Ranking Archived Documents for Structured Queries on Semantic Layers

Pavlos Fafalios, Vaibhav Kasturia, Wolfgang Nejdl

fafalios@L3S.de

L3S Research Center

University of Hannover, Germany

Document Archives

- Newspaper archives
- Web archives
- Social Media archives
- …

- Valuable sources for research in many disciplines

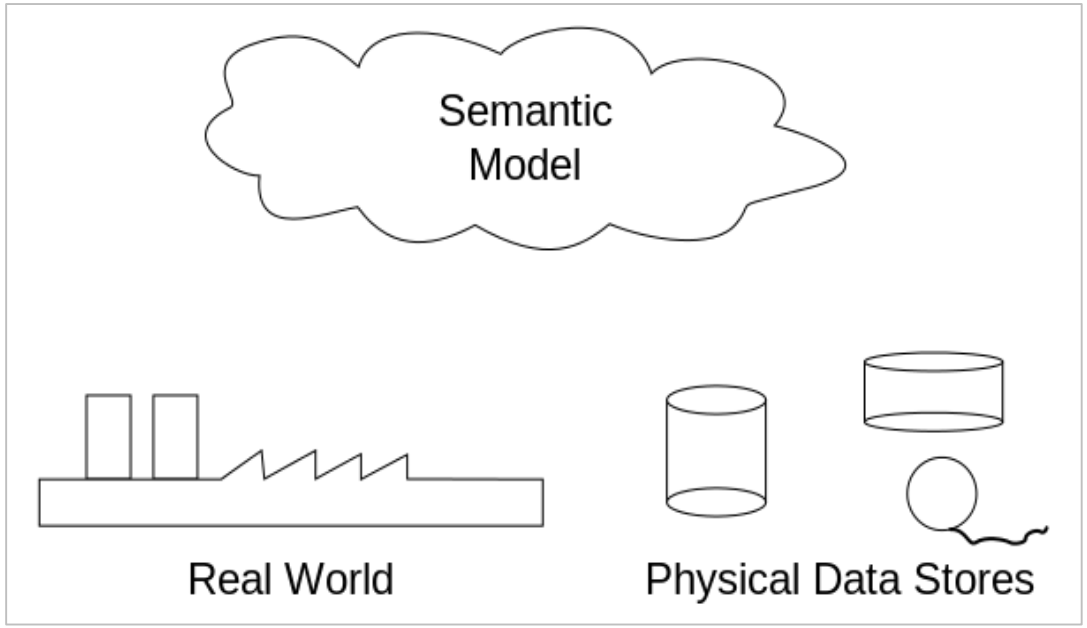
Semantic Models for document archives

- Semantic Layer

P. Fafalios, H. Holzmann, V. Kasturia, and W. Nejdl,
"Buiding and Querying Semantic Layers for Web Archives", JCDL 2017

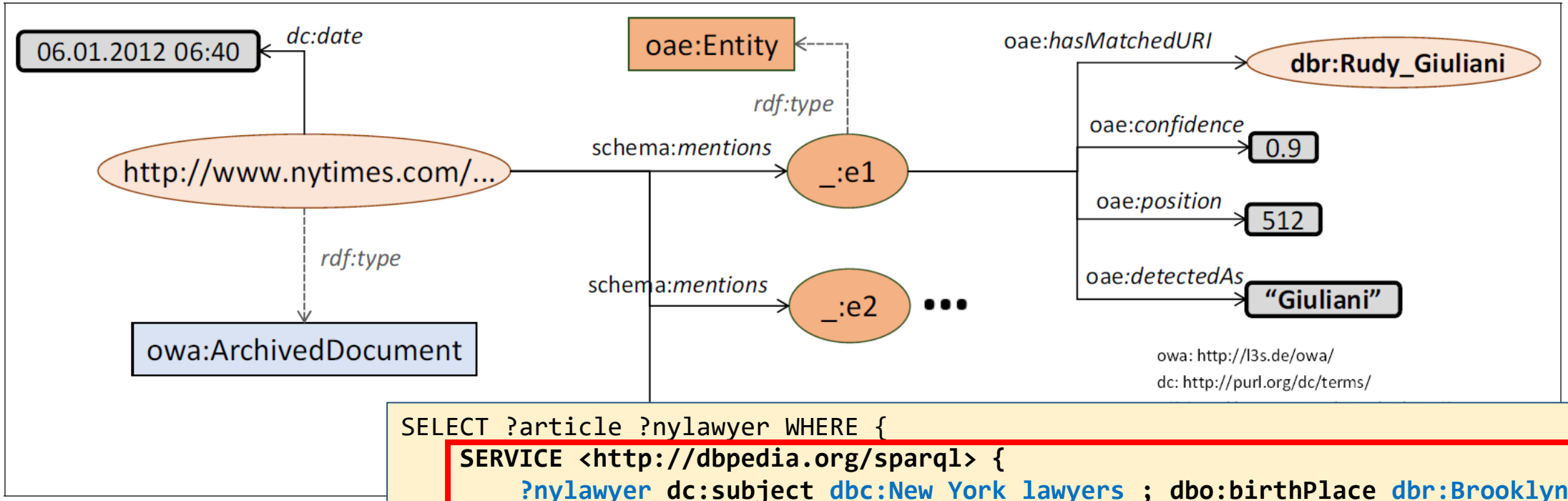
  - Structured data (RDF triples) describing metadata (e.g., publication date) and content-based information (e.g., mentioned entities) about the archived documents

- Benefits:
  - Advanced (entity-centric) query capabilities
  - Real-time data integration



Semantic Layer

P. Fafalios, H. Holzmann, V. Kasturia, and W. Nejdl,
"Buiding and Querying Semantic Layers for Web Archives", JCDL 2017




```
SELECT ?article ?nylawyer WHERE {
    SERVICE <http://dbpedia.org/sparql> {
        ?nylawyer dc:subject dbc:New_York_lawyers ; dbo:birthPlace dbr:Brooklyn }
    ?article dc:date ?date FILTER(year(?date) = 1989) .
    ?article schema:mentions ?entity . ?entity oae:hasMatchedURI ?nylawyer }
```

"Find articles of 1989 mentioning lawyers of New York born in Brooklyn"

The problem

- The query results can be numerous
- All results equally match the query
  - There is no relevance ranking!

