UNIVERSITY of HOUSTON

DEPARTMENT OF COMPUTER SCIENCE

Association Rule Mining

Rakesh Verma

Why Association Rules? Some security applications

Malware detection [e.g. Ding et al. Computers & Security 2013]

Hypothesis: malicious behavior exhibited by system calls.

Data: API calls and frequencies: Obtained from Windows PE head file

Stepping-stone detection [e.g. Hsiao et al. Sec. and Comm. Networks 2013]

Stepping stones are intermediate hosts on the path from an hacker to a victim

Network connection records: Each transaction contains a number of pairs (s, t) where s, t are IP addresses, s – source, t - destination

What are Frequent Item Sets?

Originally proposed by Agrawal, Imielinski, and Swami in Journal of the Association for Information Systems (1993).

Frequent Itemsets are a frequently occurring pattern in data.

Applications:

- Shopping Cart analysis
    - What do people frequently buy together.
- DNA sequence analysis
    - Which genes react to certain medication?
- Website Traffic
    - Which sites does someone who uses Reddit a lot also go to?

Frequent Patterns

Itemset: A non-empty set of items.

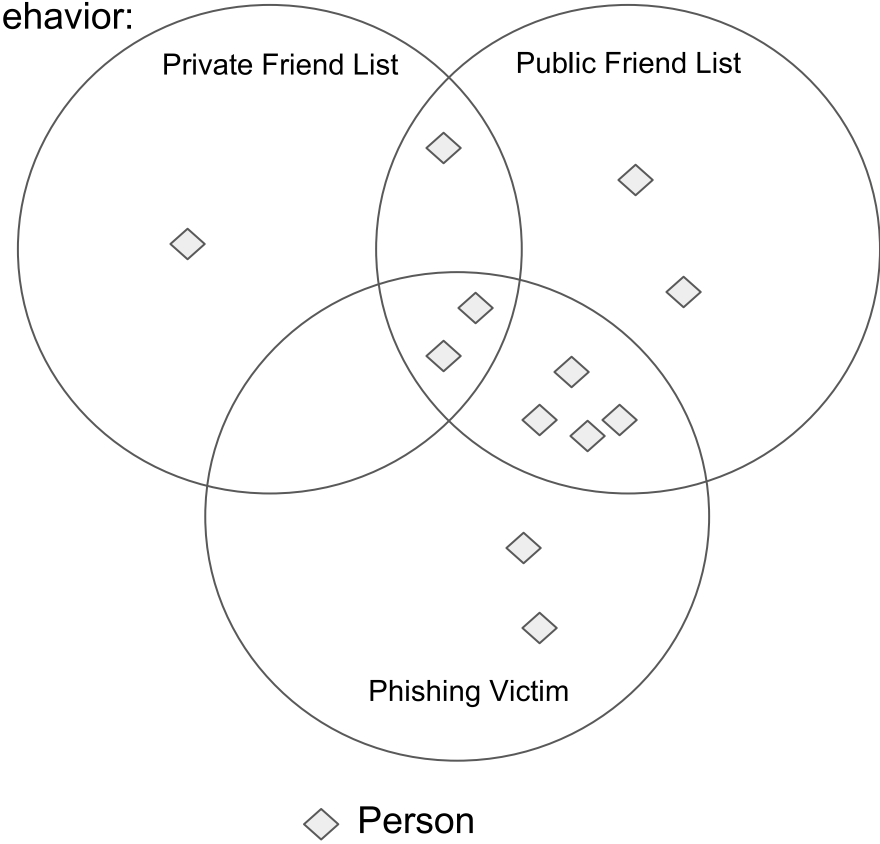
k-Itemset: An itemset with **k** elements.

Support: The frequency of occurrences of a specific itemset in the dataset.

Relative Support: The probability of an itemset in the dataset.

Frequent Itemset: An itemset is frequent if it occurs as many times as the **minimum support threshold**.

