Structural Correlation Pattern Mining for Large Graphs

Master’s Thesis Defense

Arlei Lopes da Silva

Computer Science Department
ICEX
Universidade Federal de Minas Gerais – Brasil
Agenda

Introduction

Background

Structural Correlation Pattern Mining: Definitions

Structural Correlation Pattern Mining: Algorithms

Experimental Evaluation

Conclusions
Introduction
Graphs

- Have been established as a powerful theoretical framework for modeling several types of interactions
- Have multidisciplinary applications
Graph Analysis Framework
Graph Analysis Framework: Systems

- Examples:
  - Internet
  - Web
  - Social networks
  - Molecular complexes
  - Proteins and their interactions
  - Transportation routes
  - ...

- Huge amounts of data available today
Graph Analysis Framework

SYSTEM

GRAPH
Graph Analysis Framework: Graph Representations

- Examples:
  - Simple
  - Directed
  - Weighted
  - Bipartite
  - Attributed
  - ...

- More data motivates more complex representations
Graph Analysis Framework
Graph Analysis Framework: Metrics and Patterns

- Examples:
  - Degree distribution
  - Clustering coefficient
  - Frequent subgraphs
  - Dense subgraphs
  - ...

- Complex representations may support new metrics and patterns
This Thesis: Attributed Graphs

<table>
<thead>
<tr>
<th>vertex</th>
<th>attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A, C</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A, C, D</td>
</tr>
<tr>
<td>4</td>
<td>A, D</td>
</tr>
<tr>
<td>5</td>
<td>A, E</td>
</tr>
<tr>
<td>6</td>
<td>A, B, C</td>
</tr>
<tr>
<td>7</td>
<td>A, B, E</td>
</tr>
<tr>
<td>8</td>
<td>A, B</td>
</tr>
<tr>
<td>9</td>
<td>A, B</td>
</tr>
<tr>
<td>10</td>
<td>A, B, D</td>
</tr>
<tr>
<td>11</td>
<td>A, B</td>
</tr>
</tbody>
</table>

Vertex attributes

Graph

- Vertex Attributes:
  - Social networks: personal characteristics (e.g., age, gender, interests)
  - Protein-protein networks: expression or annotation data
  - Web graph: content (keywords, tags)
This Thesis: Attributed Graphs

- Most of the existing metrics and patterns for graphs are based on the graph topology (i.e., vertices and edges)
  - Degree distribution
  - Clustering coefficient
  - Cliques
  - Motifs
  - ...
- Vertex attributes may provide new and useful knowledge from real graphs
  - How vertex attributes are related to the graph topology?
  - How can extend existing metrics and patterns in order to consider the graph topology and vertex attributes?
This Thesis: Dense subgraphs

- **Dense subgraph**: Set of vertices with high cohesion (i.e., strong connections among themselves)
- Real graphs are naturally organized into dense subgraphs:
  - Communities in social networks
  - Cyber-communities in the web graph
  - Molecular complexes in protein-protein networks
This Thesis: Vertex Attributes + Dense Subgraphs

- How do vertex attributes are associated with the dense subgraphs in real graphs?
  - How a particular set of interests induces communities in a social network?
  - How sets of keywords are related to cyber-communities in the web graph?
This Thesis: Vertex Attributes + Dense Subgraphs

- How do vertex attributes are associated with the dense subgraphs in real graphs?
  - How a particular set of interests induces communities in a social network?
  - How sets of keywords are related to cyber-communities in the web graph?

- **Structural correlation**: Measures how an attribute set induces dense subgraphs in an attributed graph
This Thesis: Vertex Attributes + Dense Subgraphs

- How do vertex attributes are associated with the dense subgraphs in real graphs?
  - How a particular set of interests induces communities in a social network?
  - How sets of keywords are related to cyber-communities in the web graph?
- **Structural correlation**: Measures how an attribute set induces dense subgraphs in an attributed graph
- **Structural correlation pattern**: Dense subgraph induced by a particular attribute set
This Thesis

- Proposes correlating attribute sets and dense subgraphs
- Defines the structural correlation pattern mining
- Presents algorithms for structural correlation pattern mining
- Applies structural correlation pattern mining to real datasets
Background
Definition: Maximal vertex set $V$ such that for each $v \in V$, the degree of $v$ in $V$ is, at least, $\lceil \gamma_{\text{min}}(|V| - 1) \rceil$

- $\gamma_{\text{min}}$: minimum density threshold
- $\text{min}_\text{size}$: minimum size threshold

A clique is an 1-quasi-clique

Figure: 0.6-quasi-clique
Quasi-clique Discovery

- **Definition:** Consists of finding the set $Q$ of maximal quasi-cliques from a graph $G(V, E)$
  - $V$: set of vertices
  - $E$: set of edges
  - $\gamma_{min}$ and $\text{min}_\text{size}$: quasi-clique parameters

- **Complexity:** The quasi-clique discovery problem is $\#P$-complete
  - $\#P$ is the counting analogue of the NP-complete class of problems
Quasi-clique Discovery: Search-space

Diagram showing a search-space tree with nodes labeled with sets of numbers, indicating different configurations or states in the search space.
Quasi-clique Discovery: Pruning

- **Anti-monotonicity:** The anti-monotonicity property does not apply for quasi-cliques
  - A property \( \varphi \) is called anti-monotone if and only if for any patterns \( P_1 \) and \( P_2 \), the fact that \( \varphi(P_1) \) holds implies that \( \varphi(P_2) \) holds if \( P_2 \subseteq P_1 \)
  - Anti-monotonicity based pruning is very popular in data mining

![Diagram](image-url)

- 0.6-quasi-clique
- not a 0.6-quasi-clique
Quasi-clique Discovery: Vertex Pruning

- Prunes vertices that can not be part of any quasi-clique

- **Diameter-based pruning:** Let $N^k(v)$ be the size $k$ neighborhood of a vertex $v$. If $|N^D(\gamma_{min},min\_size)(v)| < min\_size - 1$, $v$ can not be in any $\gamma_{min}$-quasi-clique of size greater or equal to $min\_size$

\[
D(\gamma_{min}, \alpha) \begin{cases} 
= 2 & \text{if } \frac{\alpha-2}{\alpha-1} \geq \gamma_{min} \geq 0.5 \\
= 1 & \text{if } 1 \geq \gamma_{min} > \frac{\alpha-2}{\alpha-1} 
\end{cases}
\]
Let $\gamma_{\text{min}} = 0.5$ and $\text{min}_\text{size} = 6$

- Minimum neighborhood size is $\text{min}_\text{size} - 1 = 5$
- $D(0.5, 6) \leq 2$
Let $\gamma_{\min} = 0.5$ and $\text{min}_\text{size} = 6$

- Minimum neighborhood size is $\text{min}_\text{size} - 1 = 5$
- $D(0.5, 6) \leq 2$
- $N^2(1) = 4$ (pruned)
Quasi-clique Discovery: Candidate Quasi-clique Pruning

- Prunes sets of vertices that can not be quasi-cliques
- Candidate quasi-cliques are generated based on two vertex sets:
  - \( X \): Current vertex set (initially set as \( \emptyset \))
  - \( \text{candExts}(X) \): Candidate extensions of \( X \) (initially set as \( \mathcal{V} \))
  - Vertices from \( \text{candExts}(X) \) are moved to \( X \) iteratively until all possible candidate quasi-cliques are generated
Quasi-clique Discovery: Candidate Quasi-clique Pruning

Set Enumeration Tree

Stack
Quasi-clique Discovery: Candidate Quasi-clique Pruning

Set Enumeration Tree

Stack

- \{1\}
- \{2,3,4\}
- \{2\}, \{3,4\}
- \{3\}, \{4\}
- \{4\}, \{\}\n
- \{(1),\{2,3,4\}\}
- \{(2),\{3,4\}\}
- \{(3),\{4\}\}
- \{(4),\{\}\\}

- \{(1),\{2\}\}
- \{(1),\{3\}\}
- \{(1),\{4\}\}
- \{(2),\{3\}\}
- \{(2),\{4\}\}
- \{(3),\{4\}\}

- \{(1,2),\{3\}\}
- \{(1,2),\{4\}\}
- \{(1,3),\{4\}\}
- \{(2,3),\{4\}\}

- \{(1,2,3)\}
- \{(1,2,4)\}
- \{(1,3,4)\}
- \{(2,3,4)\}
Quasi-clique Discovery: Candidate Quasi-clique Pruning

Set Enumeration Tree

Stack
Quasi-clique Discovery: Candidate Quasi-clique Pruning

Set Enumeration Tree

Stack
Lookahead pruning: For a pair \((X, candExts(X))\), the set \(X \cup candExts(X)\) can be checked before extending \(X\). If \(X \cup candExts(X)\) is a \(\gamma_{min}\)-quasi-clique, all the extensions of \(X\) can be pruned, since they cannot be larger than \(|X \cup candExts(X)|\).
Quasi-clique Discovery: Lookahead Candidate Quasi-clique Pruning