Result: **184** articles mentioning **44** distinct lawyers

```
SELECT ?article ?nylawyer WHERE {
    SERVICE <http://dbpedia.org/sparql> {
        ?nylawyer dc:subject dbc:New_York_lawyers ; dbo:birthPlace dbr:Brooklyn }
    ?article dc:date ?date FILTER(year(?date) = 1989) .
    ?article schema:mentions ?entity . ?entity oae:hasMatchedURI ?nylawyer }
```

- **How to rank the results?**
  - Discover (and show to the user) the most "**important**" documents

Contributions

- We introduce and formalize the problem of ***"ranking archived documents for structured queries on semantic layers"***

- We propose two **ranking models** for the problem at hand:
  - Probabilistic
  - Stochastic (Random Walk with restart)

- New **ground truth** dataset
  - News archive (NYT articles)
  - Queries – results – relevance judgements (+justifications)
  - Publicly available!

- **Evaluation** results
  - Effectiveness
  - Interesting findings

Related Work - **Temporal Information Retrieval**

- Surveys: Campos et al. 2015, Kanhabua et al. 2015
- Time-aware document ranking
  - Leveraging temporal expressions (Arikan et al. 2009)
  - Time-aware language model (Berberich et al. 2010)
  - Answering time-sensitive queries (Metzlere et al. 2009) (Dakka et al. 2012)
  - Diversity-aware (Singh et al. 2016)
  - Temporal archive search based on anchor texts (Holzmann et al. 2017)

<u>Difference of our case:</u>

- No access on the full contents of the documents
- Structured SPARQL queries, not keywords

