DBtrends : Publishing and Benchmarking RDF Ranking functions

Edgard Marx[1], Amrapali Zaveri[2], Mofeed Mohammed[1], Sandro Rautenberg[1], Jens Lehmann[3,4], Axel-Cyrille Ngonga Ngomo[1], and Gong Cheng[5]

[1] University of Leipzig, Institute of Computer Science, AKSW Group
`{marx|mofeed|rautenberg|ngonga}@informatik.uni-leipzig.de`
[2] Stanford Center for Biomedical Informatics Research, Stanford University, USA.
`amrapali@stanford.edu`
[3] Computer Science Institute, University of Bonn
`jens.lehmann@cs.uni-bonn.de`
[4] Fraunhofer IAIS
`jens.lehmann@iais.fraunhofer.de`
[5] State Key Laboratory for Novel Software Technology, Nanjing University, China
`gcheng@nju.edu.cn`

Abstract. Providing accurate approaches for keyword search or question answering to access the data available on the Linked Data Web is of central importance to ensure that it can be used by non-experts. In many cases, these approaches return a large number of results that need to be provided in the right order so as to be of relevance to the user. Achieving the goal of improving the access to the Linked Data Web thus demands the provision of ranking approaches that allow sorting potentially large number of results appropriately. While such functions have been designed in previous works, they have not been evaluated exhaustively. This work addresses this research gap by proposing a formal framework designed towards comparing and evaluating different ranking functions for RDF data. The framework allows combining these rankings by means of an extension of the Spearman's footrule estimation of the upper bound of this function. We supply a benchmark with a total of 60 manually annotated entity ranks by users from USA and India recruited over Amazon Mechanical Turk. Moreover, we evaluated nine entity ranking functions over the proposed benchmark.

1 Introduction

A large number of applications rely on ranking methods for querying, browsing, linking and presenting RDF data. Examples of such applications are Search Engines [2], Linked Data browsers [5], Link Discovery [8] and Machine Learning [13]. Therefore, using the right ranking method is important to achieve good results. A large number of ranking approaches for RDF resources have thus been developed [1,3,4,12]. Providing a comparison of these approaches has however been paid little attention to. This is due to two main reasons: First, creating benchmarks for ranking is a tedious and costly task. More importantly, approaches for

comparing rankings such as Spearman's footrule[6] [11] assume that the rankings to compare are permutations of the same set. We address both problems by providing a formal framework for evaluating and publishing ranking functions. This work has the following contributions:

– A rank similarity function for comparing rankings that do not cover the same set (heterogeneous rankings);
– A benchmark[7] for ranking functions over DBpedia knowledgebase;
– A public API[8] and library[9] for evaluation and easy integration of ranking functions.

The rest of the paper is structured as follows. The related work is discussed in Section 2. Thereafter, we introduce a variation of the Spearman's footrule for evaluating heterogeneous ranks in Section 3. The evaluation, results and performance of different ranks for entities are presented respectively in Section 4, Section 5. Finally, we conclude with our plans for future work in Section 6.

2 Related Work

During the last years, ranking algorithms have started to become more personalized. This means that instead of using only the data structures themselves, approaches have begun to use third-party information, i.e. information that cannot be found in the data itself. For instance, search engines use the previous query terms to rank potential results. Another valuable third-party information can be the users' previously visited web sites and their frequency. That information can help enhance the rank of the query results. Thus, ranks can be divided into two categories: dynamic and static.

Static ranks are those that can be derived from a particular data structure or information and do not change. That is the case of Page-Rank [2], DBpedia Page-Rank [12] (DB-RANK), RELIN [3] and Google Trends[10].

