Graph Query Reformulation with Diversity

Davide Mottin, University of Trento
Francesco Bonchi, Yahoo Labs - Francesco Gullo, Yahoo Labs
Pattern search

Query

\[
\text{OH} \quad \text{S} \quad \text{O}
\]

- Too many matches
- Results are not grouped
Finding specializations

510 matches

Specializations

O

OH $\rightarrow$ S $\equiv$ O

510 matches

O

OH $\rightarrow$ S $\equiv$ O
S $\rightarrow$ H$_3$C

382 matches

O

OH $\rightarrow$ S $\equiv$ O
H $\rightarrow$ CH$_3$

46 matches

O

OH $\rightarrow$ S $\equiv$ O
CH$_3$

448 matches

O

OH $\rightarrow$ S $\equiv$ O
SH

46 matches

O

OH $\rightarrow$ S $\equiv$ O

114 matches

CH$_3$
Applications

• Finding groups of molecules having a particular reagent
• Analyze a set of proteins to find diseases
• Workflow optimization
• Anomalies detection in a network
• Finding similar 3D shape search
Dealing with specializations in web and relational data

• Faceted Search
  • present aspects of the results [Roy08]

• **Query reformulation**
  • Modify some of the query conditions
    • In structured databases [Mishra09]
    • In web search [Dang10]

Frist Study of Problem on GRAPHS
Graph Query Reformulation

Query

Results

Reformulations: query supergraphs

Exponential number of reformulations
Challenges

• The number of reformulation is exponential
• Quantify the interestingness of a reformulation
• Finding query reformulations is \textbf{NP}-complete
Our Approach

Graph Query Reformulation with Diversity

- Finds $k$ meaningful specializations efficiently
Finding Meaningful Specializations

Find \( k \) meaningful specializations:
1. Span all the results
2. Present different aspects of the results
Graph Query Reformulation with Diversity

Problem

Find a set $Q^*$ of $k$ specializations of $Q$ that maximize a combination of coverage and diversity

$$f(Q) = cov(Q) + \lambda \sum_{Q', Q'' \in Q} div(Q', Q'')$$

$$Q^* = \arg\max_{Q \subseteq S_Q} f(Q)$$

subject to $|Q| = k$.

Theorem (NP-hardness)

The problem reduces to **MAX-SUM Diversification** Problem, so it is NP-hard
Solution: Greedy Algorithm

**Greedy**

While k-specializations are not found

1. Find the specialization leading to the maximum increment of the objective function (marginal gain)

2. Add the specialization to the results

**Theorem**
The algorithm is a ½-approximation

Finding the maximum gain is #P-complete [Valiant79]

**Solution**

Fast_MMPG: Branch and bound algorithm with quality guarantees
The multiplicity vector

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

Results

Output set of specializations
Upper bound on the Marginal gain

**Lemma**
The marginal gain increases if the multiplicity of the considered item is where $|Q|$ is the number of reformulations in the reformulated set constructed so far.

**Upper bound**: is the value of the objective function considering only results with multiplicity

$$\leq \frac{|Q|}{2}$$

**Theorem**
For a reformulation $Q' \in S \setminus Q$ it holds that

$$\max_{Q'' \in T_{Q'}} \Delta_f (Q, Q'') \leq \bar{\Delta}_f (Q, Q') =$$

$$= \frac{1}{2} \bar{u}_Q \cdot \bar{x}_Q^* + \lambda (\|\bar{m}_Q\| + |Q| \times \|\bar{x}_Q^*\| - 2\bar{m}_Q \cdot \bar{x}_Q^*) .$$
### Upper bound

\[ \leq \frac{|Q|}{2} \]

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

#### Results

![Graphs](image)

- **Output set of reformulations**
  - \(Q_1'\)
  - \(Q_2'\)
  - \(Q_3'\)
The Fast_MMPG Algorithm

\[ \bar{\Delta}_f(Q, Q'_1) = \text{upper bound} \]
\[ \Delta_f(Q, Q'_1) = \text{marginal gain} \]

Until the reformulation with the max upper bound and marginal gain is not found:

1. Expand the reformulation with the max upper bound
2. Prune Reformulations with marginal gain smaller than the upper bound so far

\[ \bar{\Delta}_f(Q, Q'_1) = 30 \]
\[ \Delta_f(Q, Q'_1) = 18 \]

\[ \bar{\Delta}_f(Q, Q'_2) = 21 \]
\[ \Delta_f(Q, Q'_2) = 20 \]

\[ \bar{\Delta}_f(Q, Q'_3) = 26 \]
\[ \Delta_f(Q, Q'_3) = 20 \]

\[ \bar{\Delta}_f(Q, Q'_{11}) = 22 \]
\[ \Delta_f(Q, Q'_{11}) = 22 \]

\[ \bar{\Delta}_f(Q, Q'_{12}) = 18 \]
\[ \Delta_f(Q, Q'_{12}) = 18 \]

\[ \bar{\Delta}_f(Q, Q'_{31}) = 18 \]
\[ \Delta_f(Q, Q'_{31}) = 18 \]

\[ \bar{\Delta}_f(Q, Q'_{32}) = 16 \]
\[ \Delta_f(Q, Q'_{32}) = 16 \]
Experimental Setup

• **Datasets:**
  • AIDS: 10k chemical compounds
  • Financial: 17k transaction workflows
  • Web: 13k interactions with a recommender system

• **Baseline algorithms:**
  • k-freq: returns top-k frequent supergraphs of a query
  • LIndex: informative patterns index

• **Experiments:**
  • Time and objective function value varying k, query size, $\lambda$
  • Anecdotal
  • Scalability
**Time Comparison**

**Number of reformulations**
1. k-freq runs only slightly faster
2. Time increases linearly in k
3. Fast_MMPG has real-time performance

**Query size**
1. Fast_MMPG comparable to k-freq
2. Time decreases with query size (less reformulations)
Objective function gain

<table>
<thead>
<tr>
<th></th>
<th>(\lambda)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Fast_MMPG</td>
<td>433</td>
</tr>
<tr>
<td>k-freq</td>
<td>409</td>
</tr>
<tr>
<td>gain (%)</td>
<td>6</td>
</tr>
</tbody>
</table>

Analysis
1. Lambda correctly moves the objective function towards diversity
2. k-freq only captures coverage

\[ f(Q) = cov(Q) + \lambda \sum_{Q',Q'' \in Q} div(Q', Q'') \]
Qualitative evaluation

Query \( C = O \)

*Fast_MMPG*

\[
\begin{align*}
\text{C} & \quad \text{O} \\
\text{O} & \quad \text{OH} \\
\text{O} & \quad \text{CH}_3 \\
\text{C} & \quad \text{Fe} \\
\text{C} & \quad \text{NH}_2 \\
\text{C} & \quad \text{CH}_3
\end{align*}
\]

*k-freq*

\[
\begin{align*}
\text{C} & \quad \text{C} \\
\text{C} & \quad \text{CH}_3 \\
\text{C} & \quad \text{C} \\
\text{C} & \quad \text{CH}_2 \\
\text{C} & \quad \text{NH}_2 \\
\text{C} & \quad \text{NH} \\
\text{C} & \quad \text{CH}_2
\end{align*}
\]

**Analysis**

- k-freq finds reformulation of the same superquery
- Fast_MMPG returns reformulations with more diversified structures
Conclusions

- First study of the problem in graph databases
- **Principled** objective function optimizing **coverage** and **diversity**
- Algorithmic solutions with quality guarantees and **real time responses**
Thank you!

Questions?