Empirical Analysis of Ranking Models for an Adaptable Dataset Search

Neves, Angelo; Oliveira, Rodrigo; Leme, Luiz A P P (Fluminense University/Brasil)

Lopes, Giseli R (Federal University of RJ/Brasil)

Nunes, Bernardo P; Casanova, Marco A (PUC-Rio/Brasil)

Summary

1. Motivation
2. Ranking functions
3. Experiments
4. Adaptable search
5. Conclusions

Motivation

- The big picture of LOD
  - It is growing
  - It is poorly linked
  - It is mainly linked with popular and well-known datasets
    - Safe, but it narrows LOD potential
- Reasons for the lack of interlinking
  - Dataset quality issues
  - Lack of search tools
- Challenge
  - Expand entity interlinking with other sources

Motivation

Linking process of a dataset (*target*)

1. select relevant datasets with which the target would likely contain links
    i. *using ranked lists of datasets that would favors the likelihood of finding entity links*
2. inspect their contents to infer the links
    i. *using link discovery algorithms*
3. make links explicit by adding new RDF statements to the target

Motivation

Linking process for a target dataset

Use case 1

1. Rank datasets
2. Manually select ranked datasets
3. Discover links in the selected datasets
4. Create RDF links

Use case 2

1. Rank datasets
2. For each dataset d in the $top\ n$ datasets of the ranking
   - Discover candidate links
3. Decide about candidate links
4. Create RDF links

Motivation

Linking process for a target dataset

Use case 1

1. Ranking requests done by humans
2. Ranking quality usually assessed with nDCG metric
   - because users typically randomly choose entries in the top of the rank

Use case 2

1. Ranking requests done by computer programs
2. Ranking quality better assessed with recall@ rank position.
   - because programs would traverse an entire slice of the ranking

Motivation

Questions

1. Is the best ranking function under nDCG criterion the best ranking function under recall@ position as well?
2. How far in the ranking to go in the use case 2?
3. How it would be an adaptable dataset search if ranking functions were distinct in the first question?

Ranking functions

Function 1

- Based on Bayesian classifiers
- Intuition
  - if a target dataset has a set of metadata that is very frequent among the datasets that have entity links to a dataset d_i then it is likely the target will have links to entities of d_i as well
- Metadata
  - 1) Set of linksets, 2) Set of topic categories
- Ranking Function
  - $score(d_i, d_t) = Prob(d_i|M_{d_t}) = \log(P(d_i)) + \sum_{f_j \in M_{d_t}} \log\left(P(f_j|d_i)\right)$
    - M_{d_t} is a set of metadata of the target dataset

Ranking functions

Function 2

- Uses rule-based binary classifiers (C4.5 and RIPPERk)
- Intuition
  - given binary classifiers C_{d_i} that classify a target dataset as containing links or not to d_i, the likelihood of the target has links to d_i is proportional to the *class probability* ($if\ C_{d_i}(d_t) = d_i$) or *1 – class probability* ($if\ C_{d_i}(d_t) = \neg d_i$).
- Metadata
  - 1) Set of linksets, 2) Set of topic categories
- Ranking Function
  - $score(d_i, d_t) = \begin{cases} class\ probability & , C_{d_i}(d_t)=d_i \\ 1 - class\ probability & , C_{d_i}(d_t)=\neg d_i \end{cases}$
    - M_{d_t} is a set of metadata of the target dataset

Experiments

- Data
  - Datahub → datasets and their linksets
  - Dumps, SPARQL endpoints → topic categories
    - literals $\xrightarrow{1)\ DBpedia\ Spotlight}$ entities $\xrightarrow{2)\ Query_{DBpedia}}$ topics $\xrightarrow{3)\ Query_{DBpedia}}$ categories

