Two Approaches to the Dataset Interlinking Recommendation Problem

Giseli Rabello Lopes[1], Luiz André P. Paes Leme[2], Bernardo Pereira Nunes[1], Marco Antonio Casanova[1], and Stefan Dietze[3]

[1] Department of Informatics, Pontifical Catholic University of Rio de Janeiro, Rio de Janeiro, RJ – Brazil, CEP 22451-900
{grlopes,bnunes,casanova}@inf.puc-rio.br
[2] Computer Science Institute, Fluminense Federal University, Niterói, RJ – Brazil, CEP 24210-240
lapaesleme@ic.uff.br
[3] L3S Research Center, Leibniz University Hannover, Appelstr. 9a, 30167 Hannover, Germany
{dietze}@l3s.de

Abstract. Whenever a dataset t is published on the Web of Data, an exploratory search over existing datasets must be performed to identify those datasets that are potential candidates to be interlinked with t. This paper introduces and compares two approaches to address the dataset interlinking recommendation problem, respectively based on Bayesian classifiers and on Social Network Analysis techniques. Both approaches define rank score functions that explore the vocabularies, classes and properties that the datasets use, in addition to the known dataset links. After extensive experiments using real-world datasets, the results show that the rank score functions achieve a mean average precision of around 60%. Intuitively, this means that the exploratory search for datasets to be interlinked with t might be limited to just the top-ranked datasets, reducing the cost of the dataset interlinking process.

Keywords: Linked Data, data interlinking, recommender systems, Bayesian classifier, social networks.

1 Introduction

Over the past years there has been a considerable movement towards publishing data on the Web following the Linked Data principles [1]. According to those principles, to be considered 5-star, a dataset must comply with the following requirements: (i) be available on the Web; (ii) be available as machine-readable structured data; (iii) be in a non-proprietary format; (iv) use open standards from W3C (i.e. RDF and SPARQL) to identify resources on the Web; and (v) be linked to other people's data to provide additional data. This paper addresses the last requirement.

Briefly, in the context of Linked Data, a *dataset* is a set of RDF triples. A resource identified by an RDF URI reference s *is defined in* a dataset t iff s occurs as the subject of a triple in t.

A *feature* of a dataset is a vocabulary URI, a class URI or a property URI used in triples of the dataset. One may then represent the dataset by one or more of its features.

Let t and u be two datasets. A *link* from t to u is a triple of the form (s, p, o) such that s is defined in t and o is defined in u. We say that t is *linked to* u, or that u is *linked from* t, iff there is at least one link from t to u. We also say that u is *relevant* for t iff there is at least one resource defined in u that can be linked from a resource defined in t.

The *dataset interlinking recommendation problem* can then be posed as follows:

> *Given a finite set of datasets* $\boldsymbol{D}$ *and a dataset* t*, compute a rank score for each dataset* $u \in \boldsymbol{D}$ *such that the rank score of* u *increases with the chances of* u *being relevant for* t*.*

To address the dataset interlinking recommendation problem, this paper proposes and compares two approaches respectively based on Bayesian classifiers and on Social Network link prediction measures. Both approaches define rank score functions that explore the dataset features and the known links between the datasets. The experiments used real-world datasets and the results show that the rank score functions achieve a mean average precision of around 60%. Intuitively, this means that a dataset interlinking tool might limit the search for links from a dataset t to just the top ranked datasets with respect to t and yet find most of the links from t.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 introduces our proposed approaches based on Bayesian classifiers and on Social Network Analysis techniques. Section 4 presents the experiments conducted to test and compare the approaches. Finally, Section 5 contains the conclusions and directions for future work.

2 Related Work

In this paper, we extend previous work [2,3] that introduced preliminary versions of the rank score functions respectively based on the Bayesian and the Social Network approaches. This paper contains significantly new results over our previous work in so far as it (i) explores different sets of features to compute rank score functions; (ii) uses modified rank score functions to interlink new datasets without known links; and (iii) provides a comprehensive comparison of the approaches using different feature sets.

In more detail, the paper improves previous results as follows. As for the Bayesian ranking definition, the paper formally defines how to manage the lack of observations of co-occurrences between features and links. Without this new definition, null probabilities could lead the score function to a discontinuity region ($log(0)$). In the SN-based ranking definition, we propose a new score function (not defined in [3]) which combines preferential attachment and resource allocation measures. The definition of the similarity set of the target dataset is also novel.

Furthermore, we do not assume that one knows the existing links of the dataset to which one wants to generate recommendations for. This assumption is realistic for new datasets and tackles one of the core challenges of the Linked Data principles. Indeed, for new datasets, the approach proposed in [3] will not work and that presented in [2] will generate recommendations based only on the popularity of the datasets (i.e., the recommendations will be the same for all datasets).

We explored different sets of features - properties, classes and vocabularies - to compute the rank score functions. Moreover, we thoroughly compared the performance of the improved approaches using different feature sets.

