Semanta: An Ontology Driven Semantic Link Analysis Framework

Mullai Shanmuhan
Advisor: Dr. I. Budak Arpinar
LSDIS Lab
Computer Science Department
The University of Georgia
Motivation

- Information overload
  - Many users are increasingly overwhelmed by the amount of information available
  - Examples:
    - Biomedical research, investigative or watchdog journalism

- Knowledge Starvation
  - Inability to make informed decisions backed by facts not easily discernible

- Semantic Web
  - Advances have made it possible to effectively capture the knowledge of a domain through various markup languages

- Tools aiding information analysis are needed in handling knowledge starvation
Problem Statement

- A framework for finding semantic links among entities is needed for effective decision-making
  - Links are previously unknown or hidden within knowledge-bases and/or information resources
  - Finding them might be a very cumbersome task for users
Semantic Links

- A transitive link between any two entities/classes
  E.g. Finding possible relationships between the Energy sector and the Republican Party of United States

- User may have preferences over links to be found (i.e., entities linked by a specific relation)
  E.g. Finding entities that are related to Saddam Hussein through the positive-associate relationship.
Challenges

- A relatively new scientific problem
- No existing mechanisms for querying semantic links
- Need for an orchestrated query mechanism for both knowledge-bases and information resources
- Finding useful links and presenting the results
Semantic Network
Semantic Network

- **Objective**
  - To effectively capture data in various domains

- **Layers of knowledge and information**
  - Ontology Layers
    - Class Base Layer
    - Object Base Layer
  - Information Source Layer

- **Advantages:**
  - The idea of “an ontology-driven search” for semantic links
  - Reduce cost
Class Base Layer

- Concepts
  - Name

- Relationships
  - Parent-child
  - Container relationship
  - User-defined – e.g. works-for, mother-of

- Resource Description Framework Schema (RDFS)

- Schema files capture the concepts of a specific domain

- Uniform vocabulary across schema files
Object Base Layer

- Instances of concepts
- RDF files

**Statement:** Condoleezza Rice is a board member of Chevron Corporation

**Graph Model:**

- **Condoleezza Rice**
  - Name
  - Type
- **Chevron Corporation**
  - Company
  - Type

**Triplet:** <Condoleezza Rice> <board member> <Chevron Corporation>
Information Source Layer

- Source for richer and fresh information
- Complements domain knowledge in the Ontology layers
- Defined using multiple XML documents
- Shares the same vocabulary with other layers
Semantic Link Queries
Semantic Link Queries

- **Query**
  \[ < [O_1 \ o_1 \ | \ O_1.A_1 | \ o_1.a_1 | \ x] \ [R_1 | x] \ [O_1 \ o_1 \ | \ O_1.A_1 | \ o_1.a_1 | \ x]> \]

- **User-defined Constraints**
  - Ontology constraints
    - Explicit inclusion/omission of ontologies
  - Semantic constraints
    - Relation Ontology
  - Span
    - To limit the search in Ontology layers
  - Number of results
    - Number of links between entities
Query Input Interface
### Semantic Link Queries

<table>
<thead>
<tr>
<th>Type</th>
<th>Example</th>
</tr>
</thead>
</table>
| $O_i \times O_k$ | 1) 'University' x 'Music Groups'  
                        2) 'Mountain' x 'Casualties' |
| $O_i \times a_k$ | 1) 'University' x 'R.E.M.'  
                        2) 'Nyiragongo'(Volcano) x 'Casualty' |
| $O_i \times a_k$ | 1) 'UGA' x 'R.E.M.'  
                        2) 'Bush' x 'Enron' |
| $O_i \cdot a_k \times a_k$ | 'AlQeida.Afghanistan' x 'Baath Party' |
| $O_i \times R_1 x$ | 'Person' 'positive-associate' x |
| $O_i \cdot R_1 x$ | 'Halliburton Company' 'employs' x |

**Entities Based Queries**

**Relations Based Queries**
Entities Based Queries

- Finding links between any two entities \((e_1, e_2)\)
- Entity: Class, Property, Literal
- Query types:
  - Type 1: class/property, class/property
  - Type 2: class/property, literal
  - Type 3: literal, literal
Entities Based Queries (contd.)

