Towards Proximity Pattern Mining in Large Graphs

Arijit Khan  
Computer Science Department  
University of California, Santa Barbara  
arijitkhan@cs.ucsb.edu

Xifeng Yan  
Computer Science Department  
University of California, Santa Barbara  
xyan@cs.ucsb.edu

Kun-Lung Wu  
IBM T. J. Watson, Hawthorne, NY  
klwu@us.ibm.com
Motivation

Towards Proximity Pattern Mining in Large Graphs

Homophily in Social Network

Last.FM

Nodes -&gt; Users

Edges -&gt; Links

List of Musical Bands/ Singers

What are the related Musical Bands/ Singers that co-occur frequently in neighborhood?
Motivation

Towards Proximity Pattern Mining in Large Graphs

Last.FM
Nodes -> Users
Edges -> Links
List of Musical Bands/ Singers

What are the related Musical Bands/ Singers that co-occur frequently in neighborhood?

Homophily in Social Network

- Katy Perry, Madonna
- Britney Spears
- Metallica, Megadeth
- Megadeth, Slayer
- Lady Gaga
- Beyonce, Madonna
- Britney Spears, Lady Gaga
- Metallica
- Megadeth, Slayer
**Motivation**

Towards Proximity Pattern Mining in Large Graphs

**Last.FM**

Nodes -> Users

Edges -> Links

List of Musical Bands/ Singers

What are the related Musical Bands/ Singers that co-occur frequently in neighborhood?

Homophily in Social Network
Motivation

Towards Proximity Pattern Mining in Large Graphs
Motivation

Towards Proximity Pattern Mining in Large Graphs

Motivation

Intrusion Network

What are Related Computer Attacks that Co-occur Frequently in Neighborhood?

Computers in LAN

Computers in Same LAN Attacked by Similar Intrusions

TFTP_Put, Ping_Flood

TFTP_Put

TFTP_Put, ICMP_Flood

Audit_TFTP_Get_Filename

SQL_SSRP_Slammer_Worm

SQL_SSRP_StackBo
Roadmap

- **Problem Formulation**
  - Problem Definition
  - Preliminaries

- **Framework**
  - Neighborhood Association Model
  - Information Propagation Model

- Probabilistic Itemset Mining

- Experimental Results

- Conclusion
Problem Definition

Mining Proximity Patterns in Large Graphs.

CHARACTERISTICS

- Proximity
- Frequency

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Proximity</th>
</tr>
</thead>
<tbody>
<tr>
<td>a, b</td>
<td>YES</td>
</tr>
<tr>
<td>a, b, c</td>
<td>YES</td>
</tr>
<tr>
<td>d, e, f</td>
<td>NO</td>
</tr>
</tbody>
</table>
Problem Definition

- Will Frequent Subgraph Mining Work? - **NO !!!**

- **Flexibility**

- Will Frequent Itemset Mining Work? - **NO !!!**

- No Notion of Edge in Frequent Itemset Mining

\{a, b, c\}

Frequent Subgraph – No
Frequent Itemset - No
Proximity Pattern - Yes
Preliminaries

- Labeled Graph $G = (V, E, L)$

- Item Set $I \subseteq L$ is a subset of Labels.

- **SUPPORT**: The support $sup(I)$ of an itemset $I \subseteq L$ is the number of transactions in the data set that contain $I$.

- **DOWNWARD CLOSURE**: For a frequent itemset, all of its subsets are frequent; and thus for an infrequent itemset, all of its superset must be infrequent.
Roadmap

- Problem Formulation
  - Problem Definition
  - Preliminaries

- Framework
  - Neighborhood Association Model
  - Information Propagation Model

- Probabilistic Itemset Mining

- Experimental Results

- Conclusion
Neighborhood Association Model

- **EMBEDDING:**
  - \( \{v_1, v_2, v_3\} \) an embedding of \( \{a, b, e\} \) with two possible Mappings:
    - \( \Phi_1: a \) to \( v_2 \), \( b \) to \( v_1 \), \( e \) to \( v_3 \)
    - \( \Phi_2: a \) to \( v_2 \), \( b \) to \( v_3 \), \( e \) to \( v_3 \)
  - \( f(\pi) \) measures how tightly the mapped labels in the embedding \( \pi \) are connected. i.e., the inverse of diameter of \( \pi \)

- **SUPPORT:** Find all embeddings \( \pi_1, \pi_2, \ldots, \pi_m \) of an itemset \( I \). Define \( \text{sup}(I) = \sum_i f(\pi_i) \).
Neighborhood Association Model

- Overlap + Not Downward Closure !!!

