PageRank and Generic Entity Summarization for RDF Knowledge Bases

Dennis Diefenbach & Andreas Thalhammer

07.06.2018
Presentation at ESWC 2018
What is PageRank over RDF graphs?

An art draw drawn by Felipe Micaroni Lalli (micaroni@gmail.com).
What is Entity Summarization for RDF graphs?
Lake Garda (Q6414)

Lake in Italy
Lago di Garda

<table>
<thead>
<tr>
<th>Language</th>
<th>Label</th>
<th>Description</th>
<th>Also known as</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>Lake Garda</td>
<td>lake in Italy</td>
<td>Lago di Garda</td>
</tr>
</tbody>
</table>

**Statements**

- instance of lake
  - 1 reference

**Image**

Punta San Vigilio.jpg
1,023 × 691; 395 KB
1 reference
Welcome to the SummaClient!
This is an example webpage embedding the SummaClient!

Example
This is how the SummaClient looks like:

<table>
<thead>
<tr>
<th>Lake Garda</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>country</td>
<td>Italy</td>
</tr>
<tr>
<td>mountain range</td>
<td>Alps</td>
</tr>
<tr>
<td>watershed</td>
<td>Po basin</td>
</tr>
</tbody>
</table>
Outline

1. Relation between PageRank scores and Entity Summarization
2. PageRankRDF: a command line tool to compute PageRank over RDF graphs
3. SummaServer: a service to compute entity summaries
4. Use Case: Questions Answering
Outline

1. Relation between PageRank scores and Entity Summarization
2. PageRankRDF: a command line tool to compute PageRank over RDF graphs
3. SummaServer: a service to compute entity summaries
4. Use Case: Questions Answering
Existing idea presented by Andreas Thalhammer (working on DBpedia)
Step1: How to compute entity summaries?

Offline phase (compute PageRank scores)

1. Extract the link structure of Wikipedia, i.e. which page is linked to which page and consider it as a graph.

2. Compute the PageRank scores of each page

Note: Each page in Wikipedia corresponds to an entity in DBpedia. So we have a PageRank score for each entity in DBpedia also.
Step 2: How to compute entity summaries?

**Online phase** (compute summaries)

1. For a given resource \( R \) find all resources \( R_1, \ldots, R_N \) at distance 1 in the RDF graphs
2. Between \( R_1, \ldots, R_N \) select the \( k \) resources with the highest PageRank scores, i.e. \( S_1, \ldots, S_K \)
3. Show in the summary the \( k \) facts that connect \( R \) with \( S_1, \ldots, S_K \).
Idea: extend this to generic RDF graphs and publish source code, datasets, and web services for PageRank computations and Entity Summarization.
Outline

1. Relation between PageRank scores and Entity Summarization
2. PageRankRDF: a command line tool to compute PageRank over RDF graphs
3. SummaServer: a service to compute entity summaries
4. Use Case: Questions Answering
PageRank over RDF graphs, is this difficult? It is not difficult to implement the algorithm, it is difficult to implement it in a time and memory efficient way.
Observation: the URIs are not needed for PageRank computations, only the graph structure.
Idea: use HDT as a data-structure since it separates the URIs from the graph structure.

- Indexing comes for free
- Compression is built in
Usage: java -jar pagerank.jar -in file -out pagerank [options]
Options:
   --damping, -D
   specifying the damping factor for the PageRank computation
   Default: 0.85
   --format, -f
   specify the output format for the PageRank scores, either "tsv" or "nt"
   Default: nt
   --help
   displays the list of possible parameters
* --input, -in
   specify a file in some RDF format or in HDT
   --iteration, -I
   specifying the number are performed by PageRank
   Default: 40
* --output, -out
   specify the file where the pagerank scores are stored
   --start-value, -S
   specifying the start value for the PageRank computation
   Default: 0.1