- **Type 1**
  - Class/property, class/property

- **Kinds of relationships**
  - Class-attribute relationship
    - Address, city
  - Parent-child relationship
    - Industry <subclass-of> Telecom <subclass-of> Wireless Communication Services
  - Co-classes relationship
    - Event <funded-by> Organization, Project <funded-by> Organization
  - Linked classes relationship
    - Person <works-for> Company <funded-by> Organization
Entities Based Queries (contd.)

- **Type 2:**
  - Class/property, literal
  - Example:
    - Find links between the ‘Energy Sector’ and the ‘Republican Party of United States’

- **Type 3:**
  - Literal, literal
  - Object Base layer or text elements in Information Source layer
  - Example:
    - Find links between ‘Liming Cai’ and ‘Robert Robinson’
Relations Based Queries

- Finding entities that are related to a given entity through a user-specified relationship


**OR-Complex Relation**

- A group of member relations
- No order among them
- Does not require the presence of all member relations
AND-Complex Relation

- All member relations will have to be present
System Architecture
System Architecture

- **Semanta Knowledge Store**
  - Class base Store (RDFS Files)
  - Object base Store (RDF Files)
  - Information Store (XML Files)

- **JENA API**
- **XPATH, JDOM API**

- **Semanta API**

- **Hints Generator**
- **Path Finder**
- **Direct Path Finder**
- **Indirect Path Finder**
- **ISS Path Finder**

- **Knowledge Store Updater**

- **Protégé**

- **Information Source Extractor**

- **Gather and Rank**

- **Results?**
Semanta API

- CB Layer API
  - Over Jena 1.5.0, to access the RDFS files
  - Class Based API
  - Property Based API

- OB Layer API
  - Over Jena 1.5.0, to access the RDF files
  - Methods to get details of CB layer, given the OB layer details
Semanta API (contd.)

- IS Layer API
  - To access XML Documents
  - Stored via Apache Xindice
  - Uses Xpath API for accessing parts of the documents
  - JDOM, for accessing and manipulating XML documents
Searching the Ontology Layers

- Find paths within the Ontology layers (Path Finder)
- Generated hints for IS Layer processing (Hints Generator)
Path Finder

- Finding a path in the Ontology layers – finding a ‘semantic relationship’ between the entities

- Semantic relationship
  - Class-property relation
  - Property-property relation
  - Class-class relation
Path Finder (contd.)

Class-property Relationship

- Property is an attribute of the class
- Property (directly) links the class to another class
- Property is a transitive link to another class
Path Finder (contd.)

**Property-property Relationship**

- Both properties belong to the same class, or the same instance of the class
- Hierarchical relationship between the properties
Class-class Relationship

- Instance of the classes have property values that match
- Hierarchical relationship between the classes
- Classes are linked through property links
Path Finder (contd.)

- Start from entity $e_1$
  - Use Semanta API to look for $e_2$ in nodes connected through parent-child, class-attribute, co-classes, linked-classes relationship using Semanta API

- Direction is not significant

- Analogous to Breadth First Search (BFS)
  - Complexity - $O(n^{span})$

- Heuristics to improve search:
  - Directed BFS
  - Interactive Deepening
Directed BFS

Favor paths passing through user-defined domains (i.e., ontology constraints) and filter out other paths

\[ P = [\text{window\_size}, \text{span}] \]

Nodes at level \( l \):
1. Neighboring nodes belong to the same domain
2. None of the neighboring nodes belong to any specified domain
3. Neighboring nodes belong to multiple domains
Interactive Deepening

- User is able to control the search
- Depth and two-way search provides flexibility in interaction level and makes it easy to progress towards “meeting” paths respectively
- \( P = [\text{depth, span}] \)
- At \( l = \text{depth} \)
  - Existing paths are presented to the user.
  - User selects the nodes at this level, which s/he wishes to pursue.
  - Only the selected nodes are considered for finding subsequent paths.
  - Alternation (i.e., two-way search)
Traversing the Semantic Links
Finding the Semantic Links
(Overall Picture)

1. Identify candidate nodes
2. Check for links in Ontology layers
3. Gather Hints from Ontology Layers
4. Generate XPath Queries
5. Check for links in Information Source layer
6. Present the results
Finding the Semantic Links (contd.)

