ρ-Queries: Enabling Querying for Semantic Associations on the Semantic Web

WWW2003 (Budapest, May 23, 2003)
Paper Presentation

Kemafor Anyanwu
Amit Sheth
Large Scale Distributed Information Systems Lab
University of Georgia

This material is based upon work supported by the National Science Foundation under Grant No. 0219649.
From ..... 

Finding things 

to ..... 

“Finding out about” [Belew00] 
relationships!
Outline

- Semantic Associations: Introduction
- A Formal Framework for Semantic Associations on the Semantic Web
- $\rho$-Queries For Discovering Semantic Associations
  - Implementation Strategies & Issues
- Related Work
- Conclusion & Future Work
Web Search/Query Techniques are “Entity-Centric”
But......

“An object by itself is intensely uninteresting”.

Grady Booch, Object Oriented Design with Applications, 1991
We need

- Mechanisms for querying about and retrieving complex relationships between entities.

1. A is related to B by $x.y.z$
2. A is related to C by
   i. $x.y'.z'$
   ii. $u.v$ (undirected path)
3. A is “related similarly” to B as it is to C
   ($y' \subseteq y$ and $z' \subseteq z \rightarrow x.y.z \cong x.y'.z'$)
   So are B and C related?
Why do we need this?

- Very useful in information analytics
  - national security
  - business intelligence

- Avoids the task of familiarizing oneself with schemas in order to formulate queries
  - especially when multiple schemas are involved!
Example in 9-11 context

- What are relationships between Khalid Al-Midhar and Majed Moqed?
  - **Connections**
    - Bought tickets using same frequent flier number
  - **Similarities**
    - Both purchased tickets originating from Washington DC paid by cash and picked up their tickets at the Baltimore-Washington Int'l Airport
    - Both have seats in Row 12

- “What relationships exist (if any) between Osama bin Laden and the 9-11 attackers”
A Foundation for Semantic Associations on the Semantic Web
Complex Relationships?

- Traditional notions of relationships are captured by **single** n-ary relations
  - e.g. RDF:Property, UML Association, E-R:relationship, etc.
- Complex relationships can be viewed as specific compositions of multiple single n-ary relations
  - e.g. Sequence composition of binary relations allows us to capture paths
- Relation Sequences + certain operations allow us to detect very interesting relationships
  - Connectivity
  - Similarity
Semantic Web

- RDF is the current W3C standard for metadata representation on the Semantic Web
- Other proposals include OWL, DAML+OIL, UML, Topic Maps, etc.
- In RDF, the basic unit of relationship is a Property
Formal Data Model for RDF

(Karvounarakis et al 2002) gives a formalization of RDF/RDFS which forms the basis for a typed RDF query language – RQL.

- It provides a type system for RDF Schemas
- For each type e.g. class type $\tau_c$, property type $\tau_p$, there is a mapping $[[ ]]$ to its members
- e.g. for a property type $p$, $[[p]]$ is defined as $\{[[v_1, v_2]] \mid v_1 \in [[[p \text{.domain }]], v_2 \in [[[p \text{.range }]]] \} \cup \{ [[[p']] \mid ' \subseteq p\}$
We add