Nikolov et al. [4,5] propose an approach to identify relevant datasets for interlinking, with two main steps: (i) searching for potentially relevant entities in other datasets using as keywords a subset of labels in the new published dataset; and (ii) filtering out irrelevant datasets by measuring concept similarities obtained by applying ontology matching techniques.

Kuznetsov [6] describes a linking system which is responsible for discovering relevant datasets for a given dataset and for creating instance level linkage. Relevant datasets are discovered by using the *referer* attribute available in the HTTP message header, as described in [7], and ontology matching techniques are used to reduce the number of pairwise comparisons for instance matching. However, this work does not present any practical experiments to test the techniques.

When compared with these approaches, the rank score functions proposed in this paper use only metadata and are, therefore, much simpler to compute and yet achieve a good performance.

The next set of papers aim at recommending datasets with respect to user queries, which is a problem close, but not identical to the problem discussed in this paper. Lóscio et al. [8] address the recommendation of datasets that contribute to answering queries posed to an application. Their recommendation function estimates a degree of relevance of a given dataset based on an information quality criteria of correctness, schema completeness and data completeness. Wagner et al. [9] also propose a technique to find relevant datasets for user queries. The technique is based on a contextualization score between datasets, which is in turn based on the overlapping of sets of instances of datasets. It uses just the relationships between entities and disregards the schemas of the datasets. Oliveira et al. [10] use application queries and user feedback to discover relevant datasets. Application queries help filter datasets that are potentially strong candidates to be relevant and user feedback helps analyze the relevance of such candidates.

Toupikov et al. [11] adapt the original PageRank algorithm to rank existing datasets with respect to a given dataset. The technique uses the Linksets descriptions available in VoID files as the representation of relationships between datasets and the number of triples in each Linkset as the weight of the relationships. Results show that the proposed technique performs better than traditional ranking algorithms, such as PageRank, HITS and DRank. As the rank score functions defined in this paper, the version of the PageRank algorithm the authors propose depends on harvesting VoID files.

3 Ranking Techniques

Sections 3.1 and 3.2 introduce two approaches to compute rank score functions, leaving a concrete example to Section 3.3.

3.1 Bayesian Ranking

This section defines a rank score function inspired on conditional probabilities. However, we note that the rank score is not a probability function. We proceed in a stepwise fashion until reaching the final definition of the rank score function, in Equation 9.

Let $\boldsymbol{D}$ be a finite set of datasets, d_i be a dataset in $\boldsymbol{D}$ and t be a dataset one wishes to link to datasets in $\boldsymbol{D}$. Let T denote the event of selecting the dataset t, D_i denote the event of selecting a dataset in $\boldsymbol{D}$ that has a link to d_i, and F_j denote the event of selecting a dataset that has feature f_j (recall that a *feature* of a dataset is a vocabulary URI, a class URI or a property URI used in triples of the dataset).

We tentatively define the rank score function as a conditional probability:

$$score_0(d_i, t) = P(D_i|T) \tag{1}$$

that is, $score_0(d_i, t)$ is the conditional probability that D_i occurs, given that T occurred. As required, this score function intrinsically favors those datasets with the highest chance of defining links from t.

We then rewrite $score_0$, using Bayes's rule, as follows:

$$score_1(d_i, t) = \frac{P(T|D_i)}{P(T)} P(D_i) \tag{2}$$

As in Bayesian classifiers [12,13], by representing t as a bag of features $\boldsymbol{F} = \{f_1, ..., f_n\}$, one may rewrite $score_1$ as:

$$score_2(d_i, t) = \frac{P(\{f_1, ..., f_n\}|D_i)}{P(\{f_1, ..., f_n\})} P(D_i) \tag{3}$$

By the naive Bayes assumption [12,13], $P(\{f_1, f_2, ..., f_n\}|D_i)$ can be computed by multiplying conditional probabilities for each independent event F_j (the event of selecting datasets with just the feature f_j). Moreover, $P(\{f_1, ..., f_n\})$ does not change the rank order because it is the same for all d_i. Hence, we remove this term. The new score function becomes:

$$score_3(d_i, t) = \left(\prod_{j=1..n} P(F_j|D_i) \right) P(D_i) \tag{4}$$

The final score function is obtained from $score_3$ by replacing the product of the probabilities by a summation of logarithms, with the help of auxiliary functions p and q that avoid computing $log(0)$.

Intuitively, the definitions of functions p and q penalize a dataset d_i when no dataset with feature f_j is linked to d_i or when no dataset is linked to d_i. The definitions depend on choosing a constant C that satisfies the following restriction (where m is the number of datasets in $\boldsymbol{D}$ and n is the number of features considered):

$$\begin{aligned} &C < min(C', C'') \\ &C' = min\{P(F_j|D_i) \in [0,1] \ / \ P(F_j|D_i) \neq 0 \wedge j \in [1,n] \wedge i \in [1,m]\} \\ &C'' = min\{P(D_i) \in [0,1] \ / \ P(D_i) \neq 0 \wedge i \in [1,m]\} \end{aligned} \tag{5}$$

Then, p is defined as follows:

$$p(F_j, D_i) = \begin{cases} C, & if P(F_j|D_i) = 0 \\ P(F_j|D_i), & \text{otherwise} \end{cases} \tag{6}$$

Intuitively, p avoids computing $log(P(F_j|D_i))$ when $P(F_j|D_i) = 0$, that is, when no dataset with feature f_j is linked to d_i. In this case, d_i is penalized and $p(F_j, D_i)$ is set to C.