Dynamic ranks change according to a given third-party information. Examples of such ranks are CHR, DFF, CNN and COMB introduced by *Cheng et al.* [4]. The CHR, DFF, CNN and COMB are designed for the task of entity linking and use the target text for ranking the possible linking candidates. Another example of dynamic rank is LDRANK [1], a query-biased algorithm for ranking RDF resources. LDRANK uses a combination of explicit and implicit relationship inferred from RDF resources. The explicit relationship is extracted through a PAGE-RANK like algorithm applied to the RDF graph. The implicit relation is inferred from the text of the resource web page. However, static ranks

---

[6] Note: not to be confused with the more well-known Spearman's rho

[7] http://benchmark.dbtrends.org

[8] http://dbtrends.org/api/entities/get?db=dbpedia&v=3.9&resources=http://dbpedia.org/resource/Leipzig,http://dbpedia.org/resource/Berlin&encode=jsonld

[9] http://dbtrends.org

[10] https://www.google.com/trends/

are the basis of a wide range of applications such as Search Engines [2], Linked Data browsers [5], Link Discovery [8] and Machine Learning [13]. In the following we discuss the related work in four parts: (1) Ranking, (2) Dataset Statistics (3) Rank Similarity Functions.

2.1 Rank Similarity Functions

Ranks are sequence of similar elements, sorted in a particular order. The problem of measuring ranking similarities can be focused on finding how distant or close two ranks are from each other by comparing the order of their elements. In order to find how similar two ranks are from each other, there are two approaches: Spearman's Footrule [11] and Kendall rank correlation coefficient [6] —usually referred as Kendall's tau coefficient. Both similarity functions are designed for measuring the distance among ranks containing the same set of elements. Spearman's Footrule measures the distance between an element belonging to two different ranks. Kendall's tau coefficient computes the number of swaps (*Bubble sort*) operations necessary to sort the first rank accordingly to the second. However, the correlation between two ranks (r_β, r_γ) of Spearman's Footrule (S_F) is bounded by Kendall's tau coefficient [7] (K), that is, $\forall r_\beta, r_\gamma \ K(r_\beta, r_\gamma) \leq S_F(r_\beta, r_\gamma) \leq 2K(r_\beta, r_\gamma)$.

2.2 Ranking

Ranking methods have been studied for a long time as they are useful for measuring the relevance of a certain feature. Ranking methods can be dynamic or static. Ranking algorithms for RDF data are usually designed for three main features: (1) entities (objects or individuals), (2) properties and (3) classes.

Ranking for entities is the most common type of ranking available. For instance, a query "persons" can return thousands of entities if applied to the DBpedia knowledge base, but not all the information can be useful. In this way, entity ranks can help search engines sort the resulting set according to its relevance.

Page-Rank [2] is based on the probability of randomly finding a page in a network by following a path starting from any other page. The concept of Page-Rank can be applied to any graph network. For instance, *DBpedia Page-Rank* [12] is a variant of the original Page-Rank algorithm where the rank of a DBpedia entity corresponds to the rank of its Wikipedia page.

Despite that, entities can have a large number of properties, but a big portion of them might not be interesting to users. To deal with this problem of so-called as *entity summarization*, approaches implementing different types and levels of abstractions were introduced. *RELIN* [3] is a ranking function that explores a variant of the random surfer model revised by a more specific notion of centrality designed for property ranking. The authors also implemented a baseline called *RandomRank*, which trivially generates a random ranking of property-value pairs. Also, Roa-Valverde et al. [10] provides a systematic review on ranking approaches for the Web of Data.

2.3 Dataset Statistics

Another type of measure that can be used for ranking is the dataset statistics. Some of the dataset statistics are, for example, the number of instances of a certain resource. Another set of information can be the number of references, predicates or outgoing links. These statistics are specifically useful for ranking entities.

However, as the Semantic Web usually deals with real world entities, some approaches have introduced rankings using statistics coming from external sources [3]. That is, those statistics are outside the dataset. This approach can be applied to knowledge bases because an entity usually refers to a real world resource. Thus, for instance, one can extract statistics related to the resource's web page —i.e. the PageRank. This possibility opens new perspectives for RDF ranking.

Currently there are many available search engines that are able to crawl big portions of the Web as Google, Yahoo and Bing. Beside that, they can find related content to a give query. By crawling a large volume of the Web, the search engines can also became a big source of information that can help when ranking resources. One source of information can be, for instance, the number of available Web documents with a particular term or sentence. Another source of information can be the query log. For instance, *Google Trends*[10] is a public web platform containing the historical search index of a particular term in Google Knowledge Graph topics, Search interest, trending YouTube videos, and Google News articles. The Google Trends index is based on how often a particular term is searched in relation to the total search-volume across various regions of the world, and in various languages. The Google Trends index can be used to either rank entities, properties, and their correlation.