- Let $\gamma_{min} = 0.5$ and $min\_size = 6$.
- Let $X = \{\}$ and $candExts(X) = \{6, 7, 8, 9, 10, 11\}$
Quasi-clique Discovery: Lookahead Candidate Quasi-clique Pruning

- Let $\gamma_{\text{min}} = 0.5$ and $\text{min\_size} = 6$.
- Let $X = \emptyset$ and $\text{candExts}(X) = \{6, 7, 8, 9, 10, 11\}$
- $X \cup \text{candExts}(X)$ is a quasi-clique
- **Pruned:** $\langle \{6\}, \{7, 8, 9, 10, 11\} \rangle$, $\langle \{6, 7\}, \{8, 9, 10, 11\} \rangle$, ...
Frequent Itemset Mining

Definition: Consists of identifying the set of frequent itemsets $\mathcal{F}$, such that $\mathcal{F} = \{X \subseteq \mathcal{I} | \text{support}(X) \geq \text{min}_\text{sup}\}$

- $\mathcal{I}$: set of items
- $\mathcal{D} = \langle T_1, T_2, \ldots T_n \rangle$: database
- $T_i \subseteq \mathcal{I}$: transaction
- $\text{support}(X) = \{|\{ T \in \mathcal{D} | X \subseteq T \}|$
- $\text{min}_\text{sup}$: minimum support threshold

Complexity: $\#P$-complete

Anti-monotonicity property: Given two itemsets, $\mathcal{I}_1$ and $\mathcal{I}_2$, if $\mathcal{I}_1 \subseteq \mathcal{I}_2$ and $\mathcal{I}_2$ is frequent, then $\mathcal{I}_1$ is frequent
Frequent Itemset Mining: Search-space

Figure: 0.6-quasi-clique
Frequent Itemsets and Quasi-cliques in Structural Correlation Pattern Mining

- We select frequent attribute sets for structural correlation pattern mining
  1. Analyzing the complete set of possible attribute sets is not feasible in real scenarios
  2. Not enough evidence about the structural correlation of infrequent attribute sets
- Quasi-cliques are considered as dense subgraphs
  - Simple and flexible definition
Structural Correlation Pattern Mining: Definitions
Attributed Graph

<table>
<thead>
<tr>
<th>vertex</th>
<th>attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A, C</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A, C, D</td>
</tr>
<tr>
<td>4</td>
<td>A, D</td>
</tr>
<tr>
<td>5</td>
<td>A, E</td>
</tr>
<tr>
<td>6</td>
<td>A, B, C</td>
</tr>
<tr>
<td>7</td>
<td>A, B, E</td>
</tr>
<tr>
<td>8</td>
<td>A, B</td>
</tr>
<tr>
<td>9</td>
<td>A, B</td>
</tr>
<tr>
<td>10</td>
<td>A, B, D</td>
</tr>
<tr>
<td>11</td>
<td>A, B</td>
</tr>
</tbody>
</table>

Vertex attributes

Graph

- **4-tuple** $G = (V, E, A, F)$:
  - $V$ is the set of vertices
  - $E$ is the set of edges
  - $A = \{a_1, a_2, \ldots, a_n\}$ is the set of vertex attributes
  - $F: V \rightarrow P(A)$ is a function that returns the set of attributes of a vertex
Attribute Set and Induced Subgraph

- We define an *attribute set* $S$ as a subset of $A$ ($S \subseteq A$)
- $\mathcal{V}(S) \subseteq \mathcal{V}$ is the vertex set induced by $S$ (i.e., $\mathcal{V}(S) = \{ v_i \in \mathcal{V} | S \subseteq \mathcal{F}(v_i) \}$)
- $\mathcal{E}(S) \subseteq \mathcal{E}$ is the edge set induced by $S$ (i.e., $\mathcal{E}(S) = \{ (v_i, v_j) \in \mathcal{E} | v_i \in \mathcal{V}(S) \land v_j \in \mathcal{V}(S) \}$)
- The graph $\mathcal{G}(S)$, induced by $S$, is the pair $(\mathcal{V}(S), \mathcal{E}(S))$
Definition: Given an attribute set $S$, let $q_1, q_2, \ldots q_k$ be the set of quasi-cliques in the graph $G(S)$. The value of $\kappa(S)$ is given by:

$$\kappa(S) = \left| \bigcup_{0<i\leq k} q_i \right|$$
Structural Coverage

\[ \kappa(\{A\}) = 9 \]

In the graph induced by \{A\} but not in a dense subgraph.

In a dense subgraph in the graph induced by \{A\}. 
Structural Coverage

$\kappa(\{A, B\}) = 6$

- Not in the graph induced by \{AB\}.
- In a dense subgraph in the graph induced by \{AB\}.
Definition: Given an attribute set $S$, the structural correlation of $S$, $\epsilon(S)$, is given by:

$$\epsilon(S) = \frac{\kappa(S)}{|\mathcal{G}(S)|}$$
Structural Correlation

\[ \epsilon(\{A\}) = 0.82 \]
Structural Coverage

$\varepsilon(\{A, B\}) = 1$

Not in the graph induced by $\{AB\}$.

In a dense subgraph in the graph induced by $\{AB\}$. 
Normalized Structural Correlation ($\epsilon_{exp}$)

- How can we evaluate a given structural correlation value?
- How can we compare the structural correlation of two attribute sets?
Normalized Structural Correlation ($\epsilon_{exp}$)

- How can we evaluate a given structural correlation value?
- How can we compare the structural correlation of two attribute sets?
- Idea: Normalizing the structural correlation based on its expected value
Null Model for the Structural Correlation

- Gives the expected structural correlation $\epsilon_{\text{exp}}(\sigma(S))$ of an attribute set $S$ with support $\sigma(S)$
- The null model considers that there is no correlation between attribute sets and dense subgraphs
- Must consider three aspects:
  1. Graph topology
  2. Attribute set support ($\sigma$)
  3. Quasi-clique parameters ($\gamma_{\text{min}}$ and $\text{min\_size}$)
Simulation Null Model

Input : $\mathcal{G}$, $\sigma$, $\gamma_{\text{min}}$, min_size, $r$
Output: $\text{sim-}\epsilon_{\text{exp}}$

\[
i \leftarrow 0;
\]

$\text{sim-}\epsilon_{\text{exp}} \leftarrow 0$;

while $i < r$ do

\[
V \leftarrow \text{random-vertices}(\mathcal{G}, \sigma);
\]

\[
n \leftarrow 0;
\]

for $v \in V$ do

\[
\text{if is-in-quasi-clique}(v, \mathcal{G}, \gamma_{\text{min}}, \text{min_size}) \text{ then}
\]

\[
n \leftarrow n + 1;
\]

\[
\text{sim-}\epsilon_{\text{exp}} \leftarrow \epsilon_{\text{exp}} + (n/\sigma);
\]

\[
i \leftarrow i + 1;
\]

\[
\text{sim-}\epsilon_{\text{exp}} \leftarrow \text{sim-}\epsilon_{\text{exp}}/r;
\]
Definition: Given an attribute set $S$ with support $\sigma(S)$, the simulation-based expected structural correlation of $S$ is given by:

$$\delta_1(S) = \frac{\epsilon(S)}{sim-\epsilon_{exp}(\sigma(S))}$$
Analytical Upper-bound of the Expected Structural Correlation ($\max-\epsilon_{exp}$)

- Given a random subgraph $G_\sigma$ of $G$ with $\sigma$ vertices, what is the probability $\epsilon_{exp}$ of a vertex from $G_\sigma$ to be part of a quasi-clique?
Analytical Upper-bound of the Expected Structural Correlation ($\text{max-} \epsilon_{\text{exp}}$)

- Given a random subgraph $G_\sigma$ of $G$ with $\sigma$ vertices, what is the probability $\epsilon_{\text{exp}}$ of a vertex from $G_\sigma$ to be part of a quasi-clique?

4 vertices selected, all in quasi-cliques 4 vertices selected, none in quasi-cliques

- 330 possible random subgraphs $G_\sigma$
Analytical Upper-bound of the Expected Structural Correlation (max-$\epsilon_{exp}$)

- Simplified problem: What is the probability max-$\epsilon_{exp}$ of finding a vertex with degree, at least, $\lceil \gamma_{min} \cdot (min\_size - 1) \rceil$ in $G_\sigma$?
Analytical Upper-bound of the Expected Structural Correlation ($\max-\epsilon_{\text{exp}}$)

- Simplified problem: What is the probability $\max-\epsilon_{\text{exp}}$ of finding a vertex with degree, at least, $\lceil \gamma_{\text{min}} \cdot (\text{min\_size} - 1) \rceil$ in $G_\sigma$?