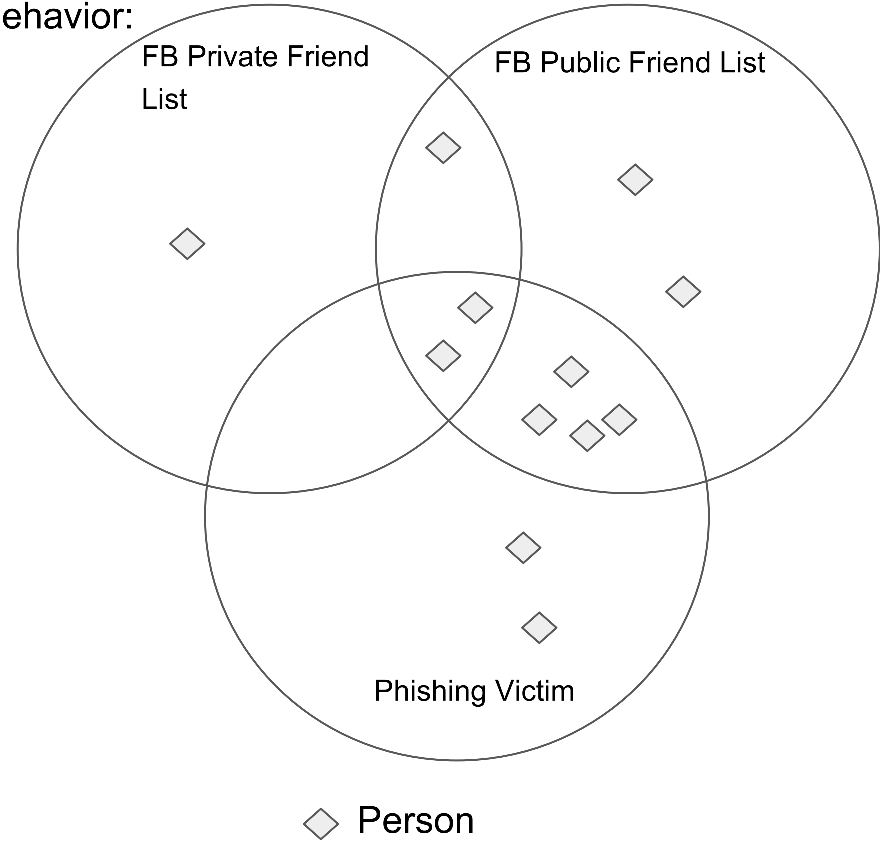
Social Network Behavior:



Association Rules

- **Objective**: Find all rules **X→Y** within the minimum support threshold and minimum confidence threshold.
  - Confidence: The probability **P(Y|X)**, the probability of an itemset having **Y** if it already has **X**.
- Public Friend List→ Phishing Victim
  - Support: 6 out of 12
  - Confidence: 66.67%
- Phishing Victim → Public Friend List
  - Support: 6 out of 12, Confidence: 75%

Social Network Behavior:



Issues with finding pattern

Long patterns contain exponentially many sub-patterns.

If a pattern contains **N** items, there are $\mathbf{2^N}$ sub-patterns.

Dealing with exponential anything is too computationally expensive.

Alternatives to finding every rule is to find **closed patterns** and **max-patterns**.

Closed patterns: An itemset is closed if there's no itemset that contains it with the same support count.

Using only closed patterns is akin to compression.

Max-patterns: An itemset is a max-pattern if there's no frequent itemset that contains it.

Downward Closure Property and Scalable Mining

If **X** is frequent, then every subset of **X** is frequent.

- **“FB public friends”, “Myspace public friends”, “Phishing victim”** is frequent, therefore **“FB public friends”, “Myspace public friends”** and **“FB public friends”, “Phishing victim”** are, too.

Scalable Mining Methods:

- Apriori
- Frequent Pattern Growth
- Vertical Data Format

Apriori

Apriori Pruning Principle

If there is an infrequent itemset, do not test or generate its supersets.

Method

Collect all frequent 1-itemsets.

From all collected k-itemsets, generate candidate (k+1)-itemsets.

Prune candidates of infrequent itemsets and collect the frequent ones.

Repeat until no new candidates can be generated or all candidates generated in the last pass were pruned.

Example Run of Apriori

Frequent 1-itemsets

A, B, C, D

Generate Candidates

AB, AC, AD, BC, BD, CD

Frequent 2-itemsets

AC, BC

Generate Candidates

No candidates to generate

ABC contains AB which is infrequent.