3   A Heterogeneous Rank Similarity Function

As discussed before, Spearman's Footrule is a distance measure function designed to measure similarities among homogeneous rankings. Here in, we propose a new rank similarity function measure based on Spearman's Footrule for measuring similarity among heterogeneous rankings. That is, rankings that have different set of resources.

According to Spearman's Footrule, the similarity between two ranks is measured by a summation of the difference among the positions of the element e in the two ranks r_β and r_γ. The Spearman's Footrule is formally defined by the function $S_F(r_\beta, r_\gamma) = \sum_{i=1, f_\beta = r_\beta(i)}^{|r_\beta|} | r_\beta^{-1}(f_\beta) - r_\gamma^{-1}(f_\beta)|$.

However, to measure how similar a rank is from each other, it is necessary to compute the maximal distance between the two ranks. The maximal distance between two ranks in Spearman's Footrule can be obtained by induction in a very trivial process and will not be discussed here. However, the maximal distance between two ranks is given by the function $s_{f_{max}}$ that receives a rank size and computes the maximal distance. Herein, we define the length of a list as a natural ex-

tension of a set cardinality denoted by $|r|$. The Spearman's Footrule maximal distance of a rank is given by $S_{F_{max}}(|r|) = \begin{cases} 2(\frac{|r|}{2})^2 & \text{if } |r| \bmod 2 = 0 \\ 2(\frac{|r|-1}{2})^2 + (|r|-1) & \text{else } |r| \bmod 2 = 1 \end{cases}$.

The intuition behind the extension. Let A and B be two rankings, and, let $C = A/B$ and $D = B/A$. The idea behind our extended rank similarity function is to devise a function that is the upper bound of any possible ranking that (1) conforms to A and B and (2) contains all elements of $A \cup B$. The Spearman's Footrule is very powerful for measuring homogeneous ranks. However, imagine a scenario where there are two ranks $A = \{a, b, c\}$ and $B = \{a, b\}$. The problem is that, in this case, the Sperman's Footrule for measuring the distance between A and B is not defined. Thus, if we apply the Sperman's Footrule to the given sets, it might return zero, even though there is a visible difference between the two ranks. The problem is that c is not in B. However, by using the extension the difference is clear, $d(A, B) = 3$. That is, there is one element with the distance of three which does not belong to one of the sets, which is the difference among them. Furthermore, the distance is symmetric, that is, $d(A, B) = d(B, A)$.

Now, let's discuss a more complex example. Imagine that we have a rank $c = \{d\}$. By definition $d(A, C) = d(C, A)$, but what is the reason behind it? A naive thinking can lead to imagine that there is a missing information as only the elements in A are contributing to the distance between A and C. The real reason behind the distance between A and C is that the disjoint elements in C contribute as much to the distance as the disjoint elements in A. As A is bigger than C, the distance of a disjoint element $c_{\not A}$ of C in A is $distance(c_{\not A}) = (\sum_{a \notin C \cap A} A(a))/(\sum_{c \notin C \cap A} 1)$. In a few words, while the distance of the element $c \in C$ with relation to A is the summation of the position of the elements in A, the distance of an element in A to C is its position. Thus, the distance of the disjoint elements in the smallest set is always equal to the disjoint elements in the biggest set. That is, $\sum_{c \notin C \cap A} distance(c) = \sum_{a \notin C \cap A} A(a)$. To simplify, the proposed Sperman's Footrule extension uses the summation of the disjoint elements in the biggest set. The usability of the proposed extension will be shown in Section 4.

To overcome the problem of measuring heterogeneous ranks, we propose a variant of Spearman's Footrule. The difference from the original formula is that it consists of the sum of the position of the element of the highest rank that does not intersect, which can be formally defined as follows:

$$
\begin{aligned}
F(r) =& (F|r = (f_1, f_2, ..., f_n) : f \in F) \\
D(r_\beta, r_\gamma) =& D_\cap(r_\beta, r_\gamma) + D_{\not A}(r_\beta, r_\gamma) \\
D_\cap(r_\beta, r_\gamma) =& \sum_{f_\beta \in F(r_\beta) \cap F(r_\gamma)} |\, r_\beta^{-1}(f_\beta) - r_\gamma^{-1}(f_\beta)| \\
D_{\not A}(r_\beta, r_\gamma) =& \begin{cases} \sum_{f_\beta \notin F(r_\beta) \cap F(r_\gamma)} r_\beta^{-1}(f_\beta) & \text{else if } |r_\beta| > |r_\gamma| \\ \sum_{f_\gamma \notin F(r_\beta) \cap F(r_\gamma)} r_\gamma^{-1}(f_\gamma) & \text{, otherwise} \end{cases}
\end{aligned}
\tag{1}
$$

