SAMPLD

STRUCTURAL PROPERTIES AS PROXY FOR SEMANTIC RELEVANCE

LAURENS RIETVELD

HTTP://PRESENTATIONS.LAURENSRIETVELD.NL/NOW
QUANTITY OVER QUALITY

- Datasets become too large to run on commodity hardware
- We *use* only a small portion
- Can't we extract the part we are interested in?

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#triples</th>
<th>#queries</th>
<th>coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBpedia</td>
<td>459M</td>
<td>1640</td>
<td>0.003%</td>
</tr>
<tr>
<td>Linked Geo Data</td>
<td>289M</td>
<td>891</td>
<td>1.917%</td>
</tr>
<tr>
<td>MetaLex</td>
<td>204M</td>
<td>4933</td>
<td>0.016%</td>
</tr>
<tr>
<td>Open-BioMed</td>
<td>79M</td>
<td>931</td>
<td>0.011%</td>
</tr>
<tr>
<td>BIO2RDF (KEGG)</td>
<td>50M</td>
<td>1297</td>
<td>2.013%</td>
</tr>
<tr>
<td>Semantic Web Dog Food</td>
<td>0.24M</td>
<td>193</td>
<td>62.4%</td>
</tr>
</tbody>
</table>
RELEVANCE BASED SAMPLING

- Find the smallest possible RDF subgraph, that covers the maximum number of potential queries
- How can we determine which triples are relevant, and which are not?
- Can we implement a scalable sampling pipeline?
- Can we evaluate the results in a scalable fashion?
HOW TO DETERMINE RELEVANCE OF TRIPLES

INFORMED SAMPLING

• We know exactly which queries will be asked
• Extract those triples needed to answer the queries
• Problem: only a limited number of queries known

UNINFORMED SAMPLING

• We do not know which queries will be asked
• Use information contained in the graph to determine relevance
• Rank triples by relevance, and select the \( k \) best triples (\( 0 < k < \text{size of graph} \))
APPROACH

- Use the topology of the graph to determine relevance (network analysis)
- Evaluate the relevance of our samples against the queries that we do know
- Is network structure a good predictor for query answerability?
NETWORK ANALYSIS

- Example: Explain real-world phenomena
- Find central parts of the graph
- Betweenness Centrality
- Google PageRank

- We apply
  - In Degree
  - Out Degree
  - PageRank
RANKED LIST OF TRIPLES

- Rewrite

- Apply network analysis
- Nodes Weights $\rightarrow$ Triples Weights
  \[ W(\text{triple}) = \max(W(\text{sub}), W(\text{obj})) \]
- Weighted list of triples $\rightarrow$ Sample
EVALUATION

- Sample sizes: 1% - 99%
- Baselines:
  - Random Sample (10x)
  - Resource Frequency
NAIVE EVALUATION DOES NOT SCALE

\[ t_e(t_d) = \sum_{i=1}^{99} \frac{i}{100} \cdot \text{methods}_s \cdot \text{methods}_b \cdot t_d \]

• Over 15,000 datasets, and over 1.4 trillion triples
• Requirements
  ▪ Fast loading of samples
  ▪ Powerful hardware
• Not Scalable: load all triples, execute queries, and calculate recall
SCALABLE APPROACH

- Retrieve which triples are used by a query
- Use a hadoop cluster to find the weights of these triples
- Analyze whether these triples *would have been* included in the sample