- Use **maximum independent set** of all embeddings of an itemset. (S. N. Bringmann, PAKDD’08)

- \( \text{Sup}(a, b) = f(\pi_1) + f(\pi_4) \).

- Downward Closure.

- Finding the maximum independent set is NP-hard

Embeddings of \( \{a, b\} \)
Information Propagation Model

- Influence Based Information Propagation.

- Information Propagation is modeled using First Order Markov Model.

- Labels are propagated with certain probability from each node to its neighbors.

- Labels are propagated independent to each other.
Information Propagation Model

- **NEAREST PROBABILISTIC ASSOCIATION (NPA):**
  - If label $l$ present in node $u$, $A_u(l) = 1$.
  - Otherwise, propagate $l$ to $u$ from its immediate neighbor $v$.
  - $A_u(l) = A_v(l) \cdot e^{-\alpha}$
  - $\alpha > 0$ is the decay constant.
  - Recursive to propagate beyond one hop.

- **SUPPORT:**
  $$sup(I) = (1/|V|) \sum_{u \in V} A_u(l_1) \ldots A_u(l_m)$$
  $$I = \{l_1, \ldots, l_m\}.$$
Information Propagation Model

- Downward Closure.
- Consistent with graph structure.

**Table (a)**

<table>
<thead>
<tr>
<th></th>
<th>$l_1$</th>
<th>$l_2$</th>
<th>$l_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>node$_1$</td>
<td>1</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>node$_2$</td>
<td>0.37</td>
<td>1</td>
<td>0.37</td>
</tr>
<tr>
<td>node$_3$</td>
<td>0.37</td>
<td>0.37</td>
<td>1</td>
</tr>
</tbody>
</table>

$\text{Sup}(l_1, l_2, l_3) = 0.14$

**Table (b)**

<table>
<thead>
<tr>
<th></th>
<th>$l_1$</th>
<th>$l_2$</th>
<th>$l_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>node$_1$</td>
<td>1</td>
<td>0.37</td>
<td>0.14</td>
</tr>
<tr>
<td>node$_2$</td>
<td>0.37</td>
<td>1</td>
<td>0.37</td>
</tr>
<tr>
<td>node$_3$</td>
<td>0.14</td>
<td>0.37</td>
<td>1</td>
</tr>
</tbody>
</table>

$\text{Sup}(l_1, l_2, l_3) = 0.08$
Information Propagation Model

PROBLEM WITH NEAREST PROBABILISTIC ASSOCIATION (NPA):

\[ \text{sup}(l_1, l_2) = 0.37 \]

Towards Proximity Pattern Mining in Large Graphs
Information Propagation Model

- NORMALIZED PROBABILISTIC ASSOCIATION (NmPA):

$$A_u(l) = A_v(l) \cdot \left[ \frac{m}{n+1} \right] e^{-\alpha}$$

- $m$ = # of 1-hop neighbors of $u$ containing label $l$.
- $n$ = # of 1-hop neighbors of $u$.

$$sup(l_1, l_2) = 0.37 \times (1/2) = 0.19$$
$$sup(l_1, l_2) = 0.37 \times (2/3) = 0.25$$
Roadmap

- Problem Formulation
  - Problem Definition
  - Preliminaries

- Framework
  - Neighborhood Association Model
  - Information Propagation Model

- Probabilistic Itemset Mining

- Experimental Results

- Conclusion
Probabilistic Itemset Mining

- **Frequent-Pattern (FP) Tree** cannot handle fractional association values because of the new definition of Support.
- Modify FP Tree Structure and Algorithm.
- C. C. Aggarwal *et. al* (KDD '09), Bernecker *et. al* (KDD '09).

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00</td>
<td>0.12</td>
<td>0.00</td>
<td>0.12</td>
</tr>
<tr>
<td>2</td>
<td>0.19</td>
<td>0.00</td>
<td>0.00</td>
<td>1.00</td>
</tr>
<tr>
<td>3</td>
<td>0.12</td>
<td>1.00</td>
<td>0.12</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>0.00</td>
<td>0.19</td>
<td>1.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>
Probabilistic Itemset Mining

- **Probabilistic FP-Growth (pFP):** associating a **bucket** with each node of the FP-tree.