The extension of the original Spearman's Footrule formula makes the measurement of the maximal distance between two ranks more complex and it can be divided into three cases: (1) when the ranks are homogeneous, (2) when they intersect and (3) when they do not intersect. The simplest cases are the homogeneous and the without intersection ones. When the ranks are homogeneous, the distance can be measured by the function $S_{F_{max}}$. In other cases, the distance is given by an arithmetic progression of the size of the biggest rank. The arithmetic progression is used as an anchor because an arithmetic progression of a rank with n entries is higher than the maximal distance of Spearman's Footrule $\sum_{i=1}^{|r|} i > S_{F_{max}}(|r|)$. Apart from that, the similarity function for heterogeneous ranks uses the position of the element in the biggest rank. The biggest distance between two ranks occurs when they do not have elements in common. Thus, a maximal distance between two ranks is defined by the function

$$D_{max}(r_\beta, r_\gamma) = \begin{cases} S_{F_{max}}(|r_\beta|) & \text{if } r_\beta \equiv r_\gamma \\ D_{\mathcal{D}}(r_\beta, r_\gamma) & \text{, otherwise} \end{cases}.$$

4 Evaluation

In this section, we describe the evaluation performed for benchmarking nine different ranking functions: (1) the in (DB-IN) and (2) out (DB-OUT) degree of a resource in the dataset; the (3) in (PAGE-IN) and (4) out degree (PAGE-OUT) of the resource's Wikipedia page; the (5) DBpedia page-rank (DB-RANK); the (6) number of external links pointing to the resource's Wikipedia web page (E-PAGE-IN); the (7) Page Authority measured by SEO (SEO-PA); the (8) Wikipedia Page-Rank (PAGE-RANK); the (9) social shared links (SHARES-LINKS); and the (10) best entity rank (Best). We first describe the tasks followed by details of the crowdsourcing experiment (workers and wage). We chose to use crowdsourcing because it facilitates finding the target audience. Thereafter, we measure the rank distances and finally, we provide further details about the implementation and benchmark.

The evaluation was designed to answer the following research questions:

- How different are the rankings performed across different countries?
- Is there any similarity between the ranking performed by different users?
- Which of the entity ranking functions performs better?
- Is there any similarity between the entity ranking performed by a particular ranking function in any particular type of resource?

In previous works [3,4], the ranks were evaluated by using best rank selection. That is, first different ranking functions are applied to a target data producing different ranks. The produced ranks are then shown to humans, who select the most relevant one.

In our methodology, at first, the resources that are going to be evaluated are selected based on the connectivity. That is the number of incoming and outgoing edges. Thereafter, those resources are used to create ranking tasks to be executed by workers from different locations and crowdsourcing platforms (e.g. Amazon

Mechanical Turk). The tasks consist of sorting the resources according to their relevance, generating rank profiles. These profiles are then used for benchmarking different ranking functions. We have published the ranking functions[9] and profiles[7] so that anyone can easily use and benchmark their applications.

Finding the top four classes. In order to find the top four most used classes, the DBpedia classes were sorted taking into consideration the number of instances. By doing so, the top first 11 classes sorted in descending order of number of instances were dbo:Agent, dbo:Person, dbo:Place, dbo:CareerStation, dbo:PopulatedPlace, dbo:Settlement, dbo:Work, dbo:Organization, dbo:Athlete, dbo:SportTeamMembe, dbo:OrganizationTeamMember and dbo:Species. Thereafter, we discarded classes that (1) were a super-class of a more specific class (e.g. dbo:Agent is a super-class of dbo:Person) or (2) were not a sub-class but have an overlapping with any previous taken class (e.g. dbo:SportTeamMember is not an dbo:Athlete but overlaps it). We started excluding the classes based on these two criteria from the first top ranked class until we reached the top four.