Code: https://github.com/WDAqua/PageRankRDF
Experiment

- Compare a straightforward implementation with one using HDT.
- Dataset: Wikidata dump of the 28 September 2017 (Size of 237 Gb in ntriples, 2.2 billion triples)
- Server: Intel(R) Xeon(R) CPU E5-2667 v3 @ 3.20GHz and 94Gb of RAM
Results

Computation time

Maximal memory consumption

# Minutes

# Gb

Non-HDT | HDT | Non-HDT | HDT

conversion | index | pagerank | conversion | index | pagerank

0 | 0 | 800 | 397 | 90 | 60

0 | 0 | 462 | 20 | 18
Question: What is the quantitative difference between PageRank scores over RDF relations vs. Wikipedia links?
We compare:

- Wikidata, PageRankRDF: 38 433 113 Q-IDs (total 80 646 048 resources)
- Wikidata, danker (Wikipedia links): 17 645 575 Q-IDs
- SubjectiveEye3D: 6 211 717 Q-IDs

Note: PageRankRDF \( \cap \) danker \( \cap \) SubjectiveEye3D: 4 253 903 Q-IDs
**Table 1.** Spearman’s $\rho$ / Kendall’s $\tau$ correlations of PageRank on RDF relations vs. Wikipedia links (via danker) and the comparison to SubjectiveEye3D.

<table>
<thead>
<tr>
<th></th>
<th>PageRankRDF</th>
<th>danker</th>
<th>SubjectiveEye3D</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PageRankRDF</strong></td>
<td>1.000 / 1.000</td>
<td>0.427 / 0.328</td>
<td>0.184 / 0.138</td>
</tr>
<tr>
<td><strong>danker</strong></td>
<td>0.427 / 0.328</td>
<td>1.000 / 1.000</td>
<td>0.400 / 0.276</td>
</tr>
<tr>
<td><strong>SubjectiveEye3D</strong></td>
<td>0.184 / 0.138</td>
<td>0.400 / 0.276</td>
<td>1.000 / 1.000</td>
</tr>
</tbody>
</table>
Observation:

- All PageRank measures correlate positively with click-based ranking (SubjectiveEye3D).
- T-Box elements receive extremely high scores with PageRankRDF.
- This result should be taken with care: click-based measures are not necessarily a gold standard.
Resources

R1 PageRankRDF: a command line tool to compute PageRank over RDF graphs, available at: https://github.com/WDAqua/PageRankRDF

PageRank scores for:

- R1.1 DBLP
- R1.2 DBpedia
- R1.3 Freebase
- R1.4 MusicBrainz
- R1.5 Scigraph
- R1.6 Wikidata

Available at: https://figshare.com/account/home#/projects/28119
Outline

1. Relation between PageRank scores and Entity Summarization
2. PageRankRDF: a command line tool to compute PageRank over RDF graphs
3. SummaServer: a service to compute entity summaries
4. Use Case: Questions Answering
Welcome to the SummaServer!

A summarization service for RDF Knowledge Bases

About

You have a Knowledge Base and you want to retrieve the top-k most relevant facts about an entity? Then you are at the right place! The SummaServer does the job for you. It currently supports the following KBs:

- freebase
- dblp
- musicbrainz
SummaServer API
SummaServer API has two components:

- SUMMA Vocabulary
- RESTful interaction mechanism
SUMMA Vocabulary
RESTful interaction mechanism

Client


201 CREATED
Location: http://example.com/summary?entity=dbpedia:Barack_Obama&topK=10
@ prefix summa: <http://purl.org/voc/summa/> .
...

Server

GET http://example.com/summary?entity=dbpedia:Barack_Obama&topK=10

200 OK
@ prefix summa: <http://purl.org/voc/summa/> .
...

@ prefix summa: <http://purl.org/voc/summa/> .
How to make the SummaServer work on a new dataset D?