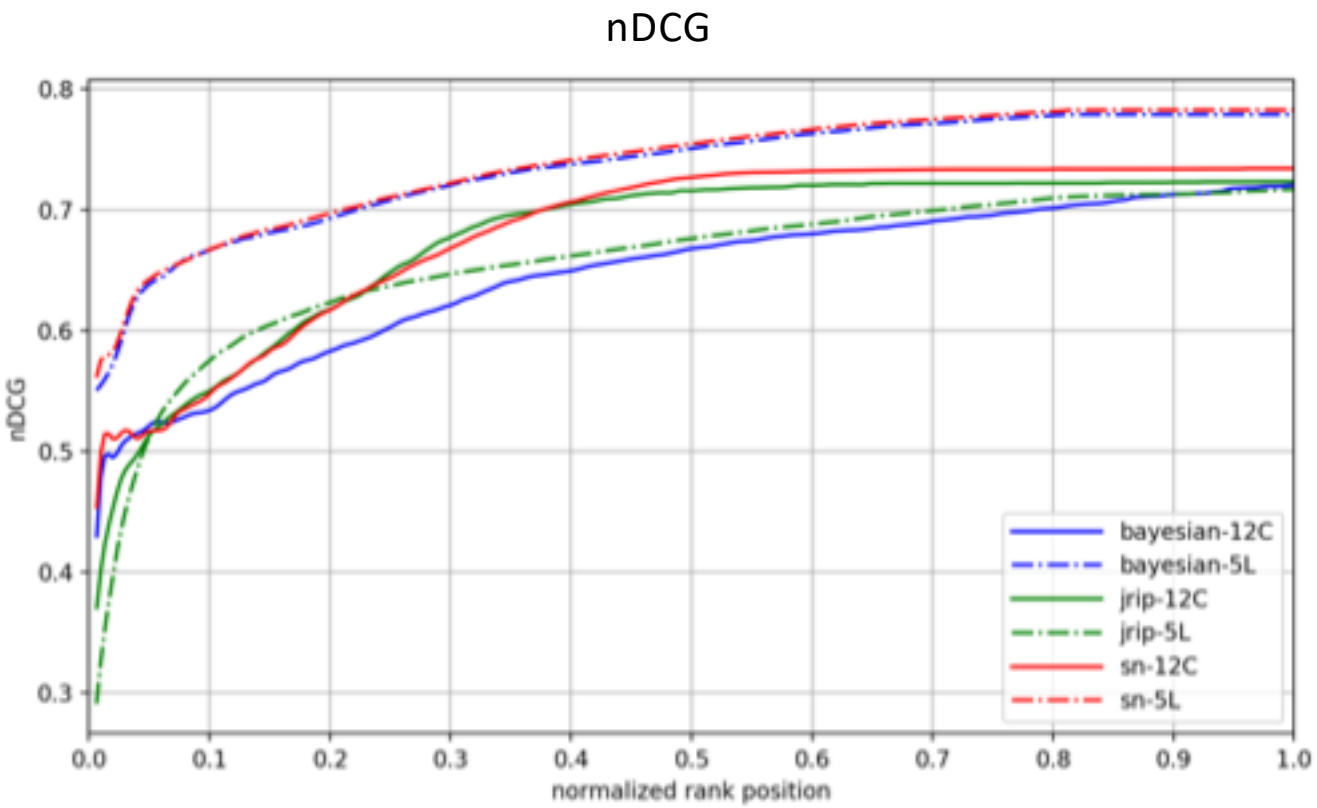
Experiments

Process

10-fold cross-validation

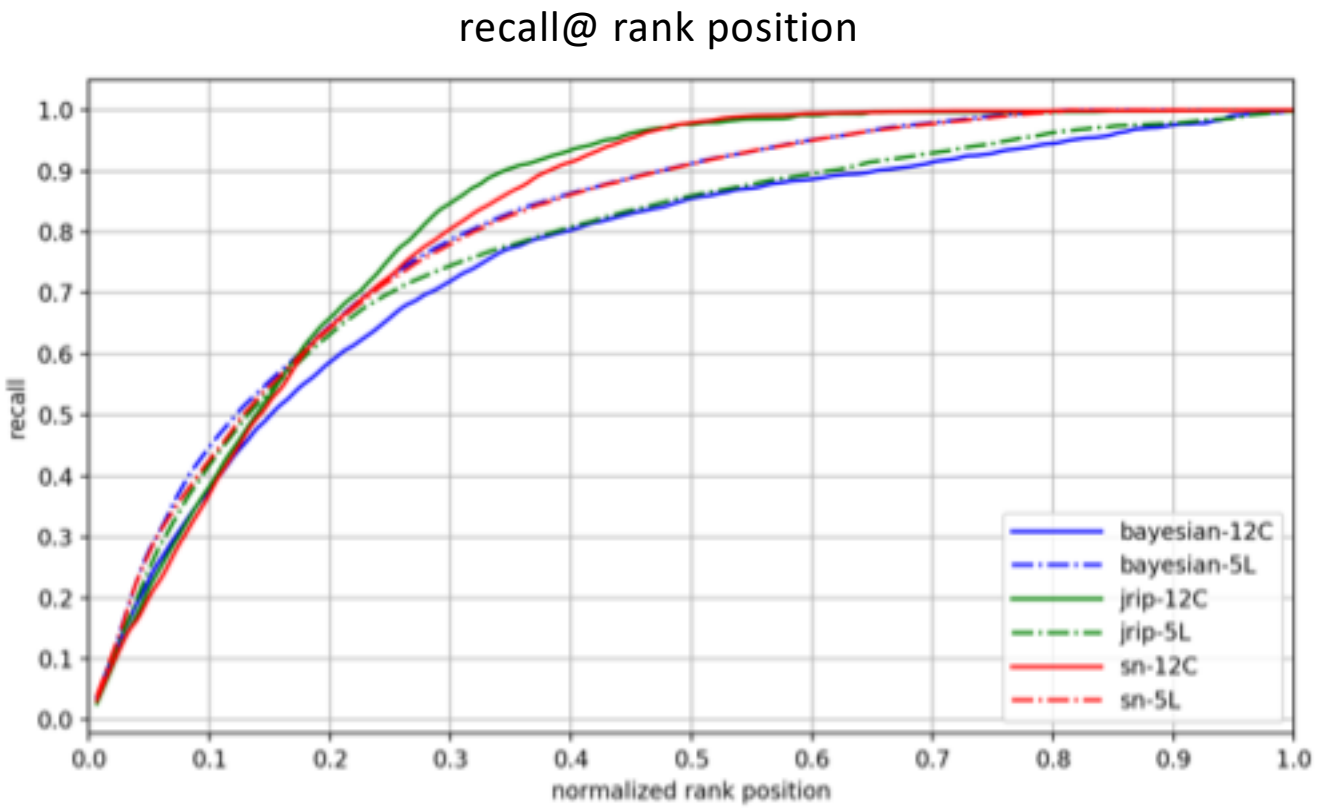
- for each partition P_j of the set of datasets D
  - for each dataset $d_t \in P_j$
    - rank datasets in remaining partitions to d_t
    - compute nDCG and $recall@(position)$ of the ranking
- compute the average nDCG and $recall@(position)$