Likewise, q is defined as follows:

$$q(D_i) = \begin{cases} C, & if P(D_i) = 0 \\ P(D_i), & \text{otherwise} \end{cases} \tag{7}$$

Intuitively, q avoids computing $log(P(D_i))$ when $P(D_i) = 0$, that is, when no dataset is linked to d_i. In this case, d_i is also penalized and $q(D_i)$ is set to C.

We define the final rank score function in two steps. We first define:

$$score(d_i, t) = \left(\sum_{j=1..n} log(p(F_j, D_i)) \right) + log(q(D_i)) \tag{8}$$

and then eliminate $p(F_j, D_i)$ from Equation 8 :

$$score(d_i, t) = c \ |N_i| + \left(\sum_{f_j \in P_i} log(P(F_j|D_i)) \right) + log(q(D_i)) \tag{9}$$

where

- $c = log(C)$
- $N_i = \{f_j \in \boldsymbol{F} / P(F_j|D_i) = 0\}$
- $P_i = \boldsymbol{F} - N_i$

In particular, we note that, when t does not have any feature (i.e., when $n = 0$), the score function takes into account only the unconditional probability $P(D_i)$. In this case, the most popular datasets, such as DBpedia[1] and Geonames[2], will be favored by the score function at the expenses of perhaps more

[1] `http://dbpedia.org/`
[2] `http://www.geonames.org/`

appropriate datasets. The ranking may not be accurate in such borderline cases, but a popularity-based ranking is preferable to no ranking at all, when nothing is known about t.

Equation 9, therefore, defines the final score function that induces the ranking of the datasets in $\boldsymbol{D}$ (from the largest to the smallest score). Section 3.3 illustrates how the score is computed.

Based on the maximum likelihood estimate of the probabilities [13] in a training set of datasets, the above probabilities can be estimated as follows:

$$P(F_j|D_i) = \frac{count(f_j, d_i)}{\sum_{j=1}^{n} count(f_j, d_i)} \tag{10}$$

$$P(D_i) = \frac{count(d_i)}{\sum_{i=1}^{m} count(d_i)} \tag{11}$$

where $count(f_j, d_i)$ is the number of datasets in the training set that have feature f_j and that are linked to d_i, $count(d_i)$ is the number of datasets in the training set that are linked to d_i, disregarding the feature set. Thus, for any dataset t represented by a set of features, the rank position of each of the datasets in $\boldsymbol{D}$ can be computed using Equations 7, 9, 10 and 11.

Note that Equation 10 depends on the correlation between f_j and d_i in the training set. This means that the higher the number of datasets correlating feature f_j with links to d_i, the higher the probability in Equation 10. Moreover, as Equation 4 depends on the joint probability of the features f_j of t, the higher the number of features shared by t and the datasets linked to d_i with high probability, the higher $score(d_i, t)$ will be. That is, if the set of features of t is very often correlated with datasets that are linked to d_i and t is not already linked to d_i, then it is recommended to try to link t to d_i.

Finally, we stress that, if a dataset t exhibits a set of features $\boldsymbol{F}$, one can choose any subset of $\boldsymbol{F}$ as the representation of t. Thus, each possible representation may generate different rankings with different performances and one cannot predict in advance which representation will generate the best ranking. Section 4 then compares the results obtained for several different feature sets.

3.2 Social Network-Based Ranking

In Social Networks Analysis (SNA), the network is typically represented as a graph, where the nodes are the entities (e.g., users, companies) and the edges are the relationships between them (e.g., follows, shares, befriends, co-authorships). In SNA, the *link prediction problem* refers to the problem of estimating the likelihood of the existence of an edge between two nodes, based on the already existing edges and on the attributes of the nodes [14]. We propose to analyze the dataset interlinking recommendation problem in much the same way as the link prediction problem.

As in Section 3.1, let $\boldsymbol{D}$ be a finite set of datasets, d_i be a dataset in $\boldsymbol{D}$ and t be a dataset one wishes to link to datasets in $\boldsymbol{D}$. Recall again that a *feature* of

a dataset is a vocabulary URI, a class URI or a property URI used in triples of the dataset.

The *Linked Data network* for $\boldsymbol{D}$ is a directed graph such that the nodes are the datasets in $\boldsymbol{D}$ and there is an edge between datasets u and v in $\boldsymbol{D}$ iff there is a link from u to v.

The *similarity set* of a dataset t, denoted S_t, is the set of all datasets in $\boldsymbol{D}$ that have features in common with t. The *popularity set* of a dataset $d_i \in \boldsymbol{D}$, denoted P_{d_i}, is the set of all datasets in $\boldsymbol{D}$ that have links to d_i.