- If a random vertex $v$ from $G$ with degree $\alpha$ is selected to be part of $G_\sigma$, the probability of such vertex to have a degree $\beta$ in $G_\sigma$ is given by the following binomial function:

$$F(\alpha, \beta, \rho) = \binom{\alpha}{\beta} \cdot \rho^\beta \cdot (1 - \rho)^{\alpha - \beta}$$

where $\rho$ is the probability of a specific vertex $u$ from $G$ to be in $G_\sigma$, if $v$ is already taken:

$$\rho = \frac{\sigma - 1}{|V - 1|}$$
Analytical Upper-bound of the Expected Structural Correlation ($\max-\epsilon_{\text{exp}}$)

- **Definition:** Given the quasi-clique parameters $\gamma_{\text{min}}$ and $\text{min\_size}$, the structural correlation of an attribute set with support $\sigma$ is upper bounded by:

$$\max-\epsilon_{\text{exp}}(\sigma, \gamma_{\text{min}}, \text{min\_size}) = \sum_{\alpha=z}^{m} p(\alpha) \cdot \sum_{\beta=z}^{\alpha} F(\alpha, \beta, \rho)$$

where $z = \lceil \gamma_{\text{min}} \cdot (\text{min\_size} - 1) \rceil$, $m$ is the maximum degree of vertices from $\mathcal{G}$, and $p$ is the degree distribution of $\mathcal{G}$.
Definition: Given an attribute set $S$ with support $\sigma(S)$, the analytical normalized structural correlation is given by:

$$\delta_2(S) = \frac{\epsilon(S)}{\max - \epsilon_{\text{exp}}(\sigma(S))}$$
Structural Correlation Patterns

- **Definition:** A structural correlation pattern is a pair \((S, V)\)
  - \(S\) is an attribute set \((S \subseteq A)\)
  - \(V\): is a quasi-clique from \(\mathcal{V}(S)\)

- **Complexity:** \#P-complete

<table>
<thead>
<tr>
<th>vertex</th>
<th>attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A, C</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A, C, D</td>
</tr>
<tr>
<td>4</td>
<td>A, D</td>
</tr>
<tr>
<td>5</td>
<td>A, E</td>
</tr>
<tr>
<td>6</td>
<td>A, B, C</td>
</tr>
<tr>
<td>7</td>
<td>A, B, E</td>
</tr>
<tr>
<td>8</td>
<td>A, B</td>
</tr>
<tr>
<td>9</td>
<td>A, B</td>
</tr>
<tr>
<td>10</td>
<td>A, B, D</td>
</tr>
<tr>
<td>11</td>
<td>A, B</td>
</tr>
</tbody>
</table>

Vertex attributes

Graph

\((\{A, B\}, \{6, 7, 8, 9, 10, 11\})\)
Structural Correlation Pattern Mining Problem

- **Definition:** Consists of identifying the set of structural correlation patterns \((S, V)\) from an attributed graph \(G(V, E, A, F)\), such that:
  - \(\epsilon(S) \geq \epsilon_{\text{min}}\)
  - \(\sigma(S) \geq \sigma_{\text{min}}\)
  - \(V\) is a \(\gamma_{\text{min}}\)-quasi-clique
  - \(V \subseteq V(S)\)
  - \(|V| \geq \text{min\_size}\)

- **Complexity:** \#P-complete
Structural Correlation Pattern Mining Problem

<table>
<thead>
<tr>
<th>vertex</th>
<th>attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A, C</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A, C, D</td>
</tr>
<tr>
<td>4</td>
<td>A, D</td>
</tr>
<tr>
<td>5</td>
<td>A, E</td>
</tr>
<tr>
<td>6</td>
<td>A, B, C</td>
</tr>
<tr>
<td>7</td>
<td>A, B, E</td>
</tr>
<tr>
<td>8</td>
<td>A, B</td>
</tr>
<tr>
<td>9</td>
<td>A, B</td>
</tr>
<tr>
<td>10</td>
<td>A, B, D</td>
</tr>
<tr>
<td>11</td>
<td>A, B</td>
</tr>
</tbody>
</table>

Vertex attributes

<table>
<thead>
<tr>
<th>attribute set</th>
<th>σ</th>
<th>κ</th>
<th>ε</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>11</td>
<td>9</td>
<td>0.82</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>6</td>
<td>1.0</td>
</tr>
<tr>
<td>A, B</td>
<td>6</td>
<td>6</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Attribute sets

<table>
<thead>
<tr>
<th>pattern</th>
<th>size</th>
<th>γ</th>
</tr>
</thead>
<tbody>
<tr>
<td>({A}, {6, 7, 8, 9, 10, 11})</td>
<td>6</td>
<td>0.60</td>
</tr>
<tr>
<td>({A}, {3, 4, 5, 6})</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>({A}, {3, 4, 6, 7})</td>
<td>4</td>
<td>0.67</td>
</tr>
<tr>
<td>({A}, {3, 5, 6, 7})</td>
<td>4</td>
<td>0.67</td>
</tr>
<tr>
<td>({A}, {3, 6, 7, 8})</td>
<td>4</td>
<td>0.67</td>
</tr>
<tr>
<td>({B}, {6, 7, 8, 9, 10, 11})</td>
<td>6</td>
<td>0.60</td>
</tr>
<tr>
<td>({A, B}, {6, 7, 8, 9, 10, 11})</td>
<td>6</td>
<td>0.60</td>
</tr>
</tbody>
</table>

Patterns

σ_{min}=3, ϵ_{min}=0.5
Structural Correlation Pattern Mining Problem

<table>
<thead>
<tr>
<th>vertex</th>
<th>attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A, C</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A, C, D</td>
</tr>
<tr>
<td>4</td>
<td>A, D</td>
</tr>
<tr>
<td>5</td>
<td>A, E</td>
</tr>
<tr>
<td>6</td>
<td>A, B, C</td>
</tr>
<tr>
<td>7</td>
<td>A, B, E</td>
</tr>
<tr>
<td>8</td>
<td>A, B</td>
</tr>
<tr>
<td>9</td>
<td>A, B</td>
</tr>
<tr>
<td>10</td>
<td>A, B, D</td>
</tr>
<tr>
<td>11</td>
<td>A, B</td>
</tr>
</tbody>
</table>

Vertex attributes

Graph

<table>
<thead>
<tr>
<th>attribute set</th>
<th>(\sigma)</th>
<th>(\kappa)</th>
<th>(\epsilon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>11</td>
<td>9</td>
<td>0.82</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>6</td>
<td>1.0</td>
</tr>
<tr>
<td>A, B</td>
<td>6</td>
<td>6</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Attribute sets

<table>
<thead>
<tr>
<th>pattern</th>
<th>size</th>
<th>(\gamma)</th>
</tr>
</thead>
<tbody>
<tr>
<td>({A}),({6, 7, 8, 9, 10, 11})</td>
<td>6</td>
<td>0.60</td>
</tr>
<tr>
<td>({A}),({3, 4, 5, 6})</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>({A}),({3, 4, 6, 7})</td>
<td>4</td>
<td>0.67</td>
</tr>
<tr>
<td>({A}),({3, 5, 6, 7})</td>
<td>4</td>
<td>0.67</td>
</tr>
<tr>
<td>({A}),({3, 6, 7, 8})</td>
<td>4</td>
<td>0.67</td>
</tr>
<tr>
<td>({B}),({6, 7, 8, 9, 10, 11})</td>
<td>6</td>
<td>0.60</td>
</tr>
<tr>
<td>({A, B}),({6, 7, 8, 9, 10, 11})</td>
<td>6</td>
<td>0.60</td>
</tr>
</tbody>
</table>

Patterns

\(\sigma_{min}=3, \epsilon_{min}=0.5\)
Definition: Consists of identifying the set of structural correlation patterns \((S, V)\) from an attributed graph \(G(V, E, A, F)\), such that:

- \(\sigma(S) \geq \sigma_{min}\)
- \(\delta(S) \geq \delta_{min}\)
- \(V\) is a \(\gamma\)-quasi-clique
- \(V \subseteq V(S)\)
- \(|V| \geq min\_size\)

Complexity: \#P-complete
Normalized Structural Correlation Pattern Mining Problem

<table>
<thead>
<tr>
<th>vertex</th>
<th>attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A, C</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A, C, D</td>
</tr>
<tr>
<td>4</td>
<td>A, D</td>
</tr>
<tr>
<td>5</td>
<td>A, E</td>
</tr>
<tr>
<td>6</td>
<td>A, B, C</td>
</tr>
<tr>
<td>7</td>
<td>A, B, E</td>
</tr>
<tr>
<td>8</td>
<td>A, B</td>
</tr>
<tr>
<td>9</td>
<td>A, B</td>
</tr>
<tr>
<td>10</td>
<td>A, B, D</td>
</tr>
<tr>
<td>11</td>
<td>A, B</td>
</tr>
</tbody>
</table>

Vertex attributes

![Graph](image.png)

<table>
<thead>
<tr>
<th>attribute set</th>
<th>$\epsilon$</th>
<th>$\epsilon_{exp}$</th>
<th>$\delta_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>1.0</td>
<td>0.59</td>
<td>1.69</td>
</tr>
<tr>
<td>A, B</td>
<td>1.0</td>
<td>0.59</td>
<td>1.69</td>
</tr>
</tbody>
</table>

