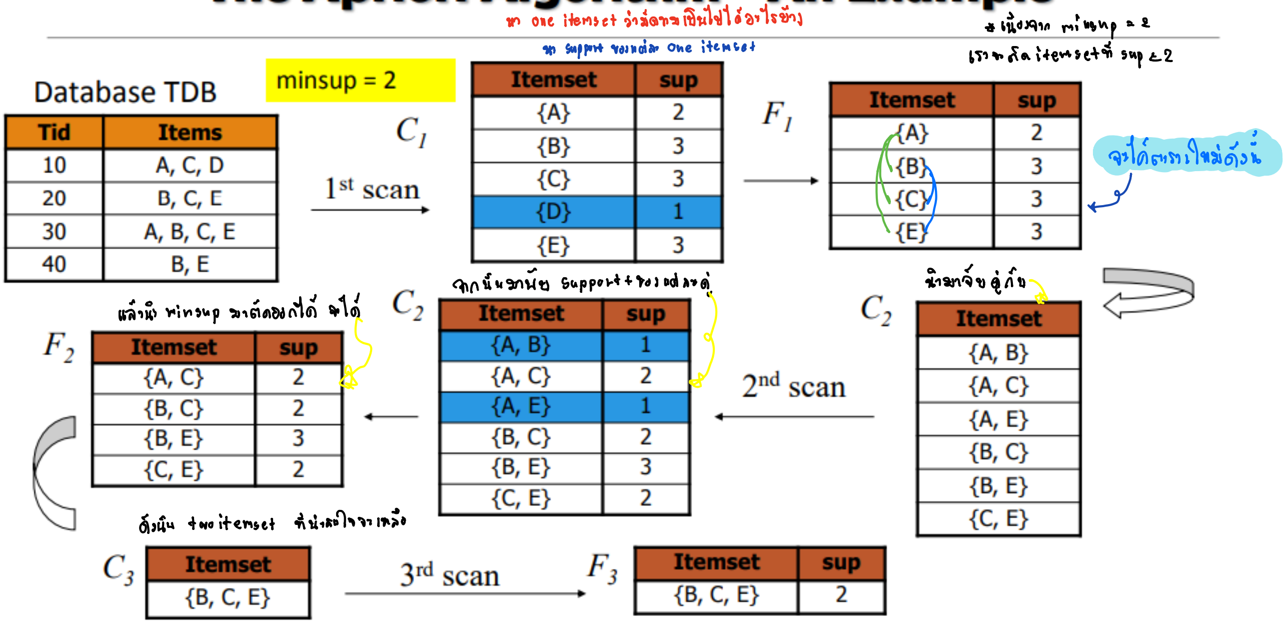
หลักการ สำคัญ

- ❑ Apriori pruning principle: If there is any itemset which is infrequent, its superset should not even be generated! (Agrawal & Srikant @VLDB'94, Mannila, et al. @ KDD' 94)
- ❑ Scalable mining Methods: Three major approaches
  - ❑ Level-wise, join-based approach: Apriori (Agrawal & Srikant@VLDB'94)
  - ❑ Vertical data format approach: Eclat (Zaki, Parthasarathy, Ogihara, Li @KDD'97)
  - ❑ Frequent pattern projection and growth: FPgrowth (Han, Pei, Yin @SIGMOD'00)

Apriori: A Candidate Generation & Test Approach

- Outline of Apriori (level-wise, candidate generation and test)
  - Initially, scan DB once to get frequent 1-itemset
  - Repeat
    - Generate length-(k+1) candidate itemsets from length-k frequent itemsets
    - Test the candidates against DB to find frequent (k+1)-itemsets
    - Set k := k +1
  - Until no frequent or candidate set can be generated
  - Return all the frequent itemsets derived

The Apriori Algorithm—An Example



หา one itemset ว่ามีตัวไหนเป็นไปได้อะไรบ้าง

หา support ของแต่ละ one itemset

* เนื่องจาก minsup = 2
เราจะตัด itemset ที่ sup < 2

Database TDB

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

minsup = 2

1st scan

C_1

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

F_1

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

จะได้ตารางใหม่ดังนี้

นำมาจับคู่กัน

C_2

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

2nd scan

จากนั้นมานับ support ของแต่ละคู่

C_2

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

แล้วนำ minsup มาตัดออกก็ได้ จะได้

F_2

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

ดังนั้น two itemset ที่น่าสนใจจะเหลือ

C_3

Itemset
{B, C, E}

3rd scan

F_3

Itemset	sup
{B, C, E}	2