Strider\textsuperscript{R}: Massive and distributed RDF Graph Stream Reasoning

Xiangnan REN, Olivier CURÉ, Hubert Naacke, Jérémy LHEZ, Ke Li

LIGM - LIP6 CNRS, FRANCE
OUTLINE
Agenda of the Presentation

- CONTEXT
- ARCHITECTURE OVERVIEW
- ENCODING - REWRITING
- CONCLUSION
Importance of stream reasoning
DATA STREAMS EVERYWHERE
WAVES PROJECT

➢ Smart water network management
  • Data streams from sensors
  • Filtering errors in measures
  • Identify sources in external events

➢ Main partner: Suez
  • 650 collaborators in Europe
  • 73 billion $US in R&D

➢ French project
  • http://www.waves-rsp.org/
Why water management?

Water SUPPLIED to the network - Water BILLED to customers = NON-REVENUE WATER (NRW)

- Billed water 65%
- NRW 35%

48.6 billion m³/year

Loss of US$14 billion/year

$2 \times$ the annual domestic water consumption of the USA
WAVES PROJECT

➢ Objectives:
  ● Robust real time engine, modular, flexible, intelligent
  ● Distribution

➢ RDF representation
  ● Integration of data/knowledge from different sources
  ● Reasoning capabilities

➢ Other applications:
  ● Banking/payments, climate, energy, power consumption, etc
ISSUES OF STREAM PROCESSING

➢ Solutions specific to the reasoning tasks
  • Materialization: huge amounts of data
  • Query rewriting: execution time

➢ Lack of performance for heavy data load

➢ Compression efficiency
  • No distribution
  • Decompression process
ARCHITECTURE

Strider organization
STRIDER ARCHITECTURE

➢ Data flow management:
  • Apache Kafka
  • Data streams partitioned

➢ Computing core
  • Query processing
  • Parallel query execution

➢ Encoding of the data
  • Static knowledge base: offline encoding (Abox + Tbox)
  • Dynamic data: encoding on the fly (Abox)

ISWC 2017, p.559-576
ENCODING - REWRITING

An encoding form conserving the hierarchy used to rewrite the queries
LITEMAT

➢ **Principle:**
  - Binary structure conserving the semantics of the ontology
  - Each identifier is prefixed by its parent’s
  - Conversion as integer identifiers
  - Supports RDFS

➢ **Advantages:**
  - Compression maintaining hierarchy
    ✓ No need for ontology at query runtime
    ✓ Execution performance
  - Easy query rewriting
LITEMAT EXAMPLE

Concepts

<table>
<thead>
<tr>
<th>owl:Thing</th>
<th>DUL:PhysicalObject</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ssn:Sensor</td>
</tr>
<tr>
<td></td>
<td>ssn:SensingDevice</td>
</tr>
<tr>
<td></td>
<td>dbo:Engine</td>
</tr>
<tr>
<td></td>
<td>DUL:SocialObject</td>
</tr>
</tbody>
</table>

Compression

1
# LITEMAT EXAMPLE

<table>
<thead>
<tr>
<th>Concepts</th>
<th>Compression</th>
</tr>
</thead>
<tbody>
<tr>
<td>owl:Thing</td>
<td>1</td>
</tr>
<tr>
<td>-&gt; DUL:PhysicalObject</td>
<td>101</td>
</tr>
<tr>
<td></td>
<td>-&gt; ssn:Sensor</td>
</tr>
<tr>
<td></td>
<td>-&gt; ssn:SensingDevice</td>
</tr>
<tr>
<td></td>
<td>-&gt; dbo:Engine</td>
</tr>
<tr>
<td>-&gt; DUL:SocialObject</td>
<td>110</td>
</tr>
</tbody>
</table>
## LITEMAT EXAMPLE