Attribute sets

<table>
<thead>
<tr>
<th>pattern</th>
<th>size</th>
<th>$\gamma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${B},{6, 7, 8, 9, 10, 11}$</td>
<td>6</td>
<td>0.60</td>
</tr>
<tr>
<td>${A, B},{6, 7, 8, 9, 10, 11}$</td>
<td>6</td>
<td>0.60</td>
</tr>
</tbody>
</table>

Patterns

$\sigma_{min}=3, \epsilon_{min}=0.5, \delta_{min}=1.0$
Normalized Structural Correlation Pattern Mining Problem

Vertex attributes

<table>
<thead>
<tr>
<th>vertex</th>
<th>attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A, C</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A, C, D</td>
</tr>
<tr>
<td>4</td>
<td>A, D</td>
</tr>
<tr>
<td>5</td>
<td>A, E</td>
</tr>
<tr>
<td>6</td>
<td>A, B, C</td>
</tr>
<tr>
<td>7</td>
<td>A, B, E</td>
</tr>
<tr>
<td>8</td>
<td>A, B</td>
</tr>
<tr>
<td>9</td>
<td>A, B</td>
</tr>
<tr>
<td>10</td>
<td>A, B, D</td>
</tr>
<tr>
<td>11</td>
<td>A, B</td>
</tr>
</tbody>
</table>

Graph

<table>
<thead>
<tr>
<th>attribute set</th>
<th>$\epsilon$</th>
<th>$\epsilon_{exp}$</th>
<th>$\delta_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>1.0</td>
<td>0.59</td>
<td>1.69</td>
</tr>
<tr>
<td>A, B</td>
<td>1.0</td>
<td>0.59</td>
<td>1.69</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>pattern</th>
<th>size</th>
<th>$\gamma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${B}, {6, 7, 8, 9, 10, 11}$</td>
<td>6</td>
<td>0.60</td>
</tr>
<tr>
<td>${A, B}, {6, 7, 8, 9, 10, 11}$</td>
<td>6</td>
<td>0.60</td>
</tr>
</tbody>
</table>

$\sigma_{min}=3, \epsilon_{min}=0.5, \delta_{min}=1.0$
Structural Correlation Pattern Mining: Algorithms
The structural correlation pattern mining brings interesting computational challenges in terms of performance

- Both the standard and the normalized versions of the problem are \#P-complete

We propose different strategies for efficient structural correlation pattern mining:

1. Search
2. Pruning
3. Sampling
4. Identification of the top-k most interesting patterns
5. Parallelization

We focus on two important steps:

1. Computing the structural correlation
2. Identifying structural correlation patterns
Naive Algorithm for Structural Correlation Pattern Mining

<table>
<thead>
<tr>
<th>Input</th>
<th>( G, \sigma_{\text{min}}, \gamma_{\text{min}}, \text{min_size}, \epsilon_{\text{min}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output</td>
<td>( P )</td>
</tr>
</tbody>
</table>

\[ I \leftarrow \text{frequent-attribute-sets}(G, \sigma_{\text{min}}); \]
\[ P \leftarrow \emptyset; \]
\[ \text{for } S \in I \text{ do} \]
\[ \quad Q \leftarrow \text{quasi-cliques}(G(S), \gamma_{\text{min}}, \text{min\_size}); \]
\[ \quad \text{if } \epsilon(Q, S) \geq \epsilon_{\text{min}} \text{ then} \]
\[ \quad \quad \text{for } q \in Q \text{ do} \]
\[ \quad \quad \quad P \leftarrow P \cup (S, q); \]
Computing the Structural Correlation: Search

- The computation of $\epsilon$ depends only on how many vertices in an induced graph are in quasi-cliques
- The naive algorithm identifies the complete set of quasi-cliques from the induced graph
- How can we check whether each vertex is in a quasi-clique visiting a small number of quasi-clique candidates?
Computing the Structural Correlation: Search

- The computation of $\epsilon$ depends only on how many vertices in an induced graph are in quasi-cliques
- The naive algorithm identifies the complete set of quasi-cliques from the induced graph
- How can we check whether each vertex is in a quasi-clique visiting a small number of quasi-clique candidates?
Computing the Structural Correlation: Search

- The computation of $\epsilon$ depends only on how many vertices in an induced graph are in quasi-cliques
- The naive algorithm identifies the complete set of quasi-cliques from the induced graph
- How can we check whether each vertex is in a quasi-clique visiting a small number of quasi-clique candidates?
Computing the Structural Correlation: Search

- The computation of $\epsilon$ depends only on how many vertices in an induced graph are in quasi-cliques
- The naive algorithm identifies the complete set of quasi-cliques from the induced graph
- How can we check whether each vertex is in a quasi-clique visiting a small number of quasi-clique candidates?
structural-correlation

\[
\text{Input} : \mathcal{G}(S), \gamma_{\text{min}}, \text{min\_size}, \kappa_{\text{min}}
\]
\[
\text{Output:} \ \varepsilon
\]
\[
X \leftarrow \emptyset;
\]
\[
candExts(X) \leftarrow \text{vertex-pruning}(\mathcal{V}(S), \gamma_{\text{min}}, \text{min\_size});
\]
\[
\text{if } |candExts(X)| \leq \kappa_{\text{min}} \text{ then}
\]
\[
\quad \text{return -1;}
\]
\[
\text{if } \text{searchStrategy} = \text{BFS} \text{ then}
\]
\[
\quad K \leftarrow \text{coverage-BFS}(X, candExts(X), \mathcal{G}(S), \gamma_{\text{min}}, \text{min\_size});
\]
\[\text{else}
\]
\[\quad \text{if } \text{searchStrategy} = \text{DFS} \text{ then}
\]
\[\quad \quad K \leftarrow \text{coverage-DFS}(X, candExts(X), \mathcal{G}(S), \gamma_{\text{min}}, \text{min\_size});
\]
\[
\epsilon \leftarrow |K|/|\mathcal{G}(S)|;
\]
\[
\text{return } \epsilon;
\]
coverage-BFS

Input: $X$, candExts($X$), $G(S)$, $\gamma_{min}$, min_size
Output: $K$

$qcCands \leftarrow \emptyset$;
$qcCands$.enqueue(($X$, candExts($X$)));
$K \leftarrow \emptyset$;

while $qcCands \neq \emptyset$ do

$q \leftarrow qcCands$.dequeue();

if candidate-quasi-clique-pruning($q.X$, $q$.candExts($X$), $G(S)$, $\gamma_{min}$, min_size) = FALSE then

if is-quasi-clique($q.X \cup q$.candExts($X$)) then

for $v \in q.X \cup q$.candExts($X$) do

$K \leftarrow K \cup v$;

endif

if is-quasi-clique($q.X$) then

for $v \in q.X$ do

$K \leftarrow K \cup v$;

endif

for $v \in q$.candExts($X$) do

$t$.candExts($X$) $\leftarrow \{u \in q$.candExts($X$)$|u > v\}$;
$t.X \leftarrow q.X \cup v$;

if $t.X \cup t$.candExts($X$) $\not\subseteq K$ then

qcCands.enqueue($t$);

endif

endif

endif

endif
Pruning

- Idea: Reducing the search space of candidate quasi-cliques and attribute sets
Pruning

- Idea: Reducing the search space of candidate quasi-cliques and attribute sets
- **Vertex pruning:** Let $K_{S_i}$ be the coverage of an attribute set $S_i$ and $K_{S_j}$ be the structural coverage of an attribute set $S_j$, if $S_i \subseteq S_j$, then $K_{S_j} \subseteq K_{S_i}$

\[ G(A) \]

\[ G(A, B) \]
Pruning

- Attribute set pruning based on the upper bound of $\epsilon$: For two attribute sets $S_i$ and $S_j$, if $S_i \subseteq S_j$ and $\sigma(S_j) \geq \sigma_{min}$, then $\epsilon(S_j) \leq \epsilon(S_i) \cdot \nu(S_i) / \sigma_{min}$.

\[ \epsilon(A) = 0.82 \]

If $\sigma_{min} = 3$, $\epsilon(A, B) \leq \frac{0.82 \cdot 11}{3} = 3$
Pruning

- Attribute set pruning based on the upper bound of $\epsilon$: For two attribute sets $S_i$ and $S_j$, if $S_i \subseteq S_j$ and $\sigma(S_j) \geq \sigma_{min}$, then $\epsilon(S_j) \leq \epsilon(S_i).|\mathcal{V}(S_i)|/\sigma_{min}$

\[ \epsilon(A) = 0.82 \]

\[ \text{If } \sigma_{min}=3, \epsilon(A, B) \leq \frac{0.82 \cdot 11}{3} = 3 \]

- Similar pruning based on $\delta$ (omitted)
Sampling

- Idea: Computing the structural correlation of $S$ considering a sample of vertices from $\mathcal{V}(S)$
Sampling

- Idea: Computing the structural correlation of $S$ considering a sample of vertices from $\mathcal{V}(S)$
- The structural correlation $\epsilon$ is estimated using a sample $Z \subseteq \mathcal{V}(S)$
- The error can also be estimated:

$$\text{error} = z_{\alpha/2} \sqrt{\frac{\epsilon(1 - \epsilon)(|\mathcal{V}(S)| - |Z|)}{|Z|(|\mathcal{V}(S)| - 1)}}$$