Frequent Itemsets:

A, B, C, D, AC, BC

Dataset

A, C, D

B, C

A, B, C

B, D

Minimum Support Threshold: 2

How to Count Candidate's Supports

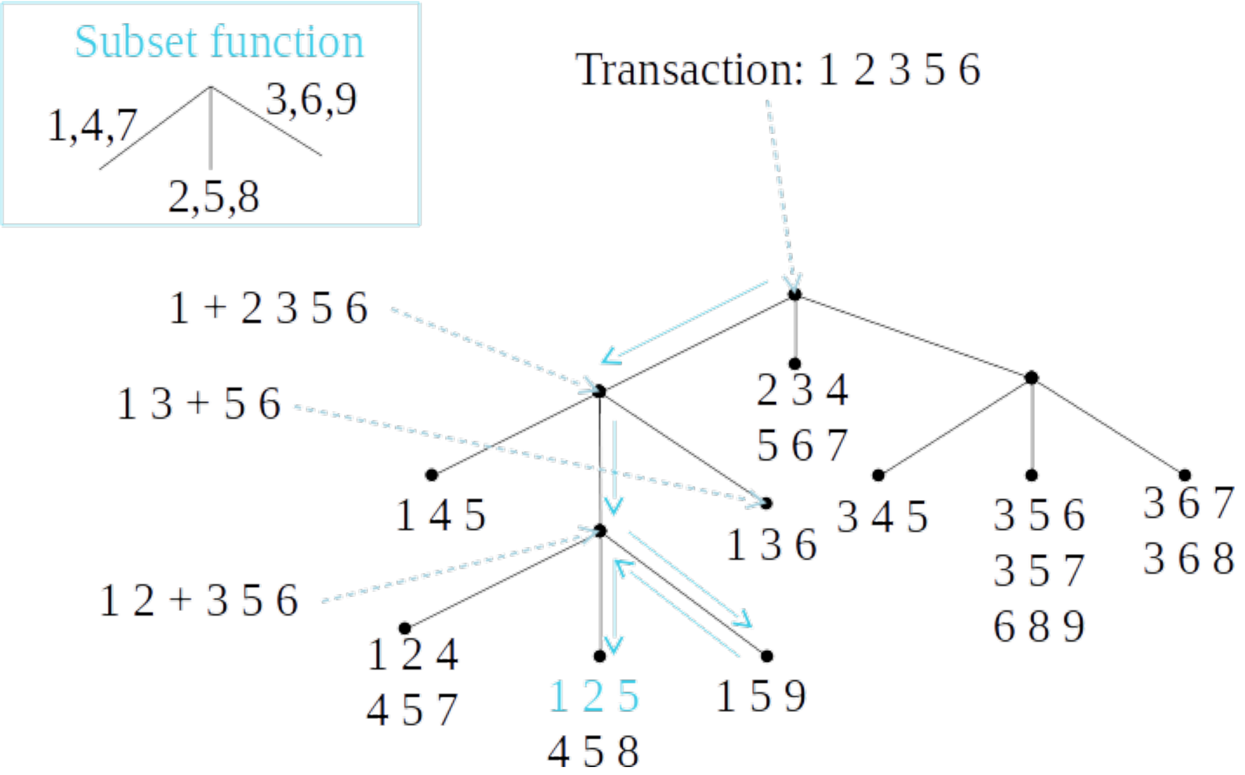
Calculating Candidate's Supports is computationally intensive.

- For every k-itemset, there are up to N-k candidate (k+1)-itemsets, where N is the number of distinct items in the dataset.

The Hashtree Method

- Candidate itemsets are stored in a hashtree.
- **Leaves**: Lists of itemsets and counts.
- **Interior Nodes**: Hashtable.
- **Subset function**: Find all candidates contained in a transaction.

Counting Candidate Support Using a Hashtree



Generating Association Rules from Frequent Itemsets

Frequent itemsets are not the same thing as association rules.

X→**Y** is an association rule if

- **X** and **Y** are disjoint and nonempty.
- Support of **X**→**Y** = the support of **X** ∪ **Y**.
- Confidence of **X**→**Y** = the support of **X** ∪ **Y** / the support of **X**.
- Confidence of **X**→**Y** ≥ minimum confidence threshold.

Example:

- **"FB Private Friend", "Phishing victim"** has support 50%
  - **FB Private Friend** and **Phishing victim** separately have support 75%
  - **FB Private Friend** → **Phishing victim** is an association rule with support 50% and confidence 66.67%
  - **Phishing victim** → **FB Private Friend** is an association rule with support 50% and confidence 66.67%