Task description. As discussed in Section 2, there are different ranking measures for entities. Ranking measures are a mix of statistics often found inside the datasets (e.g. number of instances of a resource) as well as outside them (e.g. PageRank) and they can be useful in a wide range of applications such as Search Engines [2], Linked Data browsers [5], Link Discovery [8] and Machine Learning [13]. The evaluation was designed to measure different ranking functions across different countries.

In order to evaluate the different ranking functions, we designed four tasks using the DBpedia knowledge base. In Task 1, the worker was asked to sort 20 entities, which were displayed with their label and description. The entities were extracted from the top five entities of the top four DBpedia classes (dbo:Settlement, dbo:Organization, dbo:Athlete and dbo:Species), according to their relevance (i.e. number of instances).

In Tasks 2 and 3, the user was asked to sort the top 20 most instantiated classes and predicates respectively. These classes and predicates were extracted from the highest ranked entity chosen by the user in Task 1 as the most relevant. In this manner, there is a high probability that the user is more familiar with the entitys' classes and predicates and performs better ranking.

Moreover, in all the ranking tasks the users could select one or more resource to be ranked. As can be noticed, the possibility to selecting any particular resource(s) can generate a large number of ranks with different resources and sizes. Nonetheless, this process can ensure that the generated rank is more likely to produce better results since this method allows the user to rank resources that she believes is relevant. For instance, a possible rank for a given list of resources $l = (a, b, c)$ is $r_\gamma = (a, b)$.

Finally, in Task 4, the users were asked to score her confidence in performing the previous ranking tasks between one and five, where five is that she was most confident. This task had two aims: (1) to validate the performed tasks and (2) point to possible weakness and improvements towards a better benchmark. For

instance, a large number of workers showing poor confidence in performing the task could indicate that the task should be reformulated.

The class dbo:CareerStation was discarded because it refers to a state in a period of time rather than an entity itself. The classes dbo:Place and dbo:PopulatedPlace are super class of dbo:Settlement, thus they were also discarded. The class dbo:Work was removed as it could lead to a misunderstanding of the class referring to an occupation whereas it actually refers to creative works and products. Thus, after discarding these classes, the five top classes obtained were: dbo:Agent, dbo:Person, dbo:Organization, dbo:Athlete and dbo:SportTeamMember. By applying the criteria of discarding classes that are a super-type of a more specific type, the class dbo:Agent and dbo:Person were further removed. The class dbo:Agent was removed because it is a super type of class dbo:Person and the class dbo:Person a super type of dbo:SportTeamMember. The class dbo:SportTeamMember overlaps with the classdbo:Athlete since a team member can also be an athlete. By applying the second constraint of ignoring classes that are not sub-classes but overlapping classes with more instances, the type dbo:SportTeamMember was discarded because dbo:Athlete has more instances. The same constraint applies to *dbo:OrganizationTeamMember* regarding dbo:Athlete. Thus, dbo:OrganizationTeamMember is replaced by dbo:Species. Finally, the remaining four classes are: dbo:Settlement, dbo:Organization, dbo:Athlete and dbo:Species.

Task Execution. The tasks were performed using Amazon Mechanical Turk[11]. The workers were instructed to consult any source of information available in order to execute the given tasks such as a dictionary and/or the internet.

The Workers. The tasks were executed by a total of 60 users, out of which 30 were North Americans and 30 Indians, double the amount of users commonly used in similar tasks for rank evaluation in previous works [3,4]. The workers were partitioned among distinct countries in order to evaluate differences across their evaluations. Specifically Indians and North Americans were chosen as they represent the two major groups amongst the Amazon Mechanical Turk workers [9]. They were paid a wage at the rate of 1.5$ for 25 minutes.

Measuring Rank Distances. Due to the heterogeneity of the ranks produced by the users, the rank distances were measured by the heterogeneous rank similarity function discussed in Section 3. The results achieved by all experiments are presented in Section 5.

Implementation & Benchmark. All the users evaluations are available online[7]. The rankings can also be accessed via a library or REST API over DBtrends[8]. DBtrends is an open-source project and can be easily deployed in existing applications.

