

Towards a Ranking Model for Semantic Layers over Digital Archives

Pavlos Fafalios, Vaibhav Kasturia, Wolfgang Nejdl

L3S Research Center, University of Hannover, Germany

{fafalios,kasturia,nejdl}@l3s.de

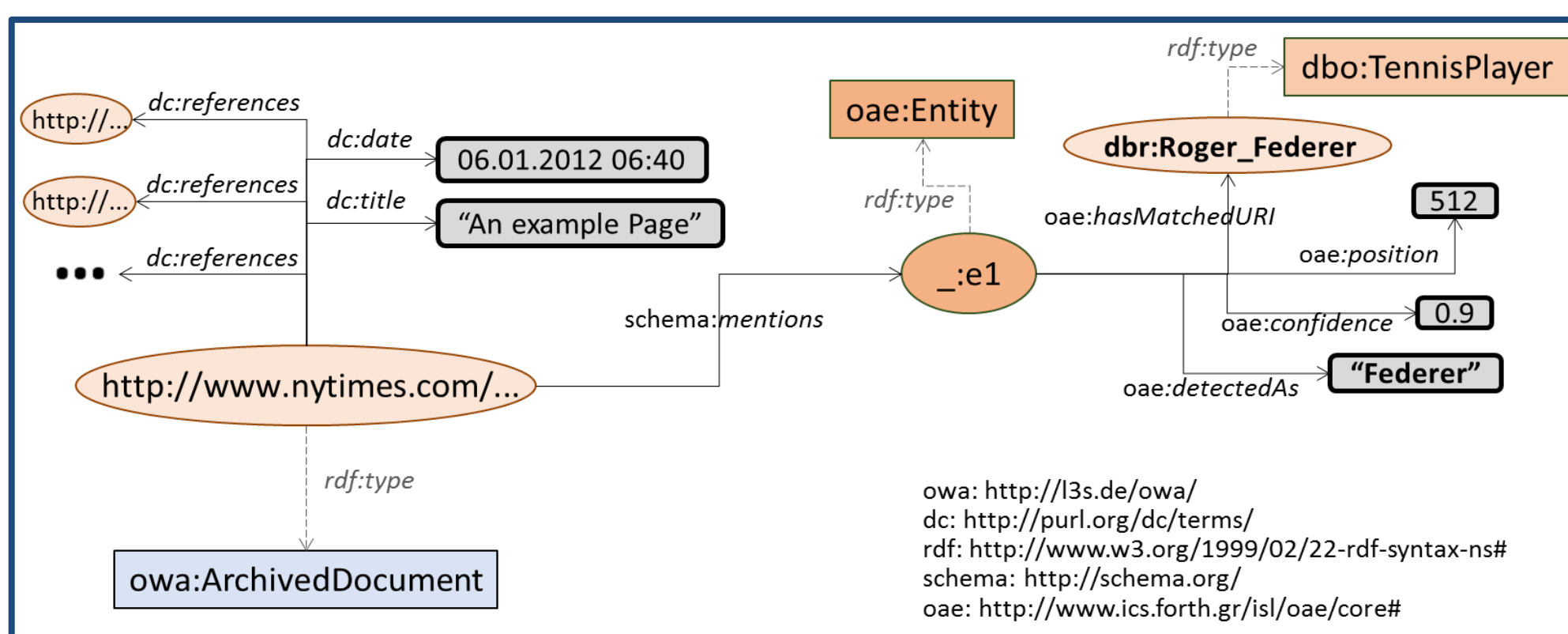
1. Motivation

- ❖ How to explore archives in a more **advanced** and **exploratory** way?
 - Find documents discussing about a specific category of entities (e.g., philanthropists), or about entities sharing some characteristics (e.g., born in Germany before 1960)?
- ❖ How to explore archives by integrating information from existing knowledge bases, like DBpedia?

2. Semantic Layer

- ❖ RDF repository describing **metadata** and **annotation** information for a collection of archived documents.
 - Allows running advanced, entity-centric SPARQL queries that combine metadata of the documents (e.g., publication date) and semantic information (e.g., mentioned entities)
 - More at: Fafalios et al., "Building and Querying Semantic Layers for Web Archives", JCDL'17

- ❖ Example for a news article:



- ❖ Example SPARQL queries over Semantic Layers

```
SELECT DISTINCT ?article WHERE {  
  ?article dc:date ?date FILTER(year(?date) = 1990) . "AND" (conjunctive) semantics  
  ?article schema:mentions ?entity1, ?entity2 .  
  ?entity1 oae:hasMatchedURI dbr:Nelson_Mandela .  
  ?entity2 oae:hasMatchedURI dbr:F_W_de_Klerk }
```