- The size of $Z$ can be defined based on the maximum error accepted $\theta_{\text{max}}$
- Algorithms for computing $\epsilon$ using sampling are similar to those that compute the exact $\epsilon$ (omitted)
Top-k Structural Correlation Patterns

- Idea: Identifying only the top-k structural correlation patterns instead of the complete set of patterns

Motivation:
- Discovering quasi-cliques is an expensive task
- The largest and densest patterns are the most interesting
- How can we identify the largest and densest patterns traversing a small fraction of the search space?
Top-k Structural Correlation Patterns

- Idea: Identifying only the top-k structural correlation patterns instead of the complete set of patterns
- Motivation:
  - Discovering quasi-cliques is an expensive task
  - The largest and densest patterns are the most interesting
Top-k Structural Correlation Patterns

- Idea: Identifying only the top-k structural correlation patterns instead of the complete set of patterns
- Motivation:
  - Discovering quasi-cliques is an expensive task
  - The largest and densest patterns are the most interesting
- How can we identify the largest and densest patterns traversing a small fraction of the search space?
Top-k Structural Correlation Patterns

- We use a current set $C$ of top-k structural correlation patterns to prune the search space of patterns.
- For a pair $(X, \text{candExts}(X))$, if $|X \cup \text{candExts}(X)|$ is smaller than the smallest and sparsest pattern in $C$, it can be pruned.
Input : $G(S), k, \gamma_{min}, \text{min\_size}$
Output: $C$

$C \leftarrow \emptyset; \text{qcCands} \leftarrow \emptyset; q.X \leftarrow \emptyset;$
$q.candExts(X) \leftarrow \text{vertex-pruning}(\forall(S), \gamma_{min}, \text{min\_size});$
$qcCands.push(q);$ $\text{while \hspace{0.5em} qcCands \neq \emptyset \hspace{0.5em}}\text{do}\
q \leftarrow qcCands.pop();$
$\text{if \hspace{0.5em} }|q.X| + |q.candExts(X)| \geq \text{min\_size} \hspace{0.5em} \text{then}\
\text{if \hspace{0.5em} candidate-quasi-clique-pruning}(q.X, q.candExts(X), G(S), \gamma_{min}, \text{min\_size}) = \text{FALSE} \hspace{0.5em} \text{then}\
\text{if \hspace{0.5em} }|q.X| + |q.candExts(X)| \geq \text{min\_size} \hspace{0.5em} \text{then}\
\text{if \hspace{0.5em} is-quasi-clique}(q.X \cup q.candExts(X), \gamma_{min}, \text{min\_size} \text{ then}\
\text{min\_size} \leftarrow \text{try-to-update-top-patterns}(q.X \cup q.candExts(X), C, \text{min\_size});$
\text{else}\
\text{if \hspace{0.5em} is-quasi-clique}(q.X) \hspace{0.5em} \text{then}\
\text{min\_size} \leftarrow \text{try-to-update-top-patterns}(q.X, C, \text{min\_size});$
\text{for } v \in q.candExts(X) \hspace{0.5em} \text{do}\
\text{t.candExts}(X) \leftarrow \{ u \in q.candExts(X) \mid u > v \};$
\text{t.X} \leftarrow q.X \cup v;$
$qcCands.push(t);$
Parallel Algorithms

- How can we compute the structural correlation of attribute sets and identify top-k structural correlation patterns exploiting multiple processing units?
Parallel Algorithms

- How can we compute the structural correlation of attribute sets and identify top-k structural correlation patterns exploiting multiple processing units?
- General solution: Work pool pattern
  - Tasks are inserted into a work pool and threads actively get new tasks whenever they are idle
- Task: processing a pair \((X, \text{candExts}(X))\)
- In the proposed algorithms, by processing one task, new tasks can be produced
- Threads have their own local pools and share work whenever the global pool is empty and there is an idle thread
par-coverage-DFS

**Input:** globalQCCands, K, G(S), γ_{min}, min_size, numActiveThreads, numThreads

while TRUE do
  if |qcCands| > 0 then
    q ← globalQCCands.pop(); localQCCands.push(q);
    numActiveThreads ← numActiveThreads + 1;
  else
    if numActiveThreads = 0 then
      BREAK;

  if |localQCCands| > 0 then
    while TRUE do
      q ← localQCCands.pop(); newQCCands ← ∅;
      par-coverage(q.X, q.candExts(X), K, G(S), γ_{min}, min_size, newQCCands);
      localQCCands.push(newQCCands);
      if |localQCCands| = 0 then
        numActiveThreads ← numActiveThreads − 1;
        BREAK;
      if numActiveThreads < numThreads AND |globalQCCands| = 0 then
        globalQCCands.push(localQCCands); localQCCands ← ∅;
        numActiveThreads ← numActiveThreads − 1;
For computing the structural correlation:
  - Using BFS
  - Using sampling + DFS
  - Using sampling + BFS

For the identification of the top-k structural correlation patterns
The SCPM Algorithm

- Solves the structural and normalized structural correlation pattern mining problems
- Can restrict the structural correlation patterns identified to the top-k in terms of size and density
- Applies the search, pruning, sampling and parallelization techniques proposed by this work
The SCPM Algorithm

Input: $G$, $\sigma_{\min}$, $\gamma_{\min}$, min_size, $\epsilon_{\min}$, $\delta_{\min}$, $k$, $\theta_{\max}$

Output: $P$

$P \leftarrow \emptyset$; $T \leftarrow \emptyset$; $I \leftarrow \text{frequent-attributes}(G, \sigma_{\min})$;

for $S \in I$ do

if $\theta_{\max} > 0$ then

if $\epsilon_{\min} > \delta_{\min} \cdot \epsilon_{\exp}(\sigma(S))$ then

$\epsilon \leftarrow \text{str-corr-sampling}(S, G(S), \gamma_{\min}, \text{min_size}, \epsilon_{\min} \cdot \sigma(S), \theta_{\max})$;

else

$\epsilon \leftarrow \text{str-corr-sampling}(S, G(S), \gamma_{\min}, \text{min_size}, \delta_{\min} \cdot \epsilon_{\exp}(S) \cdot \sigma(S), \theta_{\max})$;

else

if $\epsilon_{\min} > \delta_{\min} \cdot \epsilon_{\exp}(\sigma(S))$ then

$\epsilon \leftarrow \text{structural-correlation}(G(S), \gamma_{\min}, \text{min_size}, \epsilon_{\min} \cdot \sigma(S))$;

else

$\epsilon \leftarrow \text{structural-correlation}(G(S), \gamma_{\min}, \text{min_size}, \delta_{\min} \cdot \epsilon_{\exp}(S) \cdot \sigma(S))$;

endif

else

if $\epsilon_{\min} \geq \epsilon_{\min}$ AND $\epsilon / \epsilon_{\exp}(S) \geq \delta_{\min}$ then

$Q \leftarrow \text{top-k-structural-correlation-patterns}(G(S), k, \gamma_{\min}, \text{min_size})$;

for $q \in Q$ do

$P \leftarrow P \cup (S, q)$;

endif

if $\epsilon \geq 0$ AND $\epsilon \cdot \sigma(S) \geq \epsilon_{\min} \cdot \sigma_{\min}$ AND $\epsilon \cdot \sigma(S) \geq \delta_{\min} \cdot \epsilon_{\exp}(\sigma_{\min}) \cdot \sigma_{\min}$ then

$T \leftarrow T \cup S$;

endif

$P \leftarrow P \cup \text{enumerate-patterns}(T, G, \sigma_{\min}, \gamma_{\min}, \epsilon_{\min}, \delta_{\min}, k, \theta_{\max})$;
Experimental Evaluation
Experimental Evaluation

- Case studies:
  - Correlation between attribute sets and dense subgraphs in real datasets from different domains
  - The objective is to show the applicability of the knowledge provided by the structural correlation pattern mining in real scenarios

- Performance Evaluation:
  - Evaluation of the proposed algorithms in terms of execution time
  - The objective is to show that the proposed algorithms enable the structural correlation pattern mining in large graphs
## Datasets

<table>
<thead>
<tr>
<th>name</th>
<th>vertex</th>
<th>edge</th>
<th>attribute</th>
<th>attr. set</th>
<th>subgraph</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBLP</td>
<td>author</td>
<td>co-authorship</td>
<td>term</td>
<td>topic</td>
<td>community</td>
</tr>
<tr>
<td>LastFm</td>
<td>user</td>
<td>friendship</td>
<td>artist</td>
<td>taste</td>
<td>community</td>
</tr>
<tr>
<td>CiteseerX</td>
<td>paper</td>
<td>citation</td>
<td>term</td>
<td>topic</td>
<td>related work</td>
</tr>
<tr>
<td>Human</td>
<td>gene</td>
<td>gene interaction</td>
<td>tissue</td>
<td>tissue set</td>
<td>module</td>
</tr>
</tbody>
</table>