---

[11] https://www.mturk.com

5 Results

Table 1 display the general results achieved by the evaluation of entities (R^e).
The table contain (1) the average distance between the rank sample data R and
different ranks r, $\overline{D_R}(R,r)$; (2) the standard deviation $\sigma_{D_F}(R)$; (3) the median
$\widetilde{D_F}(R)$; (4) the average rank size $\overline{R}$ as well as (5) the average maximum distance
of the samples per country $\overline{D_{R_{max}}}(R)$. The average confidence of each country
in performing the tasks is displayed in Table 1. Considering R a rank set and r
a rank, formally R, $D_F(R)$, $D_R(R,r)$ and $D_{R_{max}}(R)$ are defined as follows:

$$
\begin{aligned}
R =& \bigcup r \\
D_F(R) =& \{d_{D_F} \mid \forall r_\beta, r_\gamma \in R, \ d_{D_F} = D(r_\beta, r_\gamma)\} \\
D_R(R,r) =& \{d_{D_R} \mid \forall r_\beta \in R, d_{D_R} = D(r_\beta, r)\} \\
D_{R_{max}}(R) =& \{d_{D_{R_{max}}} \mid \forall r_\beta, r_\gamma \in R, d_{D_{R_{max}}} = D_{max}(r_\beta, r_\gamma)\}
\end{aligned}
\tag{2}
$$

Table 2 shows the average rank of each entity per country as well as their
combined values. Table 1 displays the results of nine different entity ranks (r^e)
applied to the entity rank sample data (R^e): (1) the incoming (DB-IN) and
(2) outgoing links (DB-OUT) of a resource in the dataset; the (3) incoming
(PAGE-IN) and (4) outgoing (PAGE-OUT) links of the resource's Wikipedia
page; the (5) DBpedia page-rank (DB-RANK) [12]; the (6) number of external
incoming links to the resource's Wikipedia page (E-PAGE-IN); the (7) Page
Authority measured by SEO (SEO-PA)[12]; the (8) Wikipedia Page-Rank (PAGE-
RANK); the (9) social shared links (SHARED-LINKS); and the (10) the distance
achieved by the best entity rank (r^e) combination. The best entity rank (Best)
is the average rank of the entity based on its rank in each individual profile.
Table 2) display the average rank of each entity ($\overline{Rank}$). The E-PAGE-IN, SEO-
PA, PAGE-RANK, SHARED-LINKS where measured by SEO review tools[13].

The results in Table 1 show that, on average, Indians were ~40% more con-
fident than Americans in performing the ranking tasks. However, the internal
agreement for entities was much higher for Americans. The median $\widetilde{D_F}(R^e)$ and
average max distances $\overline{D_{R_{max}}}(R^e)$ among the entity ranks of Americans where
respectively ~44% and ~35% higher than the Indians. The same pattern did
not apply to the property ranks where the differences were not tangible. The
Indians achieved an internal rank agreement ~2% higher than Americans when
comparing the average maximal distance $\overline{D_{R_{max}}}(R^e)$ among them. The results
also show that Americans found all entities relevant whereas Indians found only
16 (out of the 20 total entities).

Regarding the measured entity ranks, the PAGE-RANK achieved the best
result and is followed close by PAGE-IN, DB-IN, E-PAGE-IN, SEO-PA and
SHARED-LINKS. The PAGE-RANK achieved an entity rank only ~5% higher
than the ideal rank (Best). However, an interesting note is that the results

---

[12] https://moz.com/learn/seo/page-authority
[13] http://www.seoreviewtools.com/

Country(R^e)	USA	India	AVG
$\sigma_{D_F}(R^e)$	26.00	47.00	36.50
$\overline{D_{R_{max}}}(R^e)$	203.00	311.68	257.34
$\widetilde{D_F}(R^e)$	96.00	144.00	120.00
$|r^e|$	20.00	16.33	18.16
$Confidence(\%)$	0.55	0.90	0.72
$\overline{D_R}(R^e, r^e)$ DB-IN	98.96	143.63	146.54
DB-OUT	104.68	132.70	143.94
PAGE-IN	93.79	136.23	140.26
PAGE-OUT	111.37	135.90	144.88
DB-RANK	107.51	145.23	151.62
E-PAGE-IN	99.13	129.18	114.15
SEO-PA	102.86	134.18	118.52
PAGE-RANK	94.57	129.58	112.07
SHARES-LINKS	101.97	126.75	112.36
Best	87.46	126.38	106.92