Related Work - **Ranking in Knowledge Graphs**

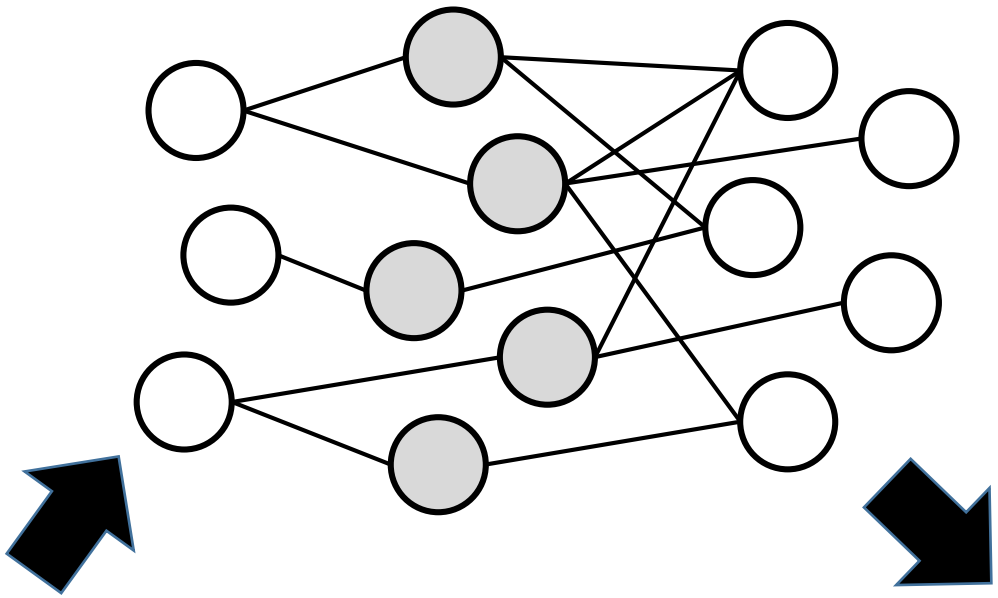
- Survey: Roa-Valverde and Sicilia, 2014
- Ad-hoc object retrieval (Pound et al. 2010) (Tonon et al. 2012)
  - SemSearch challenge
  - TREC entity track

<u>Difference of our case:</u>

- The units of retrieval are textual (unstructured) documents
- Semantic Layer: special kind of Knowledge Graph containing metadata and entity annotations about a collection of textual documents

Problem Definition

SEMANTIC LAYER



SPARQL QUERY

```
SELECT ?article ?nylawyer WHERE {
    SERVICE <http://dbpedia.org/sparql> {
        ?nylawyer dc:subject dbc:New_York_lawyers ;
           dbo:birthPlace dbr:Brooklyn }
    ?article dc:date ?date FILTER(year(?date) = 1989) .
    ?article schema:mentions ?entity .
    ?entity oae:hasMatchedURI ?nylawyer }
```

The query requests documents:

- Published within a specific **time period**, and
- Mentioning one or more **entities of interest** (query entities)

RESULTS

- Doc_a
- Doc_b
- Doc_c
- Doc_d
- …

No ranking!

RANKED RESULTS

Ranking Model

1. **Doc_y**
2. **Doc_d**
3. **Doc_m**
4. **Doc_n**
5. **…**

Query cases

- Logical “AND” semantics:
  - The documents mention **ALL** the query entities

```
SELECT DISTINCT ?article WHERE {
  ?article dc:date ?date FILTER(year(?date) = 1990) .
  ?article oae:mentions ?entity1, ?entity2 .
  ?entity1 oae:hasMatchedURI  dbr:Nelson_Mandela .
  ?entity2 oae:hasMatchedURI  dbr:F._W._de_Klerk }
```

*“Find articles of 1990 mentioning both Nelson Mandeal **and** F. W. de Klerk”*

- Logical “OR” semantics
  - The documents mention **AS LEAST ONE** of the query entities

```
SELECT DISTINCT ?article WHERE {
  SERVICE <http://dbpedia.org/sparql> {
    ?president dc:subject dbc:State_Presidents_of_South_Africa }
  ?article dc:date ?date FILTER(year(?date) = 1990) .
  ?article oae:mentions ?entity .
  ?entity oae:hasMatchedURI  ?president }
```