Among the traditional measures adopted for link prediction [15,14], we will use Preferential Attachment and Resource Allocation. Indeed, the results reported in [16], which analyzed the dataset interlinking recommendation problem using just the existing links, indicate that these two measures achieved the best performance.

Preferential Attachment. The Preferential Attachment score estimates the possibility of defining a link from t to d_i as the product of the cardinality of the similarity set of t, denoted $|S_t|$, and the cardinality of the popularity set of d_i, denoted $|P_{d_i}|$, and is defined as follows:

$$pa_0(t, d_i) = |S_t| \times |P_{d_i}| \tag{12}$$

However, since $|S_t|$ is independent of d_i, this term does not influence the rank score of the datasets. Thus, we may ignore it and define pa as follows:

$$pa(t, d_i) = |P_{d_i}| \tag{13}$$

Resource Allocation. Let d_j be a dataset in $\boldsymbol{D}$, distinct from d_i. Intuitively, if there are links from t to d_j and from d_j to d_i and there are many other datasets that have links to d_j, then d_j must be a generic dataset (eg. DBpedia, Geonames, etc.). Therefore, d_j does not necessarily suggest any possible link from t to d_i. On the other hand, if there are not many datasets that have links to d_j, then this might be a strong indication that d_j is a very particular dataset for both t and d_i and, therefore, a link from t to d_i might as well be defined. Thus, the strength of the belief in the existence of a link from t to d_i increases inversely proportional to the number of datasets which have links to d_j, i.e., depends on the cardinality of the popularity set of d_j, again denoted $\left|P_{d_j}\right|$.

The Resource Allocation score estimates the possibility of defining a link from t to d_i as a summation of the inverse of the cardinality of the popularity set of the datasets in the intersection of the datasets linked from t, which is the similarity set S_t of t, and the datasets linked to d_i, which is the popularity set $P_{d_i}$of d_i. It is defined as follows:

$$ra(t, d_i) = \sum_{d_j \in S_t \cap P_{d_i}} \frac{1}{\left|P_{d_j}\right|} \tag{14}$$

Combined Score. To obtain more accurate results, we combine the two previous scores into a new score, defined as follows:

$$score(t, d_i) = ra(t, d_i) + \frac{pa(t, d_i)}{|\boldsymbol{D}|} \tag{15}$$

This final score gives priority to the *ra* score; the *pa* score, normalized by the total number of datasets to be ranked ($|\boldsymbol{D}|$), will play a role when there is a tie or when the *ra* value is zero. Section 4.3 comments on the adequacy of defining a combined score function.

3.3 Example of Rank Score Computations

We illustrate how to compute rank score functions, using both approaches, with the help of a schematic example. We selected a subset of the datasets indexed by the DataHub[3], using the *Learning Analytics and Knowledge*[4] dataset [17], referred to as *lak* in what follows, as the target of the recommendation.

As features of *lak*, we used three classes, *swc:ConferenceEvent*, *swrc:Proceedings* and *swrc:InProceedings*, obtained from the LinkedUp project Web site[5].

As the candidates to be ranked, we selected the datasets *webscience*, *webconf*, *wordnet*, *dblp* and *courseware*. They were chosen because we considered all datasets that share at least one feature with *lak* (*webscience* and *webconf*) and all datasets linked from them (*wordnet* and *dblp*). In addition, to better illustrate the computation of the rank scores, we also considered *courseware*, one of the datasets linked to *wordnet*.

The similarity set of *lak* consists of the datasets *webscience* and *webconf*, since they share at least one feature with *lak*. The datasets *webscience* and *webconf* shares respectively the *swc:ConferenceEvent* class and the *swc:ConferenceEvent*, *swrc:Proceedings* and *swrc:InProceedings* classes with *lak*.

Table 1 and Table 2 respectively list the URIs of all such datasets and classes. Figure 1 depicts these objects, where the directed thin arrows represent the existing links among the datasets, the thick arrows denote links from *lak* to datasets in its similarity set (used only by Social Network-based approach) and the dashed lines indicate which datasets have what features. The dashed cylinders refer to groups of datasets (the number of datasets grouped is indicated inside the cylinder).

The rank score functions have to rank the datasets *webscience*, *webconf*, *wordnet*, *dblp* and *courseware* according to the chances of defining links from resources in *lak* to resources in each of these datasets. The datasets in the similarity set of *lak* (*webscience* and *webconf*) are included in the list of candidates to be ranked because they are not yet linked from *lak*.

