Learning the Query-Independent Ranking of RDF Entity Search Results

Lorand Dali, JSI
Blaž Fortuna, JSI
Thanh Tran, KIT
Dunja Mladenić, JSI
Introduction and Motivation

- Query answered using the information contained in the linked open data cloud
- We focus on queries whose answers are RDF resources (Entity Search)
- Often a high number of RDF resources match the constraints of the query
- The idea is to rank the answers such that the more important ones are presented first
What languages do they speak in Afghanistan?

- Pashto
- Uzbek
- Turkmen
- Dari
- Pashayi
- Askunu
- Moghol
- Kyrgyz
- Brahui
- Hindko
- Nuristani
- Baluchi
- Kamata-viri
- Vasi-vari
- Tregami
- Kalasha-ala
- Pamiri

1. All answers are correct
2. Ranking is important
3. Hard to obtain the true ranking
Contributions

- Learning to rank RDF resources
  - Query-independent features
  - Access logs based ground truth and training data
Learning to Rank

- Machine learning approach to building a ranking model

Ranking:
- We represent each answer (resource) as a feature vector
- The score is a linear combination of the features

Training:
- We know the true ranking (ground truth)
- The weights have to be learned

<table>
<thead>
<tr>
<th>A</th>
<th>f_1</th>
<th>f_2</th>
<th>f_3</th>
<th>f_4</th>
<th>f_5</th>
<th>f_6</th>
<th>f_7</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>f_1</td>
<td>f_2</td>
<td>f_3</td>
<td>f_4</td>
<td>f_5</td>
<td>f_6</td>
<td>f_7</td>
</tr>
<tr>
<td>C</td>
<td>f_1</td>
<td>f_2</td>
<td>f_3</td>
<td>f_4</td>
<td>f_5</td>
<td>f_6</td>
<td>f_7</td>
</tr>
</tbody>
</table>

Pairwise preferences
- A better than B
- A better than C
- B better than C
Learning to Rank

\( Q = \{q_1, q_2, \cdots, q_n\} \)  
set of queries

\( \forall q_i, R(q_i) = \{r_1^i, r_2^i, \cdots, r_k^i\} \)  
set of answers for \( q_i \)

\( F := (f_1, f_2, \cdots, f_P), \quad f_j : R(q_i) \rightarrow [0, 1] \)  
features set

\( x_j^i := (f_1(r_j^i), f_2(r_j^i), \cdots, f_P(r_j^i)) \)  
feature vector of \( r_j^i \)
Learning to Rank

\( (x^i_R, x^i_N), \quad f_t(r^i_R) > f_t(r^i_N) \quad \text{target feature } f_t \)

\[ C := \frac{2f_t(r^i_R)}{f_t(r^i_R) + f_t(r^i_N)} - 1 \quad \text{cost of the example } (x^i_R, x^i_N) \]

\[ w = (w_1, \ldots, w_p) \quad \text{ranking weights (Rank SVM)} \]

\[ score = w^T x \quad \text{ranking score of } x \]
FEATURES
Query Independent Features

- Extracted from the RDF graph
  - Centrality based (PageRank, HITS)
  - Frequencies of different patterns in the RDF graph
- Extracted from external sources
  - Web search
  - Ngram database
Centrality Features

- PageRank

\[ p = d \cdot M \cdot p + (1 - d) \cdot u, \quad p, u \in \mathbb{R}^n, M \in \mathcal{M}(n) \]

- HITS (Hubs and Authorities)

\[ \forall p, \quad auth(p) = \sum_{i=1}^{n} hub(i) \]

\[ \forall p, \quad hub(p) = \sum_{i=1}^{n} auth(i) \]
Features from RDF patterns

- Number of subjects @ K
- Number of objects @ K

\[ f_{@K} = |\{ tr \in RDF \mid \text{subj}(tr) = K \}| \]

\[ f_{@K} = |\{ tr \in RDF \mid \text{obj}(tr) = K \}| \]

Number of triples for which the grey node is a subject at K nodes away
Extracted from external sources

- Web search
  - Yahoo! BOSS

  "Neil Armstrong"

  About 7,050,000 results (0.22 seconds)

- N-gram database
  - Google n-grams
  - Long labels broken down into 3-grams
GROUND TRUTH / TRAINING DATA
Obtaining the Ground Truth (target feature)

- By asking humans – using crowdsourcing
- From access logs (automatically)

**Example**: What language do they speak in Afghanistan? – Pashto, Dari, Uzbek, …
Using Crowdsourcing

- We ask the worker to choose one answer which he would put on the first place in a ranked list of search results.
- The worker’s input is like a vote for the answer which he considers most important.
- Using several workers we define the ground truth as the number of votes an answer has got.
- We ask the worker to indicate his confidence in his choice from 1(I’m guessing) to 5(I am sure).
Using Access Logs

- Pashto
- Uzbek
- Dari
- Pashayi

The target feature is the number of times the Wikipedia page of the resource was visited.
Votes and visits correlate

NDCG = 0.917
The most voted resources are also the most visited.
Ground truths differ on difficult queries

- Where the two ground truths differ a lot the confidence and agreement of the workers is low
- Where the confidence of the workers is low their agreement is also low
EXPERIMENTS
Datasets and Queries

**General domain:**
- DBpedia, Yago
- Wikipedia access counts
  - [http://dumps.wikimedia.org/other/pagecounts-raw/](http://dumps.wikimedia.org/other/pagecounts-raw/)
- 25 queries from SemSearch Challenge

**Specific domain:**
- Semantic Web Dog Food (SWDF)
- Access logs from USEWOD 2011 Data Challenge
- 24 queries
<table>
<thead>
<tr>
<th>Specific domain</th>
<th>SWDF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NDCG</td>
</tr>
<tr>
<td>WikiLog</td>
<td>-</td>
</tr>
<tr>
<td>ResRank</td>
<td>0.6803</td>
</tr>
<tr>
<td>LM</td>
<td>0.7191</td>
</tr>
<tr>
<td>L_Centrality</td>
<td>0.7376</td>
</tr>
<tr>
<td>L_External</td>
<td>0.6149</td>
</tr>
<tr>
<td>L_RDF</td>
<td>0.7401</td>
</tr>
<tr>
<td>L_Complete</td>
<td>0.7533</td>
</tr>
</tbody>
</table>
Conclusions

- We have developed a learning to rank approach for ranking RDF entity search results using query independent features.
- We proposed a methodology for obtaining training data by (a) using crowdsourcing, and (b) automatically from access logs.
- The models trained on automatically obtained data appear to be at least as good as the ones obtained from humans.
- Certain features perform well in particular cases, but a combination of all features has a good performance overall.