- The notion of an RDFS *Schema Set*. Basically, a union of a set of RDF Schemas supplying the context for a query
  - In the example, Flight + Banking Schemas
- The notion of a *Property Sequence*, which is the sequential composition of RDF Properties and define relations on Property Sequences
- A formalization for *Semantic Associations* based on Property Sequences and their relations
<table>
<thead>
<tr>
<th>Property Sequence</th>
<th>Finite sequence of properties $\text{PS} = [P_1, P_2, P_3, \ldots P_n]$, $P_i$ is a property defined in an RDF Schema $R_{S_j}$ of a schema set RSS. e.g. $[\text{purchased, paidby}]$.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$[[\text{PS}]] \subseteq \prod_{i=1}^{n}[[P_i]]$ such that $\text{ps} \in [[\text{PS}]]$ implies</td>
</tr>
<tr>
<td></td>
<td>i. $\text{ps}[i] \in [[P_i]]$ for $1 \leq i \leq n$</td>
</tr>
<tr>
<td></td>
<td>ii. $\text{ps}[i][1] = \text{ps}[i+1][0]$</td>
</tr>
<tr>
<td>Joined Property Sequences ((\square_{\rho}))</td>
<td>$\text{PS}<em>1 \Join</em>{\rho} \text{PS}_2 \leftarrow \exists c \in (\text{PS}_1.\text{NodesOfPS()} \cap \text{PS}_2.\text{NodesOfPS()}))$. c is called join node</td>
</tr>
<tr>
<td>(\rho)-Isomorphic Property Sequences ((\cong_{\rho}))</td>
<td>$\text{PS}<em>1 \cong</em>{\rho} \text{PS}_2 \leftarrow$</td>
</tr>
<tr>
<td></td>
<td>i. $\text{PS}_1 = [P_1, P_2, P_3, \ldots P_m]$, $\text{PS}_2 = [Q_1, Q_2, Q_3, \ldots Q_m]$</td>
</tr>
<tr>
<td></td>
<td>ii. for all $i$, $1 \leq i \leq m$: $P_i = Q_i$ or $P_i \subseteq Q_i$ or $Q_i \subseteq P_i$ ((\subseteq) means subpropertyOf )</td>
</tr>
<tr>
<td></td>
<td>Note that the Property Sequences need not be exact to be $\rho$-Isomorphic, just similar.</td>
</tr>
</tbody>
</table>

A sequence such as "awarded.paidby" which means that a passenger was awarded a ticket, paid for by frequent miles is considered $\rho$-Isomorphic to "purchased.paidby".
Semantic Associations
\[\text{\(\rho\)-pathAssociation}\]

- Let PS be a Property Sequence and \(ps \in [[PS]]\).
- If \(x\) and \(y\) are the origin/terminus and terminus/origin of \(ps\) respectively,
  \[\rho\text{-pathAssociated}\ (x, y)\]
ρ-joinAssociation

- ρ-joinAssociated (x, y) ←
  a) ∃ PS₁, PS₂: PS₁ ⊙ ρ PS₂
  b) ∃ ps₁, ps₂: ps₁ ∈ [[ PS₁ ]], ps₂ ∈ [[ PS₂ ]]
     i. x is the origin of ps₁ and y is the origin of ps₂ or
     ii. x is the terminus of ps₁ and y is the terminus of ps₂.

```
ρ-joinAssociated
```

```
M’mmed
```

```
Attar
```

```
Marwan
```

```
Al-Shehhi
```

```
fname
```

```
lname
```

```
purchased
```

```
credited
```

```
paidby
```

```
join nodes
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```

