A Directed Hypergraph Model for RDF

M.Sc. Amadís Martínez, Dr. María Esther Vidal
Department of Computer Science and Information Technology
Universidad Simón Bolívar
Caracas, Venezuela

Workshop “Query Optimization for the Semantic Web”
Madrid, Spain, May 31, 2007
Our Objective

- Provide a formal representation of RDF to reduce space and time complexity
- Define physical operators to implement SPARQL efficiently
Agenda

• Motivation
• Related Work
• Our Approach
• Initial Results
• Conclusions and Future Work
Motivation

• RDF is a language proposed by the W3C to express metadata about resources in the Web
• An RDF management system requires support for two main tasks:
  1. Answering queries posed by users and software agents
  2. Semantic inference from the data to discover relationships between resources
Motivation

• The RDF data model allows several graph-based representations:
  – Labeled directed graphs
  – Bipartite graphs
  – Undirected hypergraphs

• Each one of these representations has its own limitations with respect to:
  – RDF data model expressive power
  – Support for the tasks of query answering and semantic reasoning
Related Work

- An RDF document can be viewed as a graph: nodes are resources and arcs are properties.
- Given an RDF graph $T$ (without blank nodes), the space complexity to storage $T$ is $O(|T|)$ and the time complexity of elemental query answering on $T$ is $O(|T|^k)$, $k$ is an integer number.
- The main idea is to preserve the coefficients and exponents of these polynomials as low as possible.
Related Work

Labeled Directed Graph (LDG) model

- Each RDF triple \( (s,p,o) \in T \) is represented by a labeled arc, \( s \rightarrow o \)
- The size is \( |V| \leq 2|T| \) and \( |E| = |T| \)
- This approach may violate some graph theory constraints
- Most widely used representation, but it cannot be considered a formal model
Undirected Hypergraph (UH) model

- Each RDF triple $t = (s,p,o) \in T$ is a hyperedge and each element of $t$ (subject $s$, predicate $p$, and object $o$) is a node
- The size is $|V| = |\text{univ}(T)|$ and $|E| = |T|$  
- UH represents RDF documents as a generalization of undirected graphs, losing the notion of direction, which impacts semantic reasoning
Bipartite Graph (BG) model

• There can be two types of nodes:
  – Statement nodes St (one for each RDF triple \((s,p,o) \in T\))
  – Value nodes Val (one for each element \(x \in \text{univ}(T)\))

• The size is \(|V| = |\text{univ}(T)| + |T|\) and \(|E| = 3|T|\)

• Reification, entailment, and reasoning have not been addressed yet
Indexing Techniques

- Two main indexing techniques:
  - Position-based index
  - Path-based index
- The key idea behind indexing is to “speed-up” the task of query answering
Related Work

• Indexes are normally additional structures build on the representation of an RDF document
  – A position-based index requires around 30%-35% of the size of the indexed data
  – A path-based index requires around 40%-45% of the size of the indexed data
Our Approach

• A Directed Hypergraph (DH) Model for RDF to:
  – Represent RDF documents, overcoming the limitations found in other representations
  – Provide an option for efficient storage and query evaluation of RDF documents
Let $T$ be an RDF graph. $H(T) = (W,E,\rho)$ is the RDF Directed Hypergraph representing $T$, where:

- **Nodes:** $W = \{w : w \in \text{univ}(T)\}$
- **Hyperarcs:** $E = \{e_i : 1 \leq i \leq |T|\}$
- $\rho : W \times E \to \{s, p, o\}$ is the role function of nodes w.r.t. hyperarcs

Let $t \in T$, $e \in E$, and $w \in W$ such that $w \in \text{head}(e) \cup \text{tail}(e)$. Then:

- $(\rho(w, e) = s) \iff w \in \text{tail}(e) \land w \in \text{sub}(\{t\})$
- $(\rho(w, e) = p) \iff w \in \text{tail}(e) \land w \in \text{pred}(\{t\})$
- $(\rho(w, e) = o) \iff w \in \text{head}(e) \land w \in \text{obj}(\{t\})$
  - head(e) is the set of incoming nodes of $e$
  - tail(e) is the set of outcoming nodes of $e$
Our Approach

• Given an RDF graph $T$, each node corresponds with an element $w \in \text{univ}(T)$
• The information is only stored in the nodes, and the hyperarcs only preserve the role of each node and the concept of direction
• This approach must require less amount of memory than other representations
• The size is $|V| = |\text{univ}(T)|$ and $|E| = |T|$  
• DH defines implicit position-based indexes for a RDF document which can support efficient evaluation of queries over the document
Figure 1: DH for an RDF graph
Figure 2: DH for an RDF graph (blank nodes)
Figure 3: DH for an RDF graph (reification)
Initial Results

1. A directed hypergraph model for RDF as a proposal to represent RDF documents efficiently
2. An analysis of the expressive power of this representation with respect to the RDF data model
3. A formal study of the space complexity of this representation to store the information
4. An empirical study of the impact of this approach on the task of query answering
Initial Experimental Results

- Labeled Directed Graph (LDG) and Directed Hypergraph (DH) representations were studied empirically.
- A set of twenty synthetic RDF documents randomly generated using a uniform distribution were considered in this study.
- Around 25% of the RDF triples contained resources that simultaneously played the role of subject, predicate or object.
- Documents considered in this experimental study correspond with simple RDF graphs.
Initial Experimental Results

- Two metrics were used to accomplish this preliminary experimental study:
  1. The space in memory required to load each document, measured as the number of elements (nodes and arcs) required to storage information
  2. The number of comparisons required to answer an elemental query
- LDG approach showed a trend of linear dependence on the size of the document, while DH exhibited a more independent behavior
Initial Experimental Results

Figure 4: Space Complexity

<table>
<thead>
<tr>
<th>Size of the RDF Document</th>
<th>Space Required in Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>3000</td>
</tr>
<tr>
<td>2000</td>
<td>5000</td>
</tr>
<tr>
<td>3000</td>
<td>7000</td>
</tr>
<tr>
<td>4000</td>
<td>9000</td>
</tr>
<tr>
<td>5000</td>
<td>11000</td>
</tr>
<tr>
<td>6000</td>
<td>13000</td>
</tr>
<tr>
<td>7000</td>
<td>15000</td>
</tr>
<tr>
<td>8000</td>
<td>17000</td>
</tr>
<tr>
<td>9000</td>
<td>19000</td>
</tr>
<tr>
<td>10000</td>
<td>21000</td>
</tr>
</tbody>
</table>

Legend:
- DH
- LDG

Motivation | Related Work | Our Approach | Initial Results | Conclusions / Future Work
Initial Experimental Results

![Space Complexity Graph](image)

Figure 5: Space Complexity
Initial Experimental Results

Figure 6: Number of Comparisons for Elemental Query Answering

Motivation | Related Work | Our Approach | **Initial Results** | Conclusions / Future Work
Initial Experimental Results

Figure 7: Number of Comparisons for Elemental Query Answering
Conclusions / Future Work

- Initial results make believe that this approach scales better than existing representations.

- Future work will concentrate on four major aspects:
  1. Extend this representation for RDFS graphs
  2. Develop query evaluation algorithms for conjunctive and SPARQL queries
  3. Study the impact of this model on the tasks of query answering and semantic reasoning
  4. Conduct empirical studies to analyze the goodness of this model