### Table: Descriptions

<table>
<thead>
<tr>
<th>name</th>
<th>#vertices</th>
<th>#edges</th>
<th>#attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBLP</td>
<td>108,030</td>
<td>276,658</td>
<td>23,285</td>
</tr>
<tr>
<td>LastFm</td>
<td>272,412</td>
<td>350,239</td>
<td>3,929,101</td>
</tr>
<tr>
<td>CiteseerX</td>
<td>294,104</td>
<td>782,147</td>
<td>206,430</td>
</tr>
<tr>
<td>Human</td>
<td>3,628</td>
<td>8,924</td>
<td>230</td>
</tr>
</tbody>
</table>

### Table: Statistics
Top-$\epsilon$ attribute sets: DBLP

<table>
<thead>
<tr>
<th>$S$</th>
<th>size</th>
<th>$\sigma$</th>
<th>$\kappa$</th>
<th>$\epsilon$</th>
</tr>
</thead>
<tbody>
<tr>
<td>grid applic</td>
<td>2</td>
<td>840</td>
<td>222</td>
<td>0.26</td>
</tr>
<tr>
<td>grid servic</td>
<td>2</td>
<td>599</td>
<td>138</td>
<td>0.23</td>
</tr>
<tr>
<td>environ grid</td>
<td>2</td>
<td>525</td>
<td>113</td>
<td>0.21</td>
</tr>
<tr>
<td>queri xml</td>
<td>2</td>
<td>615</td>
<td>127</td>
<td>0.21</td>
</tr>
<tr>
<td>search web</td>
<td>2</td>
<td>1031</td>
<td>209</td>
<td>0.20</td>
</tr>
<tr>
<td>search rank</td>
<td>2</td>
<td>420</td>
<td>81</td>
<td>0.19</td>
</tr>
<tr>
<td>dynam simul</td>
<td>2</td>
<td>469</td>
<td>90</td>
<td>0.19</td>
</tr>
<tr>
<td>queri data</td>
<td>2</td>
<td>1540</td>
<td>295</td>
<td>0.19</td>
</tr>
<tr>
<td>chip system</td>
<td>2</td>
<td>702</td>
<td>132</td>
<td>0.19</td>
</tr>
<tr>
<td>data stream</td>
<td>2</td>
<td>1073</td>
<td>198</td>
<td>0.18</td>
</tr>
</tbody>
</table>

$\gamma_{min}=0.5, \ min\_size=10, \ \sigma_{min}=400$
Top-$\epsilon$ attribute sets: LastFm

<table>
<thead>
<tr>
<th>$S$</th>
<th>size</th>
<th>$\sigma$</th>
<th>$\kappa$</th>
<th>$\epsilon$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radiohead</td>
<td>1</td>
<td>121,892</td>
<td>13,362</td>
<td>0.11</td>
</tr>
<tr>
<td>Coldplay</td>
<td>1</td>
<td>118,053</td>
<td>11,116</td>
<td>0.09</td>
</tr>
<tr>
<td>Beatles</td>
<td>1</td>
<td>109,037</td>
<td>10,228</td>
<td>0.09</td>
</tr>
<tr>
<td>Red Hot Chili Peppers</td>
<td>1</td>
<td>105,984</td>
<td>9,217</td>
<td>0.09</td>
</tr>
<tr>
<td>Metallica</td>
<td>1</td>
<td>83,587</td>
<td>6,405</td>
<td>0.08</td>
</tr>
<tr>
<td>Death Cap for Cutie</td>
<td>1</td>
<td>82,025</td>
<td>6,065</td>
<td>0.07</td>
</tr>
<tr>
<td>Beck</td>
<td>1</td>
<td>83,360</td>
<td>6103</td>
<td>0.07</td>
</tr>
<tr>
<td>Muse</td>
<td>1</td>
<td>94,382</td>
<td>6,792</td>
<td>0.07</td>
</tr>
<tr>
<td>Nirvana</td>
<td>1</td>
<td>100,604</td>
<td>7192</td>
<td>0.07</td>
</tr>
<tr>
<td>The Shins</td>
<td>1</td>
<td>68,480</td>
<td>4874</td>
<td>0.07</td>
</tr>
</tbody>
</table>

$\gamma_{min}=0.5$, $min\_size=5$, $\sigma_{min}=27,000$
Top-\(\epsilon\) attribute sets: Citeseer

<table>
<thead>
<tr>
<th>(S)</th>
<th>size</th>
<th>(\sigma)</th>
<th>(\kappa)</th>
<th>(\epsilon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>network sensor</td>
<td>2</td>
<td>3276</td>
<td>1545</td>
<td>0.47</td>
</tr>
<tr>
<td>network hoc</td>
<td>2</td>
<td>2744</td>
<td>1279</td>
<td>0.47</td>
</tr>
<tr>
<td>ad network hoc</td>
<td>3</td>
<td>2725</td>
<td>1198</td>
<td>0.44</td>
</tr>
<tr>
<td>network rout</td>
<td>2</td>
<td>5084</td>
<td>2063</td>
<td>0.41</td>
</tr>
<tr>
<td>network wireless</td>
<td>2</td>
<td>5242</td>
<td>2073</td>
<td>0.40</td>
</tr>
<tr>
<td>node wireless</td>
<td>2</td>
<td>2086</td>
<td>737</td>
<td>0.35</td>
</tr>
<tr>
<td>protocol rout</td>
<td>2</td>
<td>2134</td>
<td>749</td>
<td>0.35</td>
</tr>
<tr>
<td>ad network</td>
<td>2</td>
<td>3563</td>
<td>1218</td>
<td>0.34</td>
</tr>
<tr>
<td>program logic</td>
<td>2</td>
<td>5895</td>
<td>1939</td>
<td>0.33</td>
</tr>
<tr>
<td>memori cach</td>
<td>2</td>
<td>2150</td>
<td>697</td>
<td>0.32</td>
</tr>
</tbody>
</table>

\(\gamma_{min}=0.5, \ min\_size=5, \ \sigma_{min}=2,000\)
### Top-$\epsilon$ attribute sets: Human

<table>
<thead>
<tr>
<th>$S$</th>
<th>size</th>
<th>$\sigma$</th>
<th>$\kappa$</th>
<th>$\epsilon$</th>
</tr>
</thead>
<tbody>
<tr>
<td>+SHCN086 +SHCN087 +SHCN088</td>
<td>3</td>
<td>328</td>
<td>58</td>
<td>0.18</td>
</tr>
<tr>
<td>+SHCN088</td>
<td>1</td>
<td>910</td>
<td>152</td>
<td>0.17</td>
</tr>
<tr>
<td>+SHCN078 +SHCN080 +SHCN088</td>
<td>3</td>
<td>1328</td>
<td>54</td>
<td>0.16</td>
</tr>
<tr>
<td>+SHCN060 +SHCN078</td>
<td>2</td>
<td>303</td>
<td>48</td>
<td>0.16</td>
</tr>
<tr>
<td>+SHCN087 +SHCN088</td>
<td>2</td>
<td>523</td>
<td>81</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHCN080 +SHCN088</td>
<td>2</td>
<td>433</td>
<td>67</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHCN078 +SHCN087 +SHCN088</td>
<td>3</td>
<td>316</td>
<td>48</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHCN080 +SHCN087 +SHCN088</td>
<td>3</td>
<td>343</td>
<td>52</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHBW148</td>
<td>1</td>
<td>864</td>
<td>130</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHCN060 +SHCN086</td>
<td>2</td>
<td>326</td>
<td>49</td>
<td>0.15</td>
</tr>
</tbody>
</table>

$\gamma_{min}=0.5$, $min\_size=5$, $\sigma_{min}=300$
### Top-$\epsilon$ attribute sets: Human

<table>
<thead>
<tr>
<th>$S$</th>
<th>size</th>
<th>$\sigma$</th>
<th>$\kappa$</th>
<th>$\epsilon$</th>
</tr>
</thead>
<tbody>
<tr>
<td>+SHCN086 +SHCN087 +SHCN088</td>
<td>3</td>
<td>328</td>
<td>58</td>
<td>0.18</td>
</tr>
<tr>
<td>+SHCN088</td>
<td>1</td>
<td>910</td>
<td>152</td>
<td>0.17</td>
</tr>
<tr>
<td>+SHCN078 +SHCN080 +SHCN088</td>
<td>3</td>
<td>1328</td>
<td>54</td>
<td>0.16</td>
</tr>
<tr>
<td>+SHCN060 +SHCN078</td>
<td>2</td>
<td>303</td>
<td>48</td>
<td>0.16</td>
</tr>
<tr>
<td>+SHCN087 +SHCN088</td>
<td>2</td>
<td>523</td>
<td>81</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHCN080 +SHCN088</td>
<td>2</td>
<td>433</td>
<td>67</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHCN078 +SHCN087 +SHCN088</td>
<td>3</td>
<td>316</td>
<td>48</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHCN080 +SHCN087 +SHCN088</td>
<td>3</td>
<td>343</td>
<td>52</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHBW148</td>
<td>1</td>
<td>864</td>
<td>130</td>
<td>0.15</td>
</tr>
<tr>
<td>+SHCN060 +SHCN086</td>
<td>2</td>
<td>326</td>
<td>49</td>
<td>0.15</td>
</tr>
</tbody>
</table>