- Find links between ‘Energy Sector’ and ‘Republican Party’
- $e_1 = \text{Energy}$, $e_2 = \text{Republican Party}$
- Identify category - Type 2 category
- Identify related nodes
Paths in Ontology Layers

- Check for relations
  - Class-attribute
  - Parent-child
  - Co-classes
  - Linked classes
Generating Hints

- **Hint:**
  - Collection of class nodes, instance nodes and properties in the vicinity of an entity

- Gathered from the Ontology layers
  - Class nodes, Instance nodes, Properties

- Used to assist looking for links in Information Source Layer
Searching the Information Source Layer

- **IS Layer is accessed**
  - No links exist in the Ontology layers
  - Entities are not present in the Ontology layers

- **Paths in IS Layer**
  - Direct path
    - Parent-child or sibling relationship among entities
  - Indirect path
    - Parent-child or sibling relationship based on the hints
    - Matching patterns between documents
Collect relevant documents

- Generate XPath Queries

```xml
//*[normalize-space(.)="Energy"]//..//*[normalize-space(.)="Halliburton Company"]
//*[normalize-space(.)="Energy"]//..//*[normalize-space(.)="Chevron Corporation"]
//*[normalize-space(.)="Energy"]//..//*[normalize-space(.)="Tom Brown Inc."]
//Industry*[normalize-space(.)="Energy"]
//Political_Organization*[normalize-space(.)="Republican Party"]
```
Direct Paths

- Paths based on parent-child, sibling relationship

```xml
<election year="2000">
  <party="Republican Party">
    <contributor>
      <company sector="Energy">
        <name>Chevron Corporation</name>
      </company>
      <amount>113,800</amount>
    </contributor>
  </party>
</election>
```
Indirect Paths

- Paths exist between elements of the hint-sets
Indirect Paths (contd.)

- Matching patterns between documents
- Pattern:
  - Identical tag and text elements along with the hierarchical structure
<person>
  <name>
    <given> Robert </given>
    <family> Robinson </family>
  </name>
  <occupation>
    <profession>
      <industry> Education </industry>
      <company> Univ of Michigan </company>
    </profession>
  </occupation>
</person>

<person>
  <name>
    <given> Liming </given>
    <family> Cai </family>
  </name>
  <occupation>
    <profession>
      <industry> Education </industry>
      <company> Univ of Georgia </company>
      <role> Professor </role>
      <start_date> 0/0/2002 </start_date>
    </profession>
  </occupation>
</person>
Presenting Semantic Links

- User should be able to comprehend the results easily thereby aiding him/her in the end decision making.
- More detail of each result should be available, where needed.
- Information regarding the source from which results have been inferred should be available.
- Summarizing results, based on parameters such as relationships, path-lengths, etc., should be present on request.
Conclusions

- Three tier knowledge store
  - Class Base Layer, Object Base Layer, Information Source Layer

- Technologies of Semantic Web
  - RDFS, RDF, XML

- Classification of queries
  - Entities based queries
  - Relations based queries

- Semanta API and prototype Implementation
Future Work

- Template Relation
  - Enabling richer information analysis

- Visualization Tools
  - Selecting/omitting sections of ontologies or documents in Information Source layer
  - Presenting the process of finding paths to the user

- XML Schema
  - To enforce stronger binding between the Ontology Layers and Information Source Layer

- XPointer & XLink
  - To connect elements in multiple documents
Relations Based Queries (contd.)

**Template Complex Relation**

- Defined by a set of triplets
- Represent classes related by properties
- Attributes:
  - Multiplicity
  - Transitivity
  - Equivalence
  - Inverse
- Future research item
<person>
  <name>
    <first>Condoleezza</first>
    <last>Rice</last>
  </name>

  <occupation>
    <industry>Energy</industry>
    <company>Chevron Corporation</company>
    <role>Member, Board of Directors</role>
    <duration>…</duration>
  </occupation>
  …
  …

  <occupation>
    <industry>Finance</industry>
    <company>Charles Schwab Corporation</company>
    <role>Member, Board of Directors</role>
    <duration>…</duration>
  </occupation>
</person>