<table>
<thead>
<tr>
<th>Concepts</th>
<th>Compression</th>
</tr>
</thead>
<tbody>
<tr>
<td>owl:Thing</td>
<td>1</td>
</tr>
<tr>
<td>DUL:PhysicalObject</td>
<td>101</td>
</tr>
<tr>
<td>ssn:Sensor</td>
<td>10101</td>
</tr>
<tr>
<td>ssn:SensingDevice</td>
<td>10110</td>
</tr>
<tr>
<td>dbo:Engine</td>
<td>110</td>
</tr>
<tr>
<td>DUL:SocialObject</td>
<td></td>
</tr>
</tbody>
</table>
## LITEMAT EXAMPLE

<table>
<thead>
<tr>
<th>Concepts</th>
<th>Compression</th>
</tr>
</thead>
<tbody>
<tr>
<td>owl:Thing</td>
<td>1</td>
</tr>
<tr>
<td>DUL:PhysicalObject</td>
<td>101</td>
</tr>
<tr>
<td>ssn:Sensor</td>
<td>10101</td>
</tr>
<tr>
<td>ssn:SensingDevice</td>
<td>101011</td>
</tr>
<tr>
<td>dbo:Engine</td>
<td>10110</td>
</tr>
<tr>
<td>DUL:SocialObject</td>
<td>110</td>
</tr>
</tbody>
</table>
Identify the subclasses specific to a concept in a specific interval

- e.g. \([40, 48]\) = all physical objects
SAMEAS REPRESENTATION

➢ Limitation of queries

```sql
SELECT ?e ?n
WHERE {
  ?x rdf:type pDoc1
  ?x email ?e ← pDoc3
  ?x name ?n ← pDoc2
}
```

➢ Inference of properties

- Massive graph
- Complex manipulation (encoding, update...)
SAMEAS REPRESENTATION

➢ One representative clique selected
  ● Properties share the same identifier
  ● The other identifier is a reference

➢ Advantages
  ● More compact graphs
  ● Use of the dictionary for transformation
  ● Update of sameas values has no performance impact
SAMEAS REPRESENTATION

- No more query problems
  - Encoding using the identifier clique’s encoding

- Dictionary holding the sameAs identifiers

<table>
<thead>
<tr>
<th>id. value</th>
<th>concept value</th>
<th>concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>31</td>
<td>pDoc1</td>
</tr>
<tr>
<td>31</td>
<td>32</td>
<td>pDoc2</td>
</tr>
<tr>
<td>31</td>
<td>33</td>
<td>pDoc3</td>
</tr>
</tbody>
</table>

(virtual representation)
PARTIAL ENCODING

Stream:

_:x1 id "Q250"
_:x1 date 30/03/2017
_:x1 pressureMeasure _:x2
_:x2 value 4.5

LiteMat dictionary:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>40</td>
</tr>
<tr>
<td>hLocation</td>
<td>54</td>
</tr>
<tr>
<td>pMeasure</td>
<td>56</td>
</tr>
</tbody>
</table>

Partial encoding:

e1 40 i13

e1 date 30/03/2017

e1 56 m1

m1 value 4.5

➢ Some of the identifiers are not present in the static knowledge base
QUERY REWRITING

➢ Query reformulation:
  x  SELECT ?x WHERE { ?x rdf:type DUL:PhysicalObject . }
                            FILTER (?v >= 40 && ?v < 48) }

➢ Usage in WAVES:
  • Encoding of the static knowledge base and the queries
  • Partial encoding of streams

➢ Rewriting of classes, properties and sameAs
  • Using the identifiers from the static knowledge base
CONCLUSION
CONCLUSION

➢ Waves project
  • RDF Stream Processing engine
  • Use case: drinkable water network management

➢ Strider
  • Distributed RDF graph stream with reasoning
  • Support of RDFS and sameAs

➢ LiteMat
  • Compression with identifiers
  • Entity identifiers represent the ontology semantics
FUTURE WORKS

➢ Increase the expressivity of supported ontologies
  ● Transitive properties, inverseOf

➢ Improve partitioning
  ● of the dictionaries
  ● Improvement of FILTER on distributed streams
THANK YOU

QUESTIONS ?