$\gamma_{\min}=0.5, \min\_\text{size}=5, \sigma_{\min}=300$

- All tissues that are part of the top-$\epsilon$ attribute sets are related to the lymphatic system (immune functions)
Structural Correlation: Discussion

- In DBLP and Citeseer the top-$\epsilon$ attribute sets may be associated to topics in Computer Science
- In LastFm, $\epsilon$ seems to be biased towards the most frequent bands
- The top-$\epsilon$ attribute sets in the Human dataset are related to the immune system
Normalization: DBLP

Figure: simulation × analytical expected $\epsilon$

$\varepsilon_{\text{exp}} \times 10^{-4}$

$\sigma$

$\sim - \varepsilon_{\text{exp}}$

$0.1 \cdot \text{max} - \varepsilon_{\text{exp}}$
Normalization: LastFm

Figure: simulation × analytical expected $\epsilon$
Normalization: Citeseer

Figure: simulation $\times$ analytical expected $\epsilon$

\[ \sigma \quad 0.11 \text{ max} - \varepsilon_{\text{exp}} \]
Normalization: Human

Figure: simulation × analytical expected $\epsilon$
Top-$\delta_2$ attribute sets: DBLP

<table>
<thead>
<tr>
<th>$S$</th>
<th>size</th>
<th>$\sigma$</th>
<th>$\kappa$</th>
<th>$\epsilon$</th>
<th>$\delta_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>search rank</td>
<td>2</td>
<td>420</td>
<td>81</td>
<td>0.19</td>
<td>635,349</td>
</tr>
<tr>
<td>perform file</td>
<td>2</td>
<td>404</td>
<td>57</td>
<td>0.14</td>
<td>555,067</td>
</tr>
<tr>
<td>structur index</td>
<td>2</td>
<td>404</td>
<td>57</td>
<td>0.14</td>
<td>555,067</td>
</tr>
<tr>
<td>search mine</td>
<td>2</td>
<td>413</td>
<td>57</td>
<td>0.14</td>
<td>490,932</td>
</tr>
<tr>
<td>us xml</td>
<td>2</td>
<td>400</td>
<td>43</td>
<td>0.11</td>
<td>442,638</td>
</tr>
<tr>
<td>search web data</td>
<td>3</td>
<td>424</td>
<td>58</td>
<td>0.14</td>
<td>431,589</td>
</tr>
<tr>
<td>base search analysi</td>
<td>3</td>
<td>414</td>
<td>49</td>
<td>0.12</td>
<td>416,385</td>
</tr>
<tr>
<td>model internet</td>
<td>2</td>
<td>401</td>
<td>40</td>
<td>0.10</td>
<td>406,059</td>
</tr>
<tr>
<td>process data databas</td>
<td>3</td>
<td>416</td>
<td>49</td>
<td>0.12</td>
<td>405,363</td>
</tr>
<tr>
<td>perform distribut parallel</td>
<td>3</td>
<td>416</td>
<td>47</td>
<td>0.11</td>
<td>388,818</td>
</tr>
</tbody>
</table>

$\gamma_{min}=0.5$, $min\_size=10$, $\sigma_{min}=400$
Top-$\delta_2$ attribute sets: LastFm

<table>
<thead>
<tr>
<th>$S$</th>
<th>size</th>
<th>$\sigma$</th>
<th>$\kappa$</th>
<th>$\epsilon$</th>
<th>$\delta_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sufjan Stevens, Wilco</td>
<td>2</td>
<td>28,798</td>
<td>1,224</td>
<td>0.04</td>
<td>1.14</td>
</tr>
<tr>
<td>Sufjan Stevens, Of Montreal</td>
<td>2</td>
<td>28,621</td>
<td>1,188</td>
<td>0.04</td>
<td>1.13</td>
</tr>
<tr>
<td>Beirut</td>
<td>1</td>
<td>27,605</td>
<td>1,067</td>
<td>0.04</td>
<td>1.11</td>
</tr>
<tr>
<td>Sufjan Stevens, Decemberists, Beatles</td>
<td>3</td>
<td>27,415</td>
<td>1,046</td>
<td>0.04</td>
<td>1.11</td>
</tr>
<tr>
<td>Neutral Milk Hotel, Sufjan Stevens</td>
<td>2</td>
<td>29,260</td>
<td>1,231</td>
<td>0.04</td>
<td>1.10</td>
</tr>
<tr>
<td>Sufjan Stevens, Flaming Lips, Beatles</td>
<td>3</td>
<td>27,571</td>
<td>1,048</td>
<td>0.04</td>
<td>1.09</td>
</tr>
<tr>
<td>Animal Collective</td>
<td>1</td>
<td>33,555</td>
<td>1,757</td>
<td>0.05</td>
<td>1.09</td>
</tr>
<tr>
<td>Broken Social Scene, Neutral Milk Hotel</td>
<td>2</td>
<td>27,308</td>
<td>1,014</td>
<td>0.04</td>
<td>1.09</td>
</tr>
<tr>
<td>Radiohead, Spoon, Sufjan Stevens</td>
<td>3</td>
<td>27,113</td>
<td>976</td>
<td>0.04</td>
<td>1.06</td>
</tr>
<tr>
<td>Neutral Milk Hotel, Radiohead, Beatles</td>
<td>3</td>
<td>28,776</td>
<td>1,126</td>
<td>0.04</td>
<td>1.04</td>
</tr>
</tbody>
</table>

$\gamma_{min}=0.5$, $min\_size=5$, $\sigma_{min}=27,000$
Top-$\delta_2$ attribute sets: Citeseer

<table>
<thead>
<tr>
<th>$S$</th>
<th>size</th>
<th>$\sigma$</th>
<th>$\kappa$</th>
<th>$\epsilon$</th>
<th>$\delta_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>node wireless</td>
<td>2</td>
<td>2086</td>
<td>737</td>
<td>0.35</td>
<td>164.40</td>
</tr>
<tr>
<td>protocol rout</td>
<td>2</td>
<td>2134</td>
<td>749</td>
<td>0.35</td>
<td>157.57</td>
</tr>
<tr>
<td>memori cach</td>
<td>2</td>
<td>2150</td>
<td>697</td>
<td>0.32</td>
<td>143.83</td>
</tr>
<tr>
<td>network hoc</td>
<td>2</td>
<td>2744</td>
<td>1279</td>
<td>0.47</td>
<td>141.22</td>
</tr>
<tr>
<td>protocol wireless</td>
<td>2</td>
<td>2048</td>
<td>593</td>
<td>0.29</td>
<td>138.70</td>
</tr>
<tr>
<td>ad network hoc</td>
<td>3</td>
<td>2725</td>
<td>1198</td>
<td>0.44</td>
<td>134.65</td>
</tr>
<tr>
<td>network node rout</td>
<td>3</td>
<td>2075</td>
<td>523</td>
<td>0.25</td>
<td>118.26</td>
</tr>
<tr>
<td>optim queri</td>
<td>2</td>
<td>2094</td>
<td>535</td>
<td>0.26</td>
<td>118.17</td>
</tr>
<tr>
<td>perform instruct</td>
<td>2</td>
<td>2111</td>
<td>536</td>
<td>0.25</td>
<td>115.95</td>
</tr>
<tr>
<td>paper ad network</td>
<td>3</td>
<td>2081</td>
<td>485</td>
<td>0.23</td>
<td>108.86</td>
</tr>
</tbody>
</table>

$\gamma_{min}=0.5$, $min\_size=5$, $\sigma_{min}=2,000$
Top-$\delta_2$ attribute sets: Human

<table>
<thead>
<tr>
<th>$S$</th>
<th>size</th>
<th>$\sigma$</th>
<th>$\kappa$</th>
<th>$\epsilon$</th>
<th>$\delta_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>+SHCN086 +SHCN087 +SHCN088</td>
<td>3</td>
<td>328</td>
<td>58</td>
<td>0.18</td>
<td>2.09</td>
</tr>
<tr>
<td>+SHCN060 +SHCN078</td>
<td>2</td>
<td>303</td>
<td>48</td>
<td>0.16</td>
<td>2.08</td>
</tr>
<tr>
<td>+SHCN078 +SHCN080 +SHCN088</td>
<td>3</td>
<td>1328</td>
<td>54</td>
<td>0.16</td>
<td>1.96</td>
</tr>
<tr>
<td>+SHCN078 +SHCN087 +SHCN088</td>
<td>3</td>
<td>316</td>
<td>48</td>
<td>0.15</td>
<td>1.90</td>
</tr>
<tr>
<td>+SHCN068 +SHCN078</td>
<td>2</td>
<td>303</td>
<td>43</td>
<td>0.14</td>
<td>1.86</td>
</tr>
<tr>
<td>+SHBW148 +SHCN087 +SHCN088</td>
<td>3</td>
<td>310</td>
<td>45</td>
<td>0.15</td>
<td>1.85</td>
</tr>
<tr>
<td>+SHCN060 +SHCN073</td>
<td>2</td>
<td>302</td>
<td>42</td>
<td>0.14</td>
<td>1.84</td>
</tr>
<tr>
<td>+SHCN078 +SHCN086 +SHCN087</td>
<td>3</td>
<td>312</td>
<td>45</td>
<td>0.14</td>
<td>1.83</td>
</tr>
<tr>
<td>+SHCN60 +SHCN86</td>
<td>2</td>
<td>326</td>
<td>49</td>
<td>0.15</td>
<td>1.80</td>
</tr>
<tr>
<td>+SHCN080 +SHCN086 +SHCN088</td>
<td>3</td>
<td>329</td>
<td>49</td>
<td>0.15</td>
<td>1.76</td>
</tr>
</tbody>
</table>

$\gamma_{min} = 0.5$, $\text{min}_\text{size} = 5$, $\sigma_{min} = 300$
The upper bound of the expected structural correlation presents a behavior similar to the simulation results.