<table>
<thead>
<tr>
<th>transaction id</th>
<th>$l_1$</th>
<th>$l_2$</th>
<th>$l_3$</th>
<th>$l_4$</th>
<th>$l_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0.3</td>
<td>0</td>
<td>0</td>
<td>0.1</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0.2</td>
<td>0.5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.2</td>
<td>0.5</td>
<td>0</td>
<td>0.05</td>
</tr>
</tbody>
</table>

![Diagram of FP-tree with probabilities]
Probabilistic Itemset Mining

- PROBLEMS WITH PROBABILISTIC FP-TREE (pFP): slow because of frequent disk access to load and store the buckets.

- Is it possible to approximate the buckets so that the complete tree can be loaded in the main memory?

- Approximate FP-Tree (aFP)
Probabilistic Itemset Mining

- APPROXIMATE FP-TREE (aFP):

\[ \text{sup}(l_1, l_2) = 0.4 \]

\[ \text{sup}(l_1, l_2) = 0.35 \]

\[ \tilde{A}(l_x, l_y) = \frac{\text{sum}(v_x) \cdot \text{sum}(v_y)}{\max\{\text{occurrence}(v_x), \text{occurrence}(v_y)\}} \]
Top-k Interesting Pattern Mining

- How to measure “Interesting-ness”? – Randomization Test.

- Generate graph Q from graph G by randomly swapping the labels among nodes. Let, $p$ and $q$ be the support values of itemset $I$ in $G$ and $Q$ respectively. High difference indicates interestingness.

- G-test Score: $p \cdot \ln \frac{p}{q} + (1-p) \cdot \ln \frac{1-p}{1-q}$

- Vertical Pruning by Yan et. al (SIGMOD ’08).

- Proximity Patterns minus Frequent Patterns.
Roadmap

- Problem Formulation
  - Problem Definition
  - Preliminaries

- Framework
  - Neighborhood Association Model
  - Information Propagation Model

- Probabilistic Itemset Mining

- Experimental Results

- Conclusion
Experimental Results

DATASET:

<table>
<thead>
<tr>
<th></th>
<th># of Nodes</th>
<th># of Edges</th>
<th># of Labels</th>
<th>Avg. # of Labels/ Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last.FM</td>
<td>6,899</td>
<td>58,179</td>
<td>6,340</td>
<td>3</td>
</tr>
<tr>
<td>Intrusion</td>
<td>200,858</td>
<td>703,020</td>
<td>1,000</td>
<td>25</td>
</tr>
<tr>
<td>DBLP</td>
<td>684,911</td>
<td>7,764,604</td>
<td>130</td>
<td>9</td>
</tr>
</tbody>
</table>

EFFICIENCY:

<table>
<thead>
<tr>
<th></th>
<th>Last.FM</th>
<th>Intrusion</th>
<th>DBLP</th>
</tr>
</thead>
<tbody>
<tr>
<td>NmPA</td>
<td>2.0 sec</td>
<td>5.0 sec</td>
<td>187.0 sec</td>
</tr>
<tr>
<td>FP-Tree Formation</td>
<td>1.0 sec</td>
<td>10.0 sec</td>
<td>89.0 sec</td>
</tr>
<tr>
<td>Top-k Mining</td>
<td>4.0 sec</td>
<td>2.0 sec</td>
<td>254.0 sec</td>
</tr>
</tbody>
</table>
Experimental Results

- EFFECTIVENESS (Last.FM):

<table>
<thead>
<tr>
<th>#</th>
<th>Proximity Patterns</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tiësto, Armin van Buuren, ATB</td>
<td>0.62</td>
</tr>
<tr>
<td>2</td>
<td>Katy Perry, Lady Gaga, Britney Spears</td>
<td>0.58</td>
</tr>
<tr>
<td>3</td>
<td>Ferry Corsten, Tiësto, Paul van Dyk</td>
<td>0.55</td>
</tr>
<tr>
<td>4</td>
<td>Neaera, Caliban, Cannibal Corpse</td>
<td>0.52</td>
</tr>
<tr>
<td>5</td>
<td>Lucuna Coil, Nightwish, Within Temptation</td>
<td>0.47</td>
</tr>
</tbody>
</table>

- ATB, Paul van Dyk – **German DJ**
- Tiësto, Ferry Corsten, Armin van Buuren – **Dutch DJ**
- Britney Spears, Lady Gaga, Katy Gaga – **American Female Pop Singers**
- Neaera, Caliban, Cannibal Corpse – **Death Metal Bands**
- Lucuna Coil, Nightwish, Within Temptation – **Gothic Metal Bands**
## Experimental Results