“Find articles of 1990 mentioning state presidents of South Africa”

Approach

- What makes an archived document **important** given a **time period** and one or more **entities of interest?**

- Requirements:
  - Make use of the least amount of document metadata → widely applicable
  - Exploit only the contents of the semantic layer

- Aspects affecting the importance of a document to a query:
  - The **relativeness** of a document wrt the query entities
    - the document should talk about the query entities, ideally as its main topic
  - The **timeliness** of a document wrt its publication date
    - the document should have been published during an important (for the query entities) time period
  - The **relatedness** of a document wrt other important entities mentioned in it
    - The document should discuss the relation of the query entities with other important entities

Probabilistic Modeling

- Relativeness
  - **The probability to pick a document based (only) on the query entities it mentions**
  - Consider the frequency of the query entities in the document
  - Motivation: frequency is a classic numerical statistic reflecting how important a term (entity) is to a document (Leskovec et al. 2014)

$$P(d|E_Q) = \frac{score^f(d, E_Q)}{\sum_{d' \in D_Q} score^f(d', E_Q)}$$

AND semantics

$$score^f_{\wedge}(d, E_Q) = \frac{\sum_{e \in E_Q} count(e, d)}{\sum_{e' \in ents(d)} count(e', d)}$$

OR semantics

$$score^f_{\vee}(d, E_Q) = \frac{\sum_{e \in E_Q} count(e, d)}{\sum_{e' \in ents(d)} count(e', d)} \cdot \frac{|ents(d) \cap E_Q|}{|E_Q|}$$

Percentage of query entities discussed in the document

Probabilistic Modeling

- Timeliness
  - **The probability to pick a document based (only) on its publication date**
  - Favor time periods with large number of documents discussing about the query entities
  - Motivation: Considering the fraction of documents mentioning the query entities in a given time period can improve document retrieval (Singh et al. 2016)

$$P(d|t_d) = \frac{score^t(t_d)}{\sum_{d' \in D_Q} score^t(t_{d'})}$$

AND semantics

$$score^t_{\wedge}(t) = \frac{|docs(t) \cap D_Q|}{|D_Q|}$$

OR semantics

$$score^t_{\vee}(t) = \frac{|docs(t) \cap D_Q|}{|D_Q|} \cdot N(E_Q, t)$$

$$N(E_Q, t) = \frac{\sum_{d \in docs(t) \cap D_Q} \frac{|ents(d) \cap E_Q|}{|E_Q|}}{|docs(t) \cap D_Q|}$$

Average percentage of query entities mentioned in documents during t

Probabilistic Modeling

- Relatedness
  - **The probability to pick a document based (only) on other entities mentioned in it**
  - Favor entities that are co-mentioned frequently with the query entities in important time periods
  - Motivation: The co-occurrence of entities in documents of a specific time period is a strong indicator of their temporal relatedness (Zhang et al. 2016)

$$P(d|E_{D_Q}) = \frac{\sum_{e \in ents(d) \setminus E_Q} score^r(e)}{\sum_{d' \in D_Q} \sum_{e' \in ents(d') \setminus E_Q} score^r(e')}$$

AND semantics

$$score^r_{\wedge}(e) = idf_{\wedge}(e) \cdot \sum_{t \in T_Q} (score^t_{\wedge}(t) \cdot \frac{|docs(t) \cap D_Q \cap docs(e)|}{|docs(t) \cap D_Q|})$$

$$idf_{\wedge}(e) = 1 - \frac{|docs(e) \cap (\cap_{e' \in E_Q} docs(e'))|}{|\cap_{e' \in E_Q} docs(e')|}$$

Inverse document frequency of e (considering the entire corpus and documents that contain all the query entities)