Experiments



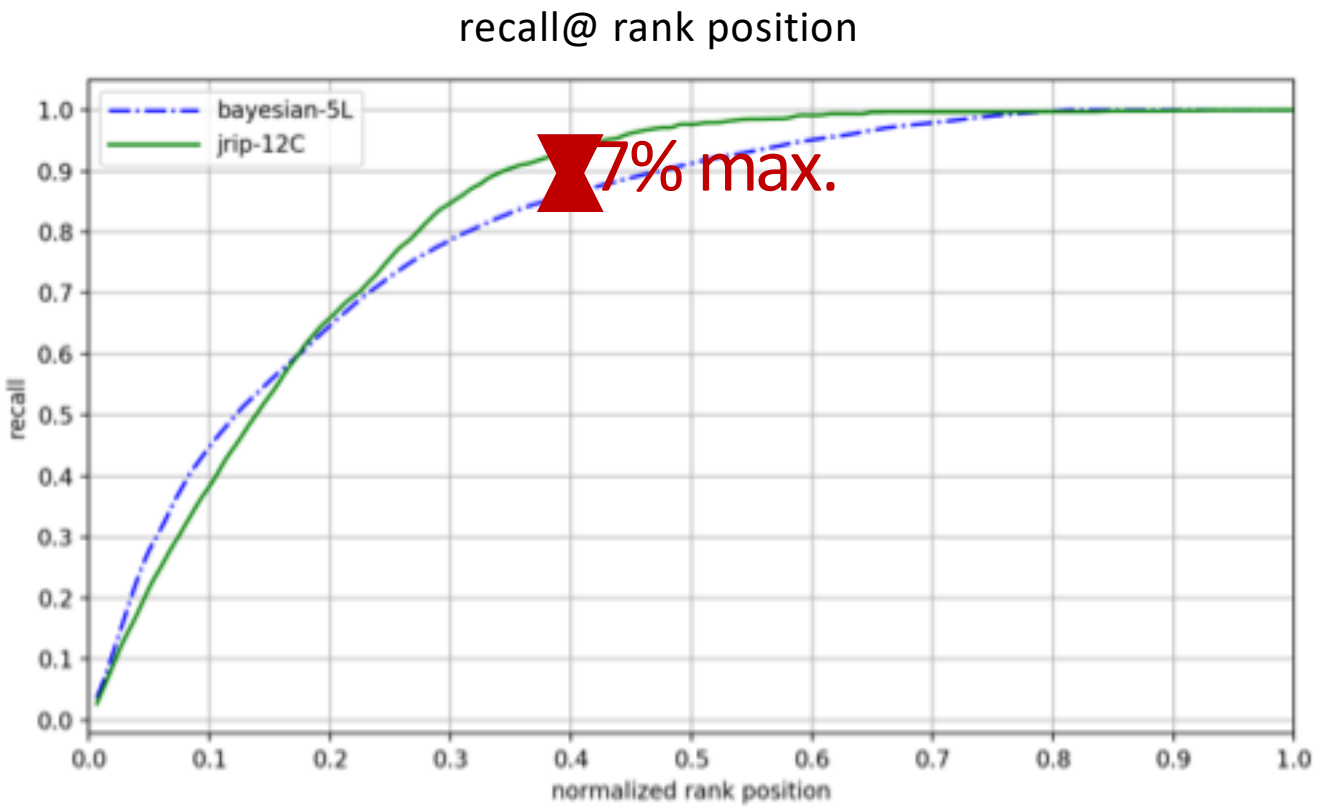
- Best function if linksets ARE available
  - Bayesian using linksets
  - Do not need similarity calc.
- Best function if linksets ARE NOT available
  - JRip (RIPPERk/ Weka) using topic categories
  - Do not need similarity cal.