```
lname
```

```
for
```

```
holder
```

```
fname
```
ρ-IsoAssociation

λ-IsoAssociated (x, y) ⊨

a)  ∃ PS_1, PS_2 : PS_1 ≅ ρ PS_2

b)  ∃ ps_1, ps_2 : ps_1 ∈ [[ PS_1]], ps_2 ∈ [[PS_2]]

i. x is the origin/terminus of ps_1 and y is the origin/terminus of ps_2.
ρ-Queries for Discovering Semantic Associations
ρ-Queries

Let \( \tau_U^{(2)} = \{ \{x, y\} : x, y \in \tau_U \text{ and } x \neq y \} \),
\( \text{PS} = \{\text{PS} : \text{PS is a Property Sequence}\} \),
\( \text{PS}^{(2)} = \{\{\text{PS}_1, \text{PS}_2\} : \text{PS}_1, \text{PS}_2 \text{ are Property Sequences}\} \)

A \( \rho \)-Query \( Q \) maps from a pair of keys to the \( \text{PS} \) and \( \text{PS}^{(2)} \) in the following manner:

- \( \rho: \tau_U^{(2)} \rightarrow 2^{\text{PS}} \)
- \( \rho\rho: \tau_U^{(2)} \rightarrow 2^{\text{PS}(2)} \)
- \( \rho\rho: \tau_U^{(2)} \rightarrow 2^{\text{PS}(2)} \)
Implementation Approaches for $\rho$-Operators

- Exploit existing RDF storage & query infrastructure:
  - Persistent Stores $\rightarrow$ Translations to query expressions at data store layer, guided by index structures
  - Memory-Resident Stores $\rightarrow$ Employ graph traversal algorithms

- Alternative Representation with complimentary indexes and algorithms i.e. search-engine type Strategy
Evaluation Testbed Ontology

RDF Description Base \textit{wrt} to this schema is populated from 30+ sources
Use of Semagix Freedom for automatic ontology-driven metadata extraction to create large RDF description-base from many sources

Semagix Freedom is based on prior research at the LSDIS Lab -> resulting SCORE technology
\( \rho \text{-PathAssociated(Transfer1, Iraq)} \)

- \( \text{Transfer1} \rightarrow \text{Account2} \rightarrow \text{IraqInternationalBank} \rightarrow \text{Iraq} \)
- \( \text{Transfer1} \rightarrow \text{Account2} \rightarrow \text{SaddamHussein} \rightarrow \text{Iraq} \)
- \( \text{Transfer1} \rightarrow \text{Account2} \rightarrow \text{SaddamHussein} \rightarrow \text{IraqGovernment} \rightarrow \text{Iraq} \)
\( \rho\text{-joinAssociated}(\text{Account2, Email1}) \)

<table>
<thead>
<tr>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account2 ( \rightarrow ) IraqInternationalBank ( \rightarrow ) Iraq</td>
</tr>
<tr>
<td>Email1 ( \rightarrow ) SaddamHussein ( \rightarrow ) Iraq</td>
</tr>
<tr>
<td>Account2 ( \rightarrow ) IraqInternationalBank ( \rightarrow ) Iraq</td>
</tr>
<tr>
<td>Email1 ( \rightarrow ) SaddamHussein ( \rightarrow ) SaddamHussein</td>
</tr>
<tr>
<td>Email1 ( \rightarrow ) SaddamHussein</td>
</tr>
</tbody>
</table>
\( \rho\text{-IsoAssociated}(\text{Account2}, \text{Account1}) \)

- **Account2** → at → **IraqInternationalBank** → locatedIn → **Iraq**
- **Account1** → at → **PakistanInternationalBank** → locatedIn → **Pakistan**

- **Account2** → p_holder → **SaddamHussein** → fromLocation → **Iraq**
- **Account1** → p_holder → **OsamaBinLaden** → fromLocation → **SaudiArabia**

- **Account2** → p_holder → **SaddamHussein** → leaderOf → **IraqGovernment** → locatedIn → **Iraq**
- **Account1** → p_holder → **OsamaBinLaden** → leaderOf → **AlQeada** → locatedIn → **Afghanistan**
Current & Future Work

- Data Preprocessing and Serialization
- Context
  - Specification & Representation
  - Streamline Query Processing
  - Ranking
- Query Processing Optimizations
  - Index structures
  - Heuristics
    - Complexity = \( \sum_{l=1}^{(n-1)} \) (# paths of length \( l \)) (probability of keeping path of length \( l \)).
- Result Presentation
- Spatio-Temporal constraints
Related Work

- IR over XML, Relational Databases
  - [Hristidis et al 02,03], [Theobald et al 02], [Guha et al 03]
- Support for Path Expressions in Semi-Structured and Object-Oriented models
  - [Christophides et al 94], [Abiteboul et al 97], [Buneman et al 00], etc.
- Graph Databases
  - [Mendelzon, Wood 89]
More info.

  - Project description, papers, presentations