### EFFECTIVENESS (Intrusion):

<table>
<thead>
<tr>
<th>#</th>
<th>Interesting Patterns</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ICMP_Flood, Ping_Flood</td>
<td>0.94</td>
</tr>
<tr>
<td>2</td>
<td>Email_Error, SMTP_Relay_Not_Allowed, HTML_NullChar_Evasion</td>
<td>0.94</td>
</tr>
<tr>
<td>3</td>
<td>Image_RIFF_Malformed, HTML_NullChar_Evasion</td>
<td>0.90</td>
</tr>
<tr>
<td>4</td>
<td>TFTP_Put, Ping_Flood, Audit_TFTP_Get_Filename</td>
<td>0.80</td>
</tr>
<tr>
<td>5</td>
<td>Email_Command_Overflow, Email_Virus_Double_Extension, Email_Error</td>
<td>0.75</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#</th>
<th>Interesting Patterns</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ping_Sweep, Smurf_Attack</td>
<td>2.42</td>
</tr>
<tr>
<td>2</td>
<td>TFTP_Put, Audit_TFTP_Get_Filename, ICMP_Flood, Ping_Flood</td>
<td>2.32</td>
</tr>
<tr>
<td>3</td>
<td>TCP_Service_Sweep, Email_Error</td>
<td>1.21</td>
</tr>
<tr>
<td>4</td>
<td>HTML_Outlook_MailTo_Code_Execution, HTML_NullChar_Evasion</td>
<td>1.15</td>
</tr>
<tr>
<td>5</td>
<td>SQL SSRP Slammer Worm, SQL SSRP StackBo</td>
<td>0.88</td>
</tr>
</tbody>
</table>

Proximity Patterns

Proximity Patterns Minus Frequent Patterns
Experimental Results

- **SCALIBILITY**

![Graph showing Information Propagation (NmPA) Time vs. No. of Nodes](image)

- Information Propagation (NmPA) Time vs. No. of Nodes
Experimental Results

- **SCALIBILITY**

 Mining Time vs. No. of Nodes

Towards Proximity Pattern Mining in Large Graphs
Experimental Results

- pFP (Exact Mining) vs. aFP (Approximate Mining)

[Last.FM]:

<table>
<thead>
<tr>
<th>#</th>
<th>Proximity Patterns</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tiësto, Armin van Buuren, ATB</td>
<td>0.62</td>
</tr>
<tr>
<td>2</td>
<td>Katy Perry, Lady Gaga, Britney Spears</td>
<td>0.58</td>
</tr>
<tr>
<td>3</td>
<td>Ferry Corsten, Tiësto, Paul van Dyk</td>
<td>0.55</td>
</tr>
<tr>
<td>4</td>
<td>Neaera, Caliban, Cannibal Corpse</td>
<td>0.52</td>
</tr>
<tr>
<td>5</td>
<td>Lacuna Coil, Nightwish, Within Temptation</td>
<td>0.47</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#</th>
<th>Proximity Patterns</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Katy Perry, Lady Gaga, Britney Spears</td>
<td>0.58</td>
</tr>
<tr>
<td>2</td>
<td>Ferry Corsten, Tiësto, Paul van Dyk</td>
<td>0.55</td>
</tr>
<tr>
<td>3</td>
<td>Tiësto, Armin van Buuren, ATB</td>
<td>0.55</td>
</tr>
<tr>
<td>4</td>
<td>Neaera, Caliban, Cannibal Corpse</td>
<td>0.51</td>
</tr>
<tr>
<td>5</td>
<td>Lacuna Coil, Nightwish, Within Temptation</td>
<td>0.46</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Steps</th>
<th>aFP(approximate)</th>
<th>pFP(exact)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FP-tree Formation</td>
<td>1.0</td>
<td>3.0</td>
</tr>
<tr>
<td>Top-k Pattern Mining</td>
<td>4.0</td>
<td>21.0</td>
</tr>
</tbody>
</table>

Table 10: Runtime Comparison (sec) (Last.fm)
Roadmap

- Problem Formulation
  - Problem Definition
  - Preliminaries

- Framework
  - Neighborhood Association Model
  - Information Propagation Model

- Probabilistic Itemset Mining

- Experimental Results

- Conclusion
Conclusion

- Novel Concept of Proximity Pattern Mining in Large Graphs.


- Effective, Efficient and Scalable framework.

- How to determine the optimal propagation measure and depth?
Questions ??

Thank You !