Experiments



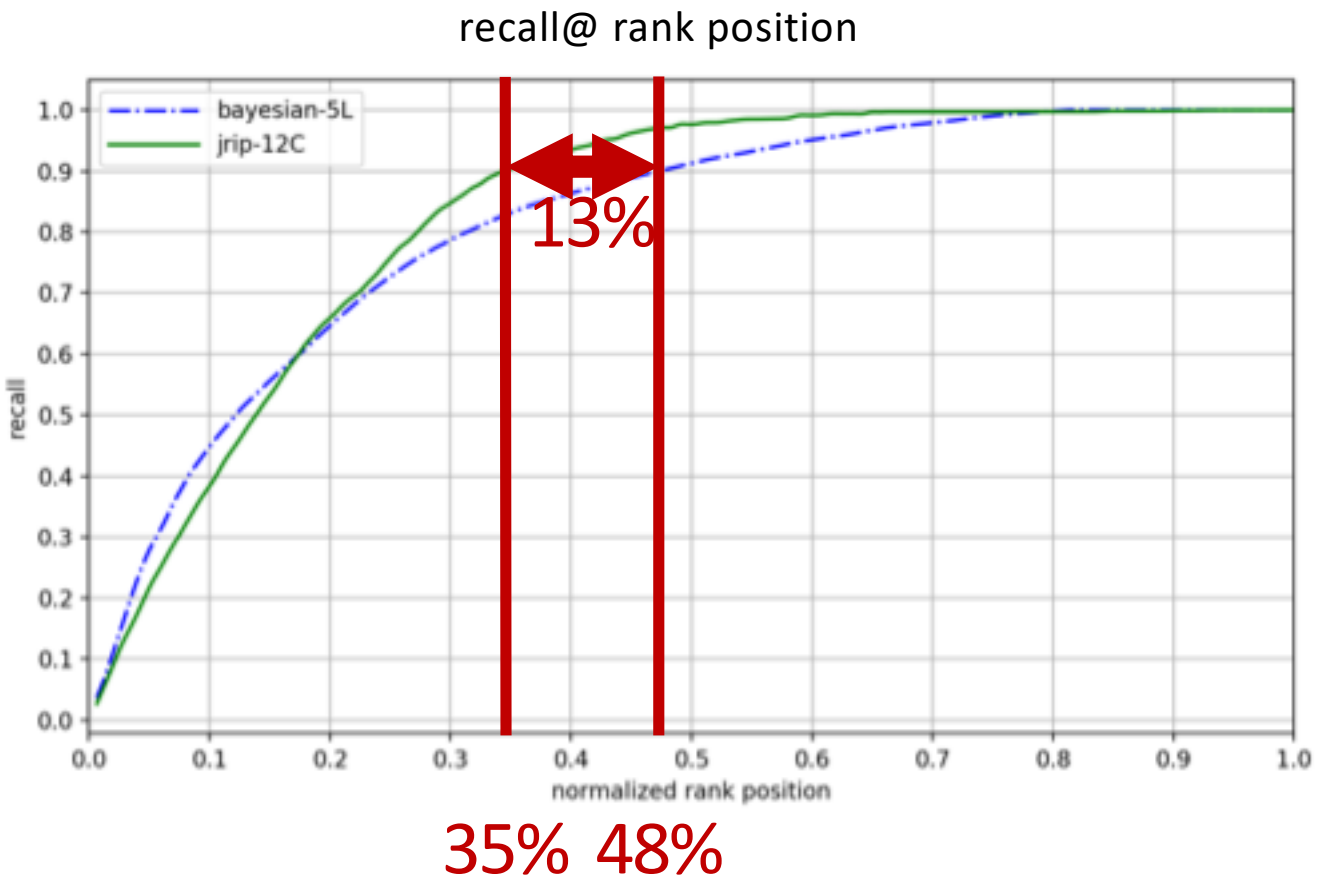
- Best function
  - JRip using topic categories

Experiments



- Best function
  - JRip using topic categories
  - Can be up to 7% better than the best model under nDCG

Experiments



- Best function
  - JRip using topic categories
  - It requires 13% less datasets to reach 90% of recall

Adaptable Dataset Search

- An *adaptable dataset search engine* would distinguish the two use cases by *content negotiation* and would apply the most suited ranking function.
  - HTTP requests contain human readable RDF MIME types $\Longrightarrow$ use case 1
  - HTTP requests contain machine readable RDF MIME types $\Longrightarrow$ use case 2

Adaptable Dataset Search

- Adaptable search

```
Use case 1: human interaction
    if linksets are available for the target dataset (check VoID)
        use Bayesian ranking with sets of linksets as metadata
    else if topic categories available (check VoID)
        use JRIP with topic categories as dataset metadata
    else return {}
Use case 2: "machine" interaction
    use the rule classifier JRIP with topic categories as metadata
```

Conclusions

- Challenge: expanding entity linking in the WoD
- Manual and automatic use cases are possible for a search tool.
- Experiments indicate that adaptable search is possible and useful

Thank You!