OR semantics

$$score^r_{\vee}(e) = idf_{\vee}(e)\, N(E_Q, e) \sum_{t \in T_Q} (score^t_{\vee}(t) \frac{|docs(t) \cap D_Q \cap docs(e)|}{|docs(t) \cap D_Q|})$$

$$idf_{\vee}(e) = 1 - \frac{|docs(e) \cap (\cup_{e' \in E_Q} docs(e'))|}{|\cup_{e' \in E_Q} docs(e')|}$$

Inverse document frequency of e (considering the entire corpus and documents that contain at least one of the query entities)

$$N(E_Q, e) = \frac{\sum_{d \in docs(e) \cap D_Q} \frac{|ents(d) \cap E_Q|}{|E_Q|}}{|docs(e) \cap D_Q|}$$

Average percentage of query entities mentioned in the documents together with e during t

Probabilistic Modeling

- Joining the models
  - Combine the different models in a single probability score

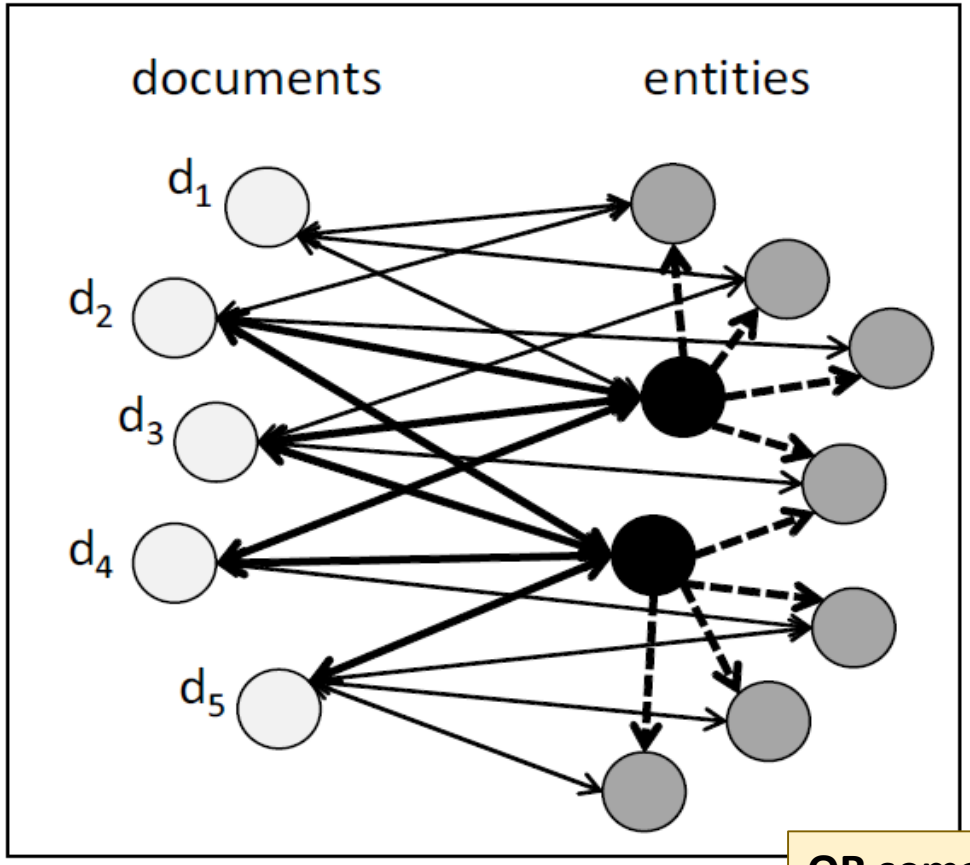
$$P(d|E_Q, t_d, E_{D_Q}) = \frac{P(d|E_Q)P(d|T_Q)P(d|E_{D_Q})}{\sum_{d' \in D_Q} P(d'|E_Q)P(d'|T_Q)P(d'|E_{D_Q})}$$

Stochastic Modeling

- Random Walk on the graph (**Markov chain**) defined by:
  - the **query entities**
  - the returned **documents**
  - the **entities** mentioned in the documents
- Stochastic analysis
  - Biased PageRank-like algorithm

