SELEcTor: Discovering Similar Entities on LinkEd DaTa by Ranking their Features

Lívia Ruback, Marco Antonio Casanova
Departamento de Informática
PUC-Rio
Rio de Janeiro, Brazil
{lrodrigues, casanova}@inf.puc-rio.br

Chiara Renso, Claudio Lucchese
HPCLab
ISTI-CNR
Pisa, Italy
{chiara.renso, claudio.lucchese}@isti.cnr.it

***Abstract*—Several approaches have been used in the last years to compute similarity between entities. In this paper, we present a novel approach to compute similarity between entities using their features available as Linked Data. The key idea of the proposed framework, called SELEcTor, is to exploit ranked lists of features extracted from Linked Data sources as a representation of the entities we want to compare. The similarity between two entities is thus mapped to the problem of comparing two ranked lists. Our experiments, conducted with museum data from DBpedia, demonstrate that SELEcTor achieves better accuracy than state-of-the-art methods.**

Keywords: Linked Data; Entity Similarity; Rank correlation

I. Introduction

The emergence of the Linked Data initiative has promoted in the last years the creation and publication of previously isolated datasets as interlinked, reusable data graphs using known Web standards, and is leading to an unprecedented global space, the Web of Data. This scenario opens up opportunities to exploit the semantic relations between Linked Data resources, or *entities*.

This field, also known as *Semantic Relatedness*, has an enormous potential to be applied to different domains, such as the academic field when comparing universities, researchers or their works, and the tourism domain, when comparing points of interests of travelers, such as the museums visited during a trip. In the later domain, the characteristics of the museums (or features) visited by a traveler can help understand his behavior, find behavioral patterns and similarities between travelers, which is especially useful for recommender systems on the tourism domain.

Inspired by this context, we focus in this paper on the problem of discovering similar Linked Data entities using their ranked features, also extracted from Linked Data. We call an *entity feature* any other Linked Data node (subject or object of a RDF triple) that may be somehow related to the entity.

We argue that the similarity between Linked Data entities is better captured if we first rank their features, in a way that the more relevant features appear before the less relevant ones, and then compare the ranked lists.

More specifically, we address the following questions:

(RQ1) Which features are most relevant to compute entity similarity?

A relevant feature characterizes an entity according to some topic of interest. A relevant feature could be, for example, a literary genre of a book published by a writer, a gender of a movie directed by a film director. In this paper, we argue that the art movements of a museum's artworks are high quality features.

(RQ2) How to measure the similarity between entities based on their ranked features?

Given two entities, we aim at discovering how similar these entities are by taking into account their features, which were previously ranked according to some relevance criteria. Therefore, the problem of measuring the similarity between two entities is mapped to the problem of comparing their ranked features.

Briefly, the contributions of this paper are:

1. We formulate the problem of *discovering similar entities on Linked Data by ranking and comparing their features.* To the best of our knowledge, this is the first work that maps this problem to the problem of comparing two ranked lists, also extracted from Linked Data.
2. We propose SELEcTor, a two-module framework that takes as input Linked Data entities, ranks the lists of entity features according to their relevance for describing the entities, compares the ranked lists using rank correlation metrics, and outputs the entities similarity.
3. We validated our framework by performing an experiment with Linked Data entities, extracted from DBpedia, that represent some of the most famous museums. We exploited a ground truth dataset based on a curated corpus of documents, from which we estimated museum similarity. We found that the art movements of the museums' artworks are high quality features and we can anticipate here that we achieved better results than a chosen baseline.

The remainder of this paper is structured as follows. Section II introduces basic definitions and concepts. Section III presents the SELEcTor framework and describes its modules. Section IV shows the experiments we performed in the museums domain. Section V summarizes related work. Finally, Section VI reports the conclusions and future work.

II. PRELIMINARIES

Basic Definitions

The problem of discovering similar entities on Linked Data by ranking and comparing their features depends on two basic definitions. Definition 1 is related to *RQ1* and Definition 2 to *RQ2*.

DEFINITION 1 (RANKED FEATURES). The *ranked features* of a Linked Data entity e is a list $F = ((f_1,s_1),\ldots,(f_n,s_n))$, where f_j is a feature of e and s_j is the *relevance score* of f_j, for $j \in [1, n]$.

Intuitively, feature f_j is more relevant to describe e than feature f_{j+1} , for $1 \leq j \leq n$.

DEFINITION 2 (ENTITY SIMILARITY). Given two Linked Data entities e_i and e_j and their ranked features F_i and F_j, the *similarity* between e_i and e_j is the distance between F_i and F_j, according to a rank correlation metric m.

Note that entities e_i and e_j may have a different number of relevant features, i.e., their lists F_i and F_j may have different sizes. Furthermore, F_i and F_j may or may not have features in common.

Semantic Relatedness of Linked Data Entities

In the literature, there are some measures to compute the similarity between Linked Data entities [1][10]. The most relevant to our work is the measure proposed in 2003 by Milne and Witten, the *Wikipedia Link-based Measure* (WLM) [11]. Formally, the relatedness between two Wikipedia articles of interest a and b is:

$$sr(a,b) = \frac{\log(\max(|A|,|B|)) - \log(|A \cap B|)}{\log(|W|) - \log(\min(|A|,|B|))}$$

where A and B are the sets of all articles that link to a and b, respectively, and W is the entire Wikipedia.