1. Compute PageRank scores for D and save them using the vrank vocabulary into a SPARQL endpoint with the dataset D.
2. Implement 3 SPARQL queries.
Listing  QUERY 1: This query retrieves for an ENTITY the corresponding label in the language LANG. For Wikidata the query is

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?l
WHERE {
  <ENTITY> rdfs:label ?l .
  FILTER regex(lang(?l), "LANG", "i") .
}
```
Listing  QUERY 2: This query must retrieve the resources connected to the resource ENTITY, order them according to the PageRank score and take the first TOPK. Moreover it retrieves the labels of the founded resources in the language LANG.

```
PREFIX rdf: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX vrank: <http://purl.org/voc/vrank#>
PREFIX wdd: <http://www.wikidata.org/prop/direct/>

SELECT DISTINCT ?o ?l ?pageRank
WHERE {
    <ENTITY> ?p ?o .
    FILTER (?p != rdf:type && ?p != wdd:P31
        && ?p != wdd:P735 && wdd:P21
        && ?p != wdd:P1343 )
    ?o rdfs:label ?l .
    regex(lang(?l), "LANG", "i") .
    graph <http://wikidata.com/pageRank> {
    }
}
ORDER BY DESC (?pageRank) LIMIT TOPK
```
Listing

QUERY 3: This query must retrieve given two resource, ENTITY and OBJECT, the label of the property between them in the language LANG. For Wikidata we use the following query:

```
PREFIX rdf: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX vrank:<http://purl.org/voc/vrank#>
SELECT ?p ?l 
WHERE {
  <ENTITY> ?p <OBJECT> . 
  OPTIONAL {
    ?o rdfs:label ?l . 
    FILTER regex(lang(?l), "LANG", "i") 
  }
}
ORDER BY asc(?p) LIMIT 1
```
Resources

R2
SummaServer: a service for Entity Summarization, available at
available at: https://github.com/WDAqua/SummaServer

- R2.1 https://wdaqua-summa-server.univ-st-etienne.fr/dblp/sum
- R2.2 https://wdaqua-summa-server.univ-st-etienne.fr/dbpedia/sum
- R2.3 https://wdaqua-summa-server.univ-st-etienne.fr/freebase/sum
- R2.4 https://wdaqua-summa-server.univ-st-etienne.fr/musicbrainz/sum
- R2.5 https://wdaqua-summa-server.univ-st-etienne.fr/scigraph/sum
- R2.6 https://wdaqua-summa-server.univ-st-etienne.fr/wikidata/sum
Outline

1. Relation between PageRank scores and Entity Summarization
2. PageRankRDF: a command line tool to compute PageRank over RDF graphs
3. SummaServer: a service to compute entity summaries
4. Use Case: Questions Answering
PageRank scores used in QA systems.
A river is a natural flowing watercourse, usually freshwater, flowing towards an ocean, sea, lake or another river. In some cases a river flows into the ground and becomes dry at the end of its course without reaching another body of water. Small rivers can be referred to using names such as stream, creek, brook, rivulet, and rill. There are no official definitions for the generic term river as applied to geographic features, although in some countries or communities a stream is defined by its size. Many names for small rivers are specific to geographic location; examples are "run" in some parts of the United States, "burn" in Scotland and northeast England, and "beck" in northern England. Sometimes a river is defined as being larger than a creek, but not always: the language is vague. Rivers are part of the hydrological cycle. Water generally collects in a river from precipitation through a drainage basin from surface runoff and other sources such as groundwater recharge, springs, and the release of stored water in natural ice and snowpacks (e.g., from glaciers). Potamology is the scientific study of rivers, while limnology is the study of inland waters in general.
Did you mean lakes in the Po basin?
Entity summarization in QA systems.
Conclusion

Contributions:

• **R1** PageRankRDF: a command line tool to compute PageRank over RDF graphs
• **R2** SummaServer: a service to compute entity summaries
• Usage of these resources in the Question Answering domain
Further reading

Literature:


Weblinks:

- http://entitysummarization.org
- people.aifb.kit.edu/ath/
- http://km.aifb.kit.edu/services/link/
Acknowledgements

The European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 642795, project: Answering Questions using Web Data (WDAqua).
Thank you for your attention!