- Why?
  - It can be easily customized (“biased”) for different types of applications
  - Allows understanding how the different aspects of the model affect the rankings

Stochastic Modeling - The State Transition Graph



OR semantics

- Start from a query entity:
  - Move to a doc mentioning it, or
  - Move to a related entity (co-mentioned in docs)
- From a doc:
  - Move to an entity mentioned in the doc
- From a related (no query) entity:
  - Move to a doc mentioning the entity

Stochastic Modeling - The Transition Probabilities

- From **query entities** to **documents**
  - Relativeness + Timeliness

$$weight(e \rightarrow d) = \frac{score^f(d, E_Q) \cdot score^t(t_d)}{\sum_{d' \in docs(e) \cap D_Q} (score^f(d', E_Q) \cdot score^t(t_{d'}))}$$

- From **query entities** to related **entities** (co-mentioned in docs)
  - Relatedness

$$weight(e \rightarrow e') = \frac{score^r(e')}{\sum_{e'' \in out(e)} (score^r(e''))}$$

- From **query entities** to another node (**document** or **related entity**)

$$weight(e \rightarrow n) = \begin{cases} p_1 \cdot weight(e \rightarrow d) & n \in D_Q \\ (1 - p_1) \cdot weight(e \rightarrow e') & n \in E_{D_Q} \end{cases}$$

p_1: probability to move to a document node

Stochastic Modeling - The Transition Probabilities

- From **documents** to **entities**
  - Entity frequency

$$weight(d \rightarrow e) = \frac{count(e,d)}{\sum_{e' \in ents(d)}(count(e',d))}$$

- From **no-query entities** to **documents**
  - Entity frequency

$$weight(e \rightarrow d) = \frac{count(e,d)}{\sum_{d' \in docs(e) \cap D_Q}(count(e,d')}$$

Stochastic Modeling - The Stochastic Analysis

- Random Walk with Restart:
  - Start from the query entities and either **follow an edge** or **jump back** to a query entity and start again the traversal ("restart")

$$r(n) = d \cdot Jump(n) + (1-d) \cdot \sum_{n' \in in(n)} (weight(n' \rightarrow n) \cdot r(n'))$$

Probability to perform a restart

$$Jump(n) = \begin{cases} 1/|E_Q| & n \in E_Q \\ 0 & n \notin E_Q \end{cases}$$

Probability to jump to node "n"

- Initial node scores:
  - Query entities: $1 / |E_Q|$
  - All other: zero
- Iteratively run to convergence

Evaluation - Ground Truth dataset

- New ground truth dataset! (due to lack of benchmark datasets for our problem)
- Semantic Layer:
  - Corpus: **New York Times**
  - Entity Linking: **Babelfy**

- 24 SPARQL queries:
  - 6 single-entity queries (Q1-Q6)
    - E.g.: articles of 1990 discussing about Nelson Mandela
  - 6 multiple-entity AND queries (Q7-Q12)
    - E.g.: articles of 1990 discussing about Nelson Mandela **and** F. W. de Klerk
  - 6 multiple-entity OR queries (Q13-Q18)
    - E.g.: articles of 1990 discussing about Nelson Mandela **or** F. W. de Klerk
  - 6 category queries (Q19-Q24)
    - E.g., articles of 1990 discussing about philanthropists

Evaluation - Ground Truth dataset

- Manual evaluation of all results (**773 totally**)
  - **Score 0:** The document has almost nothing to do with the query entities
  - **Score 1:** The topic of the document is not about the query entities, however the query entities are related to the document context
  - **Score 2:** The topic of the document is not about the query entities, however the query entities are important for the document context
  - **Score 3:** The topic of the document is about the query entities and discusses something important about them

Publicly available:

http://l3s.de/~fafalios/jcdl/evaluation_dataset.zip

(semantic layer, queries, results, relevance scores + explanations)