[3] `http://datahub.io/`
[4] `http://lak.linkededucation.org`
[5] `http://linkedup-project.eu/`

Table 1. The dataset acronym and the corresponding URI

Dataset	URI
lak	http://lak.linkededucation.org
webscience	http://webscience.rkbexplorer.com
webconf	http://webconf.rkbexplorer.com
dblp	http://knoesis.wright.edu/library/ontologies/swetodblp/
wordnet	http://www.w3.org/TR/wordnet-rdf
courseware	http://courseware.rkbexplorer.com

Table 2. The class feature acronym and the corresponding URI

Class	URI
swc:ConferenceEvent	http://data.semanticweb.org/ns/swc/ontology#ConferenceEvent
swrc:Proceedings	http://swrc.ontoware.org/ontology#Proceedings
swrc:InProceedings	http://swrc.ontoware.org/ontology#InProceedings

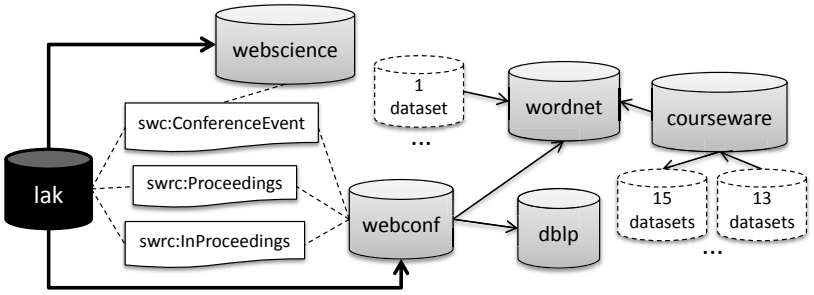
The Social Network-based rank score function (shown in Equation 15) ranks *wordnet* in the first position (the largest score value), *dblp* in the second position, *courseware* in the third position and *webscience* and *webconf* (with tied scores) in the last two positions. Recall that the Social Network-based score function is the sum of two terms, *ra* and *pa*. The first two best ranked datasets have scores determined by *ra* greater than zero because they are linked from *webconf*, which is in the similarity set of *lak*. The remaining datasets are ranked only by the *pa* term, including *webconf* and *webscience*, because they are in the similarity set of *lak*.

Using the Bayesian approach, the rank score function ranks *dblp* in the first position, *wordnet* in the second position, *courseware* in the third position and *webscience* and *webconf* (with tied scores) in the last two positions. It is not possible to adequately estimate probability values for *webscience* and *webconf* because they are both not linked from any other dataset. Thus, in this example, their score values will be the minimum, determined in this case by $c * 4 = -60$ (omitted from the table in Figure 1 for convenience). Intuitively, the top ranking positions assigned to *wordnet* and *dblp* are justified because both datasets are linked from datasets that share some feature with *lak* and the popularity of both can be estimated.

A manual inspection performed in the two best ranked datasets by both approaches indicated that the recommendation of *dblp* is justified because the DBLP digital library[6] indexes the papers published in the LAK and EDM conferences, as does the *lak* dataset. Then, resources of *lak* can be linked to resources in *dblp* (e.g., using *owl:sameAs* property). The recommendation of *wordnet* is also justified because resources of *lak* can be linked to the corresponding concepts defined in *wordnet*.

Both approaches presented in the paper (SN-based and Bayesian) are related to the correlation between features and links. Therefore, our approaches could

[6] http://www.informatik.uni-trier.de/~ley/db/



Social Network-based

$$\text{score}(\text{lak}, webscience) = 0 + \frac{1}{295} = 0.0034$$

$$\text{score}(\text{lak}, webconf) = 0 + \frac{1}{295} = 0.0034$$

$$\text{score}(\text{lak}, wordnet) = \frac{1}{1} + \frac{3}{295} = 1.0102$$

$$\text{score}(\text{lak}, dblp) = \frac{1}{1} + \frac{1}{295} = 1.0034$$

$$\text{score}(\text{lak}, courseware) = 0 + \frac{13}{295} = 0.0441$$

Bayesian Scores (using c=-15)

Partial Result	**d_1**	**d_2**	**d_3**
$count(f_1,d_i)$	1	1	0
$count(f_2,d_i)$	1	1	0
$count(f_3,d_i)$	1	1	0
$\Sigma_{i=1,...n}count(f_i,d_i)$	3	3	0
$count(d_i)$	3	1	13
$P(F_2\|D_i)$	0.33	0.33	-
$P(F_2\|D_i)$	0.33	0.33	-
$P(F_3\|D_i)$	0.33	0.33	-
$P(D_i)$	0.004	0.001	0.019
$\|N_i\|$	0	0	3
$\log_2(P(F_1\|D_i))$	-2	-2	-
$\log_2(P(F_2\|D_i))$	-2	-2	-
$\log_2(P(F_3\|D_i))$	-2	-2	-
$\log_2(P(D_i))$	-7.86	-9.45	-5.74
score(d_i,t)	**-12.61**	**-14.20**	**-50.74**

f_1=swc:ConferenceEvent, f_2=swrc:Proceedings, f_3=swrc:InProceedings
d_1=wordnet, d_2=dblp, d_3=courseware
sum($count(d_i)$)=697

Fig. 1. Example including the datasets links, associated features and the score computation

recommend two datasets that do not share any feature (vocabulary, class and property) as candidates to be interlinked. Considering the example, *lak* and *dblp* have completely different feature sets and yet could be interlinked. As there is *webconf* (that has common features with *lak*) linked to *dblp*, then our approaches can recommend to try to interlink *lak* to *dblp*.