- Scalable. Only execute each query *once*
EXAMPLE

QUERY

```
1 SELECT ?person ?country WHERE {
4 }
```

```?person ?country
:Laurens :NL
:Stefan  :Germany```

DATASET

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>:Amsterdam</td>
<td>:capitalOf</td>
<td>:NL</td>
<td>0.1</td>
</tr>
<tr>
<td>:Berlin</td>
<td>:capitalOf</td>
<td>:Germany</td>
<td>0.5</td>
</tr>
<tr>
<td>:Laurens</td>
<td>:bornIn</td>
<td>:Amsterdam</td>
<td>0.6</td>
</tr>
<tr>
<td>:Rinke</td>
<td>:bornIn</td>
<td>:Heerenveen</td>
<td>0.1</td>
</tr>
<tr>
<td>:Stefan</td>
<td>:bornIn</td>
<td>:Berlin</td>
<td>0.5</td>
</tr>
</tbody>
</table>
1. SELECT ?person ?country WHERE {
4. }

1. SELECT DISTINCT * WHERE {
4. }

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
<th>?person</th>
<th>?city</th>
<th>?country</th>
</tr>
</thead>
<tbody>
<tr>
<td>:Laurens</td>
<td>:bornIn</td>
<td>:Amsterdam</td>
<td>:Laurens</td>
<td>:Amsterdam</td>
<td>:NL</td>
</tr>
<tr>
<td>:Amsterdam</td>
<td>:capitalOf</td>
<td>:NL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>:Stefan</td>
<td>:bornIn</td>
<td>:Berlin</td>
<td>:Stefan</td>
<td>:Berlin</td>
<td>:Germany</td>
</tr>
<tr>
<td>:Berlin</td>
<td>:capitalOf</td>
<td>:Germany</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
TRIPLES → RECALL

WHICH ANSWERS WOULD WE GET WITH A SAMPLE OF 60%?

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>:Laurens</td>
<td>:bornIn</td>
<td>:Amsterdam</td>
<td>0.6</td>
</tr>
<tr>
<td>:Stefan</td>
<td>:bornIn</td>
<td>:Berlin</td>
<td>0.5</td>
</tr>
<tr>
<td>:Berlin</td>
<td>:capitalOf</td>
<td>:Germany</td>
<td>0.5</td>
</tr>
<tr>
<td>:Amsterdam</td>
<td>:capitalOf</td>
<td>:NL</td>
<td>0.1</td>
</tr>
<tr>
<td>:Rinke</td>
<td>:bornIn</td>
<td>:Heerenvien</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Triples used in query resultsets:

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>:Laurens</td>
<td>:bornIn</td>
<td>:Amsterdam</td>
</tr>
<tr>
<td>:Amsterdam</td>
<td>:capitalOf</td>
<td>:NL</td>
</tr>
<tr>
<td>:Stefan</td>
<td>:bornIn</td>
<td>:Berlin</td>
</tr>
<tr>
<td>:Berlin</td>
<td>:capitalOf</td>
<td>:Germany</td>
</tr>
</tbody>
</table>
EVALUATION

- Better specificity than regular recall
- Scalable: PIG instead of SPARQL
- Special cases, e.g. GROUP BY, LIMIT, DISTINCT, OPTIONAL, UNION
SPECIAL CASE: UNIONS

```sql
1 SELECT ?name WHERE {
2  {:laurens rdfs:label ?name} 
3  UNION 
4  {:laurens foaf:name ?name} 
5 }
```

Subject  Predicate  Object
:Laurens rdfs:label  "Laurens"
:Laurens foaf:name "Laurens"
## EVALUATION DATASETS

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#triples</th>
<th>#queries</th>
<th>coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBpedia</td>
<td>459M</td>
<td>1640</td>
<td>0.003%</td>
</tr>
<tr>
<td>Linked Geo Data</td>
<td>289M</td>
<td>891</td>
<td>1.917%</td>
</tr>
<tr>
<td>MetaLex</td>
<td>204M</td>
<td>4933</td>
<td>0.016%</td>
</tr>
<tr>
<td>Open-BioMed</td>
<td>79M</td>
<td>931</td>
<td>0.011%</td>
</tr>
<tr>
<td>BIO2RDF (KEGG)</td>
<td>50M</td>
<td>1297</td>
<td>2.013%</td>
</tr>
<tr>
<td>Semantic Web Dog Food</td>
<td>0.24M</td>
<td>193</td>
<td>62.4%</td>
</tr>
</tbody>
</table>
RESULTS

The graph shows the recall against sample size for various data sources and metrics. The x-axis represents the sample size, while the y-axis represents recall. Different datasets and metrics are indicated by different lines and colors:

- BIO2RDF - UniqueLiterals - Outdegree
- DBpedia - Path - Pagerank
- LinkedGeoData - UniqueLiterals - Outdegree
- Metalex - ResourceFrequency
- OpenBioMed - WithoutLiterals - Outdegree
- SemanticWebDogFood - Path - Pagerank
OBSERVATIONS

• The influence of a single triple
• DBpedia
  □ Good: Path + PageRank
  □ Bad: Path + Out Degree
  □ Queries: 2/3 require literals
• Other Observations
  □ # properties vs 'Context Literals' rewrite method
  □ # query triple patterns
CONCLUSION

- Scalable pipeline: network analysis algorithms + rewrite methods
- Able to eval over 15,000 datasets, and 1.4 trillion triples
- Number of query sets too limited to learn significant correlations

- Topology of the graphs *can* be used to determine *good* samples
- Mimic *semantic* relevance through *structural* properties, *without* an a-priori notion of relevance

Special thanks to Semantic Web Science Association (SWSA)