Query	Total number of results	Score 0 results	Score 1 results	Score 2 results	Score 3 results
1	65	13	13	4	35
2	28	20	5	1	2
3	28	24	2	0	2
4	23	15	3	1	4
5	38	18	6	7	7
6	24	16	4	1	3
7	29	13	6	8	2
8	61	10	27	15	9
9	42	14	10	10	8
10	28	17	4	6	1
11	27	13	10	4	0
12	24	13	2	8	1
13	30	21	1	4	4
14	37	15	8	8	6
15	27	18	5	2	2
16	23	5	14	2	2
17	21	13	2	2	4
18	25	6	4	9	6
19	41	24	4	8	5
20	22	11	8	3	0
21	29	16	5	5	3
22	47	33	7	3	4
23	31	18	3	8	2
24	23	14	3	3	3

Evaluation results

- Probabilistic modeling – all queries

Measure	Random ranking	Relativeness [A]	Timeliness [B]	Relatedness [C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.26	0.48	0.27	0.41	0.53	0.52	0.45	**0.56**
NDCG@10	0.33	0.52	0.36	0.50	0.55	0.56	0.51	**0.58**
NDCG@all	0.68	0.79	0.69	0.76	0.80	0.81	0.76	**0.82**
P@5	0.27	0.44	0.28	0.48	0.48	0.50	0.45	**0.52**
P@10	0.29	0.38	0.30	**0.45**	0.37	0.42	0.41	0.41

Evaluation results

[A]: relativeness
[B]: timeliness
[C]: relatedness

- Probabilistic modeling

Single-entity queries

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.66	0.30	0.40	0.68	0.69	0.45	**0.70**
NDCG@10	0.69	0.38	0.51	0.70	**0.71**	0.51	0.70
NDCG@all	**0.88**	0.67	0.75	0.87	**0.88**	0.72	0.87
P@5	0.57	0.23	0.50	0.57	**0.60**	0.40	**0.60**
P@10	0.40	0.27	**0.45**	0.40	0.40	0.38	0.40

Multiple-entity AND queries

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.34	0.28	0.31	**0.42**	0.35	0.38	0.40
NDCG@10	0.43	0.33	0.40	0.46	0.45	0.46	**0.47**
NDCG@all	0.76	0.71	0.75	**0.78**	0.76	0.77	0.77
P@5	0.43	0.33	0.30	**0.50**	0.47	0.43	**0.50**
P@10	0.50	0.32	0.42	0.48	**0.53**	0.47	0.50

Multiple-entity OR queries

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.68	0.24	0.44	0.70	**0.72**	0.46	0.71
NDCG@10	0.69	0.36	0.55	0.69	0.72	0.54	**0.73**
NDCG@all	0.87	0.69	0.79	0.87	**0.88**	0.79	**0.88**
P@5	0.60	0.27	0.57	0.60	**0.63**	0.43	**0.63**
P@10	0.42	0.30	**0.48**	0.40	0.45	0.42	0.45

Very low!

Reason: large number of disambiguation errors

Category queries

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.22	0.27	0.48	0.34	0.34	**0.52**	0.41
NDCG@10	0.28	0.37	**0.53**	0.36	0.38	**0.53**	0.43
NDCG@all	0.66	0.68	**0.78**	0.70	0.71	0.77	0.73
P@5	0.17	0.27	**0.53**	0.23	0.30	**0.53**	0.33
P@10	0.20	0.30	**0.45**	0.20	0.28	0.37	0.28

Evaluation results

p_1: probability to move to a document node when being at a query-entity node
p_1 = 0.0 → move only to related-entity nodes
p_1 = 1.0 → move only to document nodes

- Stochastic modeling (using random jump probability = 0.2)

Single-entity queries

Measure	p_1=0.0	p_1=0.2	p_1=0.4	p_1=0.6	p_1=0.8	p_1=1.0
NDCG@5	0.09	0.27	0.48	0.63	0.65	**0.67**
NDCG@10	0.18	0.35	0.54	0.69	0.72	**0.74**
NDCG@all	0.60	0.68	0.77	0.85	**0.87**	**0.87**
P@5	0.07	0.23	0.43	0.47	0.53	**0.60**
P@10	0.12	0.23	0.33	0.37	0.42	**0.43**