4 Experiments

4.1 Notation and Performance Measures

To motivate how we define the performance measure, recall that the goal of the rank score functions is to reduce the effort required to discover new links

from a dataset t. With the appropriate ranking, datasets more likely to contain links from t will be better positioned in the ranking so that the search may be concentrated on the datasets at the top of the ranking. Thus, in the experiments, we evaluated the rank score functions using the Mean Average Precision, which is a traditional Information Retrieval measure [18,19]. Furthermore, we remark that, since the rank score functions induce a ranking of all datasets, the recall is always 100% and is, therefore, not used as a performance measure.

To define the Mean Average Precision (MAP), we adopt the following notation (recall that a dataset u is *relevant* for a dataset t iff there is at least one resource defined in u that can be linked from a resource defined in t):

- $\boldsymbol{D}$ is a set of datasets
- $\boldsymbol{T}$ is a set of datasets, disjoint from $\boldsymbol{D}$, one wishes to link to datasets in $\boldsymbol{D}$
- $t \in \boldsymbol{T}$
- G_t is the set of datasets in $\boldsymbol{D}$ with known links from t (the *gold standard* for t)
- $Prec@k_t$ is the number of relevant datasets obtained until position k in a ranking for t, divided by k (the *precision at position k* of a ranking for t)
- $AveP_t = (\sum_k Prec@k_t)/\left|G_t\right|$, for each position k in a ranking for t in which a relevant dataset occurs (the *average precision at position k* of a ranking for t)

The *Mean Average Precision* (MAP) of a rank score function over the datasets in $\boldsymbol{T}$ is then defined as follows:

$$MAP = average\{AveP_{t_j} \ / \ t_j \in \boldsymbol{T} \ \wedge \ \left|G_{t_j}\right| > 0\} \tag{16}$$

Moreover, in order to evaluate whether the improvements are statistically significant, a paired statistical *Student's T-test* [18,19] was performed. According to Hull [20], the T-test performs well even for distributions which are not perfectly normal. We adopted the usual threshold of $\alpha = 0.05$ for statistical significance. When a paired T-test obtained a p-value (probability of no significant difference between the compared approaches) less than α, there is a significant difference between the compared approaches.

4.2 Dataset

We tested the rank score functions with metadata extracted from the DataHub catalog, a repository of metadata about datasets, in the style of Wikipedia. DataHub is openly editable and can be accessed through an API provided by the data cataloguing software CKAN[7]. The set of data used in our experiments is available at `http://www.inf.puc-rio.br/~casanova/Publications/Papers/2014-Papers/interlinking-test-data.zip` and was extracted in April 2013.

[7] `http://ckan.org`

We adopted as features the properties, classes and vocabularies used in the datasets, in different combinations. From the DataHub catalog, we managed to obtain 295 datasets with at least one feature and 697 links between these datasets. The number of distinct features was 12,102, where 10,303 were references to properties, 6,447 references to classes and 645 references to vocabularies; the number of relations between datasets and features was 17,395.

We conclude with brief comments on how we extracted metadata from the DataHub catalog.

Let t be a dataset and V be a set of VoID descriptions [21] for t, available through the catalog. We extracted classes and properties used in t from dataset partitions defined in V, using the *void:class* and the *void:property* properties. We obtained vocabularies used in t from the *void:vocabulary* property. We uncovered links of t from Linkset descriptions associated with t that occur in V. A *void:Linkset* describes a set of triples (s, p, o) that link resources from two datasets through a property p. The *void:subjectsTarget* property designates the dataset of the subject s and the *void:objectsTarget* property indicates the dataset of the object o.

We also extracted links via the catalog API, which exposes a multivalued property, *relationships*, whose domain and range is the complete set of catalogued datasets. In this case, assertions of the form "$t[relationships] = \_node$" and "$\_node[object] = u$" indicate that t is linked to a dataset u.

4.3 Testing Strategy

To evaluate the performance of the rank score functions, we adopted the traditional 10-fold cross validation approach, where a *testing set* is randomly partitioned into 10 equally-sized subsets and the testing process is repeated ten times, each time using a different subset as a *testing partition* and the rest of the objects in the testing set as a *training partition*.

In our experiments, the 295 datasets obtained from the DataHub catalog played the role of the testing set. The 10-fold cross validation then generated 10 different pairs $(\boldsymbol{T}_i, \boldsymbol{D}_i)$, for $i = 1, ..., 10$, of testing and training partitions. The known links between datasets in $\boldsymbol{D}_i$ were preserved, those between datasets in $\boldsymbol{T}_i$ were ignored, and those from datasets in $\boldsymbol{T}_i$ to $\boldsymbol{D}_i$ were used as the gold standard for the datasets in $\boldsymbol{T}_i$. Each test consisted of computing the MAP for the pair $(\boldsymbol{T}_i, \boldsymbol{D}_i)$. Then, we computed the overall average of the MAPs for the 10 tests, referred to as the *overall MAP* in Section 4.4.