Table 1. Average rank similarity for entities. The table above presents statistics (1) and ranks (2) as follows. The statistics (1) are: (1.1) the average distance between the entity rank sample data R^e and different entity ranks r^e, $\overline{D_R}(R^e, r^e)$; (1.2) the standard deviation, $\sigma_{D_F}(R^e)$; (1.3) the median $\widetilde{D_F}(R^e)$; (1.4) confidence; (1.5) the average rank size $\overline{R^e}$ as well as (1.6) the average maximum distance of the samples per country, $\overline{D_{R_{max}}}(R^e)$. The ranks (2) includes: (2.1) the in (DB-IN) and (2.2) out degree (DB-OUT) of a resource in the dataset; the (2.3) in (PAGE-IN) and (2.4) out degree (PAGE-OUT) of the resource's Wikipedia page; the (2.5) DBpedia page-rank (DB-RANK); the (2.6) number of external links pointing to the resource's Wikipedia web page (E-PAGE-IN); the (2.7) Page Authority measured by SEO (SEO-PA); the (2.8) Wikipedia Page-Rank (PAGE-RANK); the (2.9) social shared links (SHARES-LINKS); and the (2.10) best entity rank (r^e) .

achieved by each entity rank are sparse when comparing the countries individually. For instance, the best rank for Americans according to the results is PAGE-RANK while for Indians is SHARED-LINKS.

The distribution of the top first entities among countries in Table 2, produced interesting results. For instance, the first four top entities (#Top-1) amongst India and USA were the same (dbr:New_York_City, dbr:Los_Angeles, dbr:Animal and dbr:Political_divisions_of_the_United_States). This result is interesting because cities such as New York and Los Angeles do not have as much as influence in Indian history as London (dbr:London) that barely appears in the sixth position behind Chicago (dbr:Chicago), another city in USA. Furthermore, Chicago is not even top first for any of the Americans. However, when comparing the average rank of Indians, London appears at third place.

Another interesting observation is that dbr:Animal is chosen as top first for 13 Americans, in contrast of merely four of the Indians. This difference might have to be influenced by the American engagement in nature preservation. This also explains the occurrence of dbr:Lepidoptera as the most important entity for some users. However, this finding is not observed when comparing average results. For instance, dbr:Plant appears in first position for Indians and dbr:Animal at ninth, where for Americans dbr:Animal appears at second and dbr:Plant at

fifth. Moreover, the top first results of the Americans are less sparse than for the Indians. The top first entity of the Americans is devised among eight entities against 13 of the Indians. Finally, the average rank similarity among the different users ranks (internal agreement) for entities is ∼63%.

Entity	USA			India			Combined		
	AVG	#Rank	#Top-1	AVG	#Rank	#Top-1	AVG	#Rank	#Top-1
dbr:New_York_City	26.44	1	5	22.36	1	6	24.37	1	11
dbr:London	23.31	7	0	21.20	3	2	22.23	3	2
dbr:Los_Angeles	24.58	4	3	21.76	2	3	23.15	2	6
dbr:Paris	23.24	6	0	20.36	4	1	21.77	5	1
dbr:Chicago	25.00	3	0	18.90	7	3	21.89	4	3
dbr:Plant	24.37	5	0	15.80	11	1	20.01	10	1
dbr:Animal	25.65	2	13	17.76	9	4	21.64	6	17
dbr:Arthropod	17.79	12	0	15.96	10	1	16.86	12	1
dbr:Lepidoptera	16.86	16	1	14.46	18	1	15.64	17	2
dbr:Roger_Federer	17.68	13	0	15.43	13	1	16.54	13	1
dbr:Serena_Williams	17.10	15	0	15.06	14	0	16.06	15	0
dbr:Rafael_Nadal	16.58	17	0	14.56	17	0	15.55	18	0
dbr:Mollusca	18.37	11	0	15.53	12	0	16.93	11	0
dbr:Martina_Navratilova	15.68	20	0	13.73	19	0	14.69	20	0
dbr:Political_divisions_of_the_United_States	23.20	8	3	17.93	8	4	20.52	8	7
dbr:Venus_Williams	16.10	19	0	15.20	14	0	15.64	16	0
dbr:Communes_of_France	17.37	14	1	15.06	15	0	16.20	14	1
dbr:Democratic_Party_(United_States)	22.37	9	2	19.90	5	1	21.11	7	3
dbr:Forward_(association_football)	16.27	18	0	14.66	16	0	15.45	19	0
dbr:Republican_Party_(United_States)	21.93	10	2	19.16	6	2	20.52	9	4