Retrieve articles of 1990 discussing about Nelson Mandela and F. W. de Klerk

```
SELECT DISTINCT ?article WHERE {  
  ?article dc:date ?date FILTER(year(?date) = 1990) . "OR" (disjunctive) semantics  
  ?article schema:mentions ?entity .  
  ?entity oae:hasMatchedURI ?entURI .  
  ?entURI dc:subject dbc:State_Presidents_of_South_Africa }
```

Retrieve articles of 1990 discussing about state presidents of South Africa

3. The problem

- ❖ The results returned by a SPARQL query:
 - can be numerous
 - all equally match the query
- ❖ How to rank them for identifying and promoting the most important ones?
 - What makes an archived document important for a given query?

4. Related Work

- ❖ **Ranking of archived documents** (for free-text queries)
 - Time-aware Retrieval and Ranking [Kanhabua and Anand, 2016]
 - Tempas [Holzmann and Anand, 2016], HistDiv [Singh et al., 2016]
 - Works by Kanhabua et al. (2016), Vo et al. (2016)
- ❖ **Ranking in knowledge graphs**
 - Learning to rank for RDF entity search [Dali et al., 2012]
 - Swoogle [Ding et al., 2005], SemRank [Anyanwu et al., 2005]
 - NAGA [Kasneji et al., 2008], DING [Delbru et al., 2010],
 - ReconRank [Hogan et al., 2006], Noc-order [Graves et al., 2008]
- ❖ **Our approach:** Ranking archived documents for structured queries in knowledge graphs
 - Availability of metadata and entity annotations
 - No access to full contents!

5. Problem Definition

- ❖ **Ranking Documents for Structured Queries over Semantic Layers**
 - Consider a **semantic layer** over a collection of **archived documents D** published within a set of **time periods T** of fixed granularity (e.g., day), and a set of **entities E** mentioned in documents of D.
 - Given a **SPARQL query Q** requesting documents from D published within a **time period T_Q ⊆ T** and related to one or more **Entities of Interest (EoI) E_Q ⊆ E** with logical AND (mentioning all EoI) or OR (mentioning at least one EoI) semantics, the **problem** is how to rank the returned documents **D_Q ⊆ D** that match Q.

6. Towards a Ranking Model

- ❖ What makes an archived document **important** for one or more entities of interest (EoI)?
 - **Relativeness:** the document should talk about the EoI (as its main topic)
 - **Timeliness:** the document should have been published in an important (for the EoI) time period
 - **Relatedness:** the document should discuss the relation of the EoI with other important (for the EoI) entities

- ❖ **Relativeness** (of a document d)

- Consider the frequency of the EoI in d

$$Score_{D \wedge}(d) = \frac{\sum_{e \in E_Q} count(e, d)}{\sum_{e' \in E_d} count(e', d)} \quad Score_{D \vee}(d) = \frac{\sum_{e \in E_Q} count(e, d)}{\sum_{e' \in E_d} count(e', d)} \cdot \frac{|E_d \cap E_Q|}{|E_Q|}$$

"AND" (conjunctive) semantics "OR" (disjunctive) semantics

- ❖ **Timeliness** (of a time period p)

- Consider the number of documents mentioning the EoI during p

$$Score_P(p) = \frac{|D_p \cap D_Q|}{|D_Q|}$$

- ❖ **Relatedness** (of an entity e to the EoI)

- Consider the number of co-occurrences of e with the EoI in important time periods
- Avoid over-emphasizing common and general entities

$$Score_E(e) = idf(e) \cdot \sum_{p \in P_Q} (Score_P(p) \cdot \frac{|D_{e,p} \cap D_Q|}{|D_p \cap D_Q|}) \quad idf(e) = 1 - \frac{|D_e \cap (\cup_{e' \in E_Q} D_{e'})|}{|\cup_{e' \in E_Q} D_{e'}|}$$

- ❖ **Joining the models:** $S(d) = Score_P(p_d) \cdot Score_D(d) + \beta \frac{\sum_{e \in E_d \setminus E_Q} Score_E(e)}{|E_d|}$

7. Next Steps

- ❖ Create a ground truth for the problem at hand
- ❖ Evaluate our baseline ranking model and the effect of each component
- ❖ Define and evaluate more advanced models (learning to rank, stochastic, etc.)
- ❖ Investigate the case of web archives (where documents have versions and publication dates are not usually available)