We used the training partition to estimate probabilities, using Equations 10 and 11, when testing the Bayesian approach, and to construct the Linked Data network, when testing the Social Network-based approach.

4.4 Results

This section describes the experiments we conducted to evaluate the rank score functions generated by the two approaches presented in Section 3, referred to as the Bayesian approach and the Social Network-based (SN-based) approach

Table 3. Overall Mean Average Precision

Approach	*Feature set*			
	properties	*classes*	*vocabularies*	*all*
SN-based	48.46%	**57.18%**	48.27%	51.57%
Bayesian	59.18%	55.31%	51.20%	**60.29%**

(using the rank score function defined in Equation 15). We combined each of the approaches with the following feature sets: (i) only properties; (ii) only classes; (iii) only vocabularies; and (iv) all these three features.

Table 3 depicts the overall MAP results obtained by each combination of approach and feature set. The Bayesian approach using all three features achieved the best performance; the Bayesian approach using properties obtained the second best result; and the SN-based approach using classes was the third best result. In fact, the Bayesian approach obtained better results than the SN-based approach using properties or vocabularies as single features. The worst results obtained by both approaches used vocabularies as a single feature. This probably happened because, in our experiments, we have a restrict number of references to vocabularies in the datasets.

We also calculated the overall MAP of the rank score functions based only on preferential attachment (pa) and resource allocation (ra), using classes as single features. We respectively obtained 43.64% and 44.75%, which are lower than the overall MAP for the rank score function defined in Equation 15.

Finally, we applied a paired T-test to investigate whether there are statistically significant differences between the overall MAP results of the different approaches and selected feature sets. Table 4 shows the p-values obtained by all T-tests performed, where the results is boldface represent differences which are not statistically significant.

The T-test of the SN-based approaches indicate that the SN-based approach using the rank score function defined in Equation 15 and using classes as features outperforms the SN-based approaches using preferential attachment (pa) or resource allocation (ra) and classes as features.

A T-test was also performed for overall MAP results of the SN-based approaches using classes and using the other feature selections. The T-tests indicate that the SN-based approach using the rank score function defined in Equation 15 and classes achieved a statistically significant improvement when compared to all others (using properties, vocabularies and all features). Thus, there are evidences that classes are the best feature selection to be used with the SN-based approach.

For the Bayesian approach, we compared the results obtained by using all features (the configuration with the best overall MAP) with the results obtained using all other feature selections. The T-tests indicate that the overall MAP results of the Bayesian approach using all features and using only properties do not present a statistically significant difference. This suggests that using only properties is an adequate strategy to be adopted with the Bayesian approach.

Table 4. The p values applying T-test

SN-based with *classes*	SN-based with			*pa*	*ra*
	properties	*vocabularies*	*all* features		
	5.26E-05	0.00195	0.03683	5.46E-08	1.35E-05
Bayesian with *all* features	Bayesian with			SN-based with *classes*	
	properties	*classes*	*vocabularies*		
	0.10641	0.00408	0.00022	**0.07275**	

We also used a paired T-test to investigate whether there is a statistically significant difference between the overall MAP values obtained by the best configuration for the SN-based approach (using classes) and the best configuration for the Bayesian approach (using all features). The T-tests indicate that there is no statistical difference between the overall MAP results of both approaches.

In conclusion, these observations indicate that the SN-based approach using classes or the Bayesian approach using properties induce the best rank score functions, since they achieve the best results and are simple to compute. This is the main result of the paper.

5 Conclusions

This paper compared two approaches respectively based on Bayesian classifiers and on Social Network Analysis techniques to address the dataset interlinking recommendation problem. Both approaches define rank score functions that explore only metadata features - vocabularies, classes and properties - and the known dataset links. The results show that the rank score functions achieve a mean average precision of around 60%. This means that a dataset interlinking tool might use the rank score functions to limit the search for links from a dataset t to just the top ranked datasets with respect to t and yet find most of the links from t. Thus, the rank score functions are potentially useful to reduce the cost of dataset interlinking.

The computation of the rank score functions depends on harvesting metadata from Linked Data catalogs and from the datasets themselves, a problem shared by other Linked Data techniques, but they are not restricted using only VoID descriptions. This limitation in fact calls attention to the importance of harvesting metadata, that can be carried out in different ways, including the inspection of the datasets by crawlers, a problem we address elsewhere [22], to fulfill the Linked Data promises.

Finally, we plan to further improve the definition of the rank score functions. One generic strategy is to improve the network analysis-based score by considering the frequency of the schema elements. Often two datasets share similar classes and properties, but they strongly differ on the number of instances. Another aspect to explore would be feature similarity (e.g., string similarity between two features), rather than just considering the intersection of the feature sets.

Acknowledgments. This work was partly funded by the LinkedUp project (GA No:317620), under the FP7 programme of the European Commission, by CNPq, under grants 160326/2012-5, 303332/2013-1 and 557128/2009-9, by FAPERJ, under grants E-26/170028/2008, E-26/103.070/2011 and E-26/101.382/2014, and by CAPES, under grant 1410827.