Table 2. Average rank of different entities in different countries. The rank of the entities taking into consideration the evaluation of each user per country. The table above shows the number of times that the entity was chosen as the most important entity among the users (#Top-1), the the individual average position (AVG) and the average rank (#Rank) of each individual entity per country as well as the combined (Combined).

6 Conclusion, Limitations & Future Work

In this paper, we presented a formal framework for evaluating and publishing RDF ranking. Nine different ranking functions were applied to manual generated entity ranks from two different countries (America and India). Moreover, we presented a variant of the Speman's Footrule rank similarity function to measure heterogeneous ranks and demonstrated why the proposed extension is more accurate than the basic formula. Furthermore, the generated rank profiles are publicly available[7] and can be used for benchmarking other ranking functions. The evaluated results show that the use of ranks from external data sources is more efficient when ranking entities. For future work, we plan to (1) investigate rank similarity functions for heterogeneous ranks, (2) increase the number of rank profiles as well as (3) extend the evaluation to other countries and ranking functions. A limitation of the present framework is regarding dynamic ranks [1,4], which requires additional contextual information such as: (1) what information

the user is trying to find in a query or (2) what are the user's preferences and background? However, we plan to address this issue in future works.

7 Acknowledgements

References

1. M. Alsarem, P.-E. Portier, S. Calabretto, and H. Kosch. Ranking entities in the age of two webs, an application to semantic snippets. In *The Semantic Web. Latest Advances and New Domains*, volume 9088 of *Lecture Notes in Computer Science*, pages 541–555. Springer International Publishing, 2015.
2. S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the Seventh International Conference on World Wide Web 7*, WWW7, pages 107–117, Amsterdam, The Netherlands, The Netherlands, 1998. Elsevier Science Publishers B. V.
3. G. Cheng, T. Tran, and Y. Qu. RELIN: Relatedness and informativeness-based centrality for entity summarization. In *Proceedings of the 10th International Conference on The Semantic Web - Volume Part I*, ISWC'11, pages 114–129, Berlin, Heidelberg, 2011. Springer-Verlag.
4. G. Cheng, D. Xu, and Y. Qu. Summarizing entity descriptions for effective and efficient human-centered entity linking. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 184–194. International World Wide Web Conferences Steering Committee, 2015.
5. S. F. De Araújo and D. Schwabe. Explorator: a tool for exploring rdf data through direct manipulation. In *Linked data on the web WWW2009 workshop (LDOW2009)*, 2009.
6. M. Kendall. *Rank correlation methods*. Griffin, London, 1948.
7. R. Kumar and S. Vassilvitskii. Generalized distances between rankings. In *WWW*, WWW '10, pages 571–580. ACM, 2010.
8. A.-C. N. Ngomo and S. Auer. Limes-a time-efficient approach for large-scale link discovery on the web of data. *integration*, 15:3, 2011.
9. E. Pavlick, M. Post, A. Irvine, D. Kachaev, and C. Callison-Burch. The language demographics of amazon mechanical turk. *Transactions of the Association for Computational Linguistics*, 2, 2014.
10. A. Roa-Valverde and M.-A. Sicilia. A survey of approaches for ranking on the web of data. *Information Retrieval*, 17(4):295–325, 2014.
11. C. Spearman. The proof and measurement of association between two things. *American Journal of Psychology*, 15:88–103, 1904.
12. A. Thalhammer and A. Rettinger. Browsing DBpedia entities with summaries. In *The Semantic Web: ESWC 2014 Satellite Events*, pages 511–515. Springer, 2014.
13. V. N. Vapnik and V. Vapnik. *Statistical learning theory*, volume 1. Wiley New York, 1998.