The top-$\delta_2$ attribute sets have, in general:

- Structural correlation values higher than expected
- Intermediate or low support
Induced Graphs and Structural Correlation Patterns: DBLP

Figure: $G(\{\text{search, rank}\}), \sigma=420, \kappa=81, \epsilon=0.19, \delta_2=635,349$
Figure: Pattern induced by \( \{ \text{perform, system} \} \), size=37, \( \gamma=0.5 \)
Induced Graphs and Structural Correlation Patterns: DBLP

Figure: Pattern induced by \{perform, system\}, size=37, \(\gamma=0.5\)

Center for Supercomputing Research and Development (UIUC)
Common Runtime Support for High Performance Parallel Languages
Induced Graphs and Structural Correlation Patterns: LastFm

Figure: $G(\{\text{Sufjan Stevens, Wilco}\}), \sigma=28,798, \kappa=1,224, \epsilon=0.04, \delta_2=1.14$
Induced Graphs and Structural Correlation Patterns: LastFm

Figure: Pattern induced by \{Van Morrison\}, size=34, \gamma=0.52
Induced Graphs and Structural Correlation Patterns: Citeseer

Figure: $G(\{node, wireless\})$, $\sigma=2,086$, $\kappa=737$, $\epsilon=0.35$, $\delta_2=164.40$
Induced Graphs and Structural Correlation Patterns: Citeseer

Figure: Pattern induced by \( \{ \text{perform, system}\} \), size=21, \( \gamma=0.5 \)
Induced Graphs and Structural Correlation Patterns: Human

Figure: $\mathcal{G}(\{+SHCN086, +SHCN087, +SHCN088\})$, $\sigma=328$, $\kappa=58$, $\epsilon=0.18$, $\delta_2=2.09$
Induced Graphs and Structural Correlation Patterns: Human

Figure: Pattern induced by \{+SHCN086, +SHCN087, +SHCN088\}, size=7, $\gamma=0.5$

Genes involved in the immune response process according to the GO
Induced Graphs and Structural Correlation Patterns: Discussion

- Vertices covered by structural correlation patterns are part of dense cores of the induced graphs.
- Such vertices are able to capture the density of the graph induced by an attribute set.
- Relatively large structural correlation patterns have been found in real networks analyzed.
- The attribute sets carry interesting semantic value regarding the patterns discovered.
Performance Evaluation

<table>
<thead>
<tr>
<th>name</th>
<th>#vertices</th>
<th>#edges</th>
<th>#attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBLP</td>
<td>108,030</td>
<td>276,658</td>
<td>23,285</td>
</tr>
<tr>
<td>SmallDBLP</td>
<td>32,908</td>
<td>82,376</td>
<td>11,192</td>
</tr>
</tbody>
</table>

Table: Dataset statistics

- Machine: 16-core Intel Xeon 2.4 Ghz with 50GB of RAM
- Each result is the average of 5 executions
Computing the Structural Correlation

- **Algorithms:**
  - **Naive:** Identifies the complete set of structural correlation patterns
  - **SCPM-BFS:** Computes the structural correlation using BFS
  - **SCPM-DFS:** Computes the structural correlation using DFS

- **Dataset:** SmallDBLP

- **Parameters evaluated:** $\gamma_{min}$, $min_{size}$, $\sigma_{min}$, $\epsilon_{min}$, and $\delta_{min}$
Quasi-clique parameters
Attribute set parameters

![Graphs showing runtime vs. \( \sigma_{\text{min}} \), \( \epsilon_{\text{min}} \), and \( \delta_{\text{min}} \)]
Computing the Structural Correlation: Discussion

- In general, the SCPM-DFS achieves the best results.
- High impact of the quasi-clique parameters over the execution time of the algorithms.
- In some settings, the Naive algorithm performs better than the SCPM-BFS.
- The SCPM-DFS and SCPM-BFS algorithms are able to take advantage of the $\epsilon_{min}$ and $\delta_{min}$ parameters due to the pruning techniques.
Computing the Structural Correlation using Sampling

- **Algorithms:**
  - **Naive:** Identifies the complete set of structural correlation patterns
  - **SCPM-BFS:** Computes the structural correlation using BFS
  - **SCPM-DFS:** Computes the structural correlation using DFS
  - **SCPM-SAMP-BFS:** Computes the structural correlation using BFS and sampling
  - **SCPM-SAMP-DFS:** Computes the structural correlation using DFS and sampling

- **Dataset:** SmallDBLP
- **Parameter evaluated:** $\theta_{max}$
Computing the Structural Correlation using Sampling

\[ \theta_{\text{min}} \times \text{runtime} \]

\[ \theta_{\text{min}} \times \text{error} \]
Computing the Structural Correlation using Sampling: Discussion

- Sampling leads to significative performance improvements
- The SCPM-SAMP-DFS and SCPM-SAMP-BFS algorithms achieved similar results in terms of execution time
- The average error of the estimates of the structural correlation is regulated by the $\theta_{min}$ parameter
Discovering the Top-k Structural Correlation Patterns

- **Algorithms:**
  - **Naive:** Identifies the complete set of structural correlation patterns
  - **SCPM-DFS:** Computes the structural correlation using DFS and identifies the top-k patterns

- **Dataset:** SmallDBLP
- **Parameter evaluated:** $k$
Discovering the Top-k Structural Correlation Patterns

![Graph showing runtime vs k for Naive and SCPM-DFS methods.](image)
Discovering the Top-k Structural Correlation Patterns: Discussion

- Decreasing cost when $k$ is increased
- Low values of $k$ enable an efficient identification of the structural correlation patterns
Parallel Algorithms

- **Algorithms:**
  - **SCPM-PAR-BFS:** Computes the structural correlation using BFS and parallelization
  - **SCPM-PAR-DFS:** Computes the structural correlation using DFS and parallelization
  - **PAR-TOP-K:** Parallel algorithm for discovering the top-k structural correlation patterns

- Results for the SCPM-PAR-SAMP-BFS and SCPM-PAR-SAMP-DFS algorithms are omitted

- **Dataset:** SmallDBLP and DBLP

- **Parameter evaluated:** number of cores

- Evaluations consider only the execution time corresponding to the parallelized part of the algorithms
Parallel Algorithms

SCPM-PAR-DFS, SmallDBLP

SCPM-PAR-BFS, SmallDBLP

PAR-TOP-K, DBLP
Parallel Algorithms

- The algorithms SCPM-PAR-DFS and SCPM-PAR-BFS achieve **super linear speedup**
  - In addition to the expected gains in dividing such space among the threads, one particular thread may reduce the search space of the others by verifying that a vertex is part of a dense subgraph
- The PAR-TOP-K algorithm does not scale as well as the other algorithms
- In general, parallelization has shown to be a good strategy for efficient structural correlation pattern mining
Conclusions
Contributions

- **Problem statement:** Correlating vertex attributes and the existence of dense subgraphs in attributed graphs
- **Modeling:** Structural correlation, normalized structural correlation and structural correlation patterns
- **Algorithm design:** Search, pruning, sampling, and parallelization strategies to compute the structural correlation efficiently, identification of the top structural correlation patterns
- **Applications and evaluation:** Application of the structural correlation pattern mining to several real datasets from different domains and evaluation of the performance of the proposed algorithms
Limitations

- **Parameter setting:** We did not propose any technique for parameter setting in this work.
- **Performance:** We tried to analyze an attributed graph generated from the Wikipedia, with 3,135,812 vertices, 54,877,914 edges, and 1,756,599 attributes, for example, without success.
- **Applications:** It would be relevant to verify the applicability of the structural correlation pattern mining quantitatively.
- **Comparison with other methods:** We could not find any equivalent problem in the literature.
Future Work

- **Pruning:** New pruning techniques for structural correlation pattern mining
- **Distributed algorithms:** The design of distributed algorithms for structural correlation pattern mining
- **Applications:**
  - Graph clustering
  - Relational learning
  - Graph summarization and visualization
Structural Correlation Pattern Mining for Large Graphs

Master’s Thesis Defense

Arlei Lopes da Silva

Computer Science Department
ICEX
Universidade Federal de Minas Gerais – Brasil