References

1. Berners-Lee, T.: Linked Data. In: Design Issues. W3C (July 2006)
2. Leme, L.A.P.P., Lopes, G.R., Nunes, B.P., Casanova, M.A., Dietze, S.: Identifying candidate datasets for data interlinking. In: Daniel, F., Dolog, P., Li, Q. (eds.) ICWE 2013. LNCS, vol. 7977, pp. 354–366. Springer, Heidelberg (2013)
3. Lopes, G.R., Leme, L.A.P.P., Nunes, B.P., Casanova, M.A., Dietze, S.: Recommending tripleset interlinking through a social network approach. In: Lin, X., Manolopoulos, Y., Srivastava, D., Huang, G. (eds.) WISE 2013, Part I. LNCS, vol. 8180, pp. 149–161. Springer, Heidelberg (2013)
4. Nikolov, A., d'Aquin, M.: Identifying Relevant Sources for Data Linking using a Semantic Web Index. In: WWW2011 Workshop on Linked Data on the Web, Hyderabad, India. CEUR Workshop Proceedings, vol. 813. CEUR-WS.org (March 29, 2011)
5. Nikolov, A., d'Aquin, M., Motta, E.: What Should I Link to? Identifying Relevant Sources and Classes for Data Linking. In: Pan, J.Z., Chen, H., Kim, H.-G., Li, J., Wu, Z., Horrocks, I., Mizoguchi, R., Wu, Z. (eds.) JIST 2011. LNCS, vol. 7185, pp. 284–299. Springer, Heidelberg (2012)
6. Kuznetsov, K.A.: Scientific data integration system in the linked open data space. Programming and Computer Software 39(1), 43–48 (2013)
7. Mühleisen, H., Jentzsch, A.: Augmenting the Web of Data using Referers. In: WWW2011 Workshop on Linked Data on the Web, Hyderabad, India. CEUR Workshop Proceedings, vol. 813. CEUR-WS.org (March 29, 2011)
8. Lóscio, B.F., Batista, M., Souza, D.: Using information quality for the identification of relevant web data sources. In: The 14th International Conference on Information Integration and Web-Based Applications & Services, IIWAS 2012, Bali, Indonesia, December 3-5, pp. 36–44. ACM, New York (2012)
9. Wagner, A., Haase, P., Rettinger, A., Lamm, H.: Discovering related data sources in data-portals. In: Proceedings of the First International Workshop on Semantic Statistics, Co-located with the the International Semantic Web Conference (2013)
10. de Oliveira, H.R., Tavares, A.T., Lóscio, B.F.: Feedback-based data set recommendation for building linked data applications. In: I-SEMANTICS 2012 - 8th International Conference on Semantic Systems, I-SEMANTICS 2012, Graz, Austria, September 5-7, pp. 49–55. ACM (2012)
11. Toupikov, N., Umbrich, J., Delbru, R., Hausenblas, M., Tummarello, G.: Ding! dataset ranking using formal descriptions. In: Proceedings of the WWW2009 Workshop on Linked Data on the Web, LDOW 2009, Madrid, Spain. CEUR Workshop Proceedings, vol. 538. CEUR-WS.org (April 20, 2009)
12. Witten, I.H., Frank, E., Hall, M.A.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann (January 2011)
13. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press (2002)

14. Lü, L., Jin, C.H., Zhou, T.: Similarity index based on local paths for link prediction of complex networks. Physical Review E 80(4), 046122 (2009)
15. Liben-Nowell, D., Kleinberg, J.: The link-prediction problem for social networks. J. Am. Soc. Inf. Sci. Technol. 58(7), 1019–1031 (2007)
16. Caraballo, A.A.M., Nunes, B.P., Lopes, G.R., Leme, L.A.P.P., Casanova, M.A., Dietze, S.: Trt - a tripleset recommendation tool. In: Proceedings of the ISWC 2013 Posters & Demonstrations Track, Sydney, Australia. CEUR Workshop Proceedings, vol. 1035, pp. 105–108. CEUR-WS.org (October 23, 2013)
17. Taibi, D., Dietze, S.: Proceedings of the LAK Data Challenge, Leuven, Belgium, April 9. CEUR Workshop Proceedings, vol. 974. CEUR-WS.org (2013)
18. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval - the concepts and technology behind search, 2nd edn. Pearson Education Ltd., Harlow (2011)
19. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press (July 2008)
20. Hull, D.: Using statistical testing in the evaluation of retrieval experiments. In: Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1993, pp. 329–338. ACM, New York (1993)
21. Alexander, K., Cyganiak, R., Hausenblas, M., Zhao, J.: Describing Linked Datasets with the VoID Vocabulary. W3C (March 2011)
22. do Vale Gomes, R., Casanova, M.A., Lopes, G.R., Leme, L.A.P.P.: A metadata focused crawler for linked data. In: Proceedings of the 16th International Conference on Enterprise Information Systems, ICEIS 2014, Lisbon, Portugal, April 27-30, vol. 2, pp. 489–500. SciTePress (2014)