Multiple-entity AND queries

Measure	p_1=0.0	p_1=0.2	p_1=0.4	p_1=0.6	p_1=0.8	p_1=1.0
NDCG@5	0.29	0.29	0.32	0.31	0.37	**0.48**
NDCG@10	0.36	0.36	0.40	0.45	0.47	**0.52**
NDCG@all	0.72	0.73	0.74	0.75	0.78	**0.80**
P@5	0.27	0.27	0.27	0.30	0.37	**0.57**
P@10	0.30	0.30	0.38	0.45	0.47	**0.50**

Multiple-entity OR queries

Measure	p_1=0.0	p_1=0.2	p_1=0.4	p_1=0.6	p_1=0.8	p_1=1.0
NDCG@5	0.20	0.30	0.32	0.38	0.45	**0.59**
NDCG@10	0.27	0.38	0.42	0.51	0.52	**0.65**
NDCG@all	0.66	0.71	0.72	0.75	0.76	**0.84**
P@5	0.27	0.33	0.33	0.37	0.47	**0.53**
P@10	0.30	0.33	0.35	0.42	0.42	**0.45**

Category queries

VERY LOW!

Measure	p_1=0.0	p_1=0.2	p_1=0.4	p_1=0.6	p_1=0.8	p_1=1.0
NDCG@5	0.43	0.47	**0.53**	0.50	0.46	0.23
NDCG@10	0.48	0.54	**0.55**	0.52	0.54	0.36
NDCG@all	0.74	0.77	**0.79**	0.77	0.77	0.68
P@5	0.57	0.57	**0.60**	**0.60**	**0.60**	0.23
P@10	0.42	**0.43**	0.40	0.40	**0.43**	0.28

Evaluation results

- Probabilistic vs Stochastic

Measure	Random ranking	Relativeness [A]	Timeliness [B]	Relatedness [C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.26	0.48	0.27	0.41	0.53	0.52	0.45	**0.56**
NDCG@10	0.33	0.52	0.36	0.50	0.55	0.56	0.51	**0.58**
NDCG@all	0.68	0.79	0.69	0.76	0.80	0.81	0.76	**0.82**
P@5	0.27	0.44	0.28	0.48	0.48	0.50	0.45	**0.52**
P@10	0.29	0.38	0.30	**0.45**	0.37	0.42	0.41	0.41

Best Stochastic
0.57
0.62
0.83
0.58
0.45

- For multi-entity OR queries:
  - Probabilistic model > stochastic model
  - Reason: not well-connected transition graph
    - E.g., when there are no documents mentioning all query entities

p_1 = 0.4 for category queries
p_1 = 0.0 for all other

Evaluation results - Synopsis

- **Category** queries:
  - Large number of query entities → more disambiguation errors
  - Consider **relatedness**: the association of the query entities with other entities
  - Stochastic model (with $p_1 = 0.4$) **>** probabilistic model

- Other query types:
  - Consider relativeness + relatedness
  - Timeliness does not seem to affect the rankings
  - Stochastic model outperforms probabilistic model for AND queries
  - Probabilistic model outperforms stochastic model for OR queries

Conclusion

- Problem: ranking archived documents for structured queries on semantic layers
- Approach: probabilistic and stochastic models that jointly consider:
  - the **relativeness** of the documents to the query entities
  - the **timeliness** of the documents
  - the temporal **relatedness** of other entities to the query entities
- New ground truth dataset (publicly available)
  - **http://l3s.de/~fafalios/jcdl/evaluation_dataset.zip**
- Evaluation results:
  - Useful insights on the effectiveness of the proposed models
  - Relatedness can limit the negative effect caused by disambiguation errors

- Future work:
  - Diversity-aware ranking methods
  - Apply the models on other types of archives (like Web or Social Media archives)

Thank you

Comments / Questions?

fafalios@L3S.de

L3S Research Center
Leibniz University of Hannover

ALEXANDRIA Project
(ERC Nr. 339233)