Another important measure is the *Semantic Connectivity Score* (SCS), proposed by Nunes et al. [6]. The SCS between a pair of entities a and b is based on the Katz score [10] and is defined as follows:

$$SCS\,(a,b) = \sum_{l=1}^{\tau} \beta^{l} \cdot |paths_{(a,b)}^{<l>}|$$

where $|paths_{(a,b)}^{<l>}|$ is the number of paths between entities a and b of length l, τ is the maximum path length considered and β is a positive damping factor ranging from 0 to 1, responsible for exponentially penalizing longer paths.

Rank correlation metrics

Correlating ranked lists is a common problem in several areas, such as graph analysis and information retrieval. Webber et al. [11] categorize them according to two main characteristics: the *conjointness* (two conjoint lists consist of the same items) and the *weightedness* (a list is *weighted* when the items have different relevancies and a list is *top-weighted* when the top of the list is more important than the tail).

For conjoint lists, some widely used rank correlation coefficients are Kendall's and Spearman's [7]. They assume that the lists have the same items. For non-conjoint lists with items of different weights (ranks), there are some similarity measures that can be used, such as *Weighted Jaccard* [18], *Cosine Similarity*, and *Average Overlap* [2][3], among others.

The *Average Overlap* (AP) is based on set intersection, but considers the overlap at increasing depths when comparing two rankings, and is defined as follows [11]:

$$AO(S,T,k) = \frac{1}{k}\sum_{d=1}^{k} A_d$$

where S and T are two possibly infinite lists, A is their agreement at depth d (i.e., the intersection of lists S and T up to depth d) and k is the evaluation depth.

Webber et al. [11] proposed to extend this idea to incomplete ranks (i.e. they do not cover all elements in the domain). They defined the *Rank-biased overlap* (RBO) as follows.

$$\mathrm{RBO}(S,T,p) = (1-p)\sum_{d=1}^{\infty} p^{d-1} \cdot A_d$$

The RBO measure handles non-conjoint lists and weights high ranks more heavily than low ranks (their top-k item is more relevant then the top-k+1, and so on). In addition, RBO has a parameter p, which ranges from 0 to 1, and determines the strength of the weighting to top ranks, i.e., higher p values imply stronger emphasis on top ranked elements.

III. THE SELEcTOR FRAMEWORK

Figure 1 gives an overview of the SELEcTor framework. The SELEcTOR framework takes as input Linked Data entities, extracts their relevant features from datasets found in the Linked Open Data cloud, and then compares the ranked features according to some rank correlation metric to generate the entities similarity score. The first module is the *ranked features extractor*, which communicates with Linked Open datasets to extract the ranked list of relevant features that describe the entities. The second module, the *entity similarity processor*, takes these lists as input and compares them using a list correlation metric to generate as output the entity similarity.

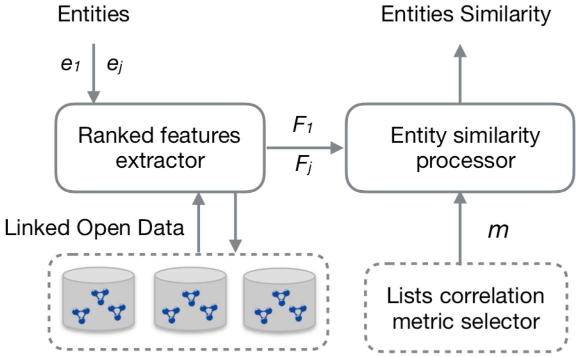
Entities
e1
ej
Entities Similarity
Ranked features extractor
F1
Fj
Entity similarity processor
Linked Open Data
m
Lists correlation metric selector

Figure 1. Overview of SELEcTor framework.

We detail both modules in what follows.

A. Extracting ranked features

The *ranked features extractor* module generates ranked lists of features that describe entities. As it can be seen in Figure 1, this module receives as input two Linked Data entities and outputs their respective ranked features. Taking an entity as input, the module navigates on the Linked Data graphs through the nodes connected to the entity to extract the features. The module accesses the Linked Data graphs through their respective SPARQL endpoints. It is important to notice that feature identification is part of an analysis process that can be aided by a domain expert.

Consider the museums scenario. We claim that two museums can be compared based on the art movements of their artworks. We also investigated other Linked Data features, such as those related to the museums' popularity based on the number of visitors, but we found that, compared with other features available as Linked Data, the art movements better describe the museums. Therefore, we claim that museums with artworks of similar art movements are similar, and museums with no art movements in common would be completely different.

The approaches to order the features generating a ranked list can be *query-based* or *graph-exploration*.

The *query-based approach* performs a pre-defined SPARQL query over one or more Linked dataset endpoint to generate the ranked features. In the case of museums, the SPARQL query would match a certain path pattern to get all the art movements (features) that describe the museum (the input entity), according to some relevance criterion. The relevance criterion is applied using some group function that aggregates the features, for instance, by counting the number of artworks of each art movement. In the next section, we present an example of a SPARQL query for the museums domain.

We note that, depending on the aggregation function chosen, a tie may happen between two or more features, in the sense that they have the same number of feature values. In these cases, the SPARQL query can be re-formulated to untie the elements according to some other criteria.

When applying the *graph-exploration* approach, the module navigates through the RDF graph and then calculates the importance of each node to describe a certain entity. There are several approaches in the literature to measure relationships within a graph, commonly referred as *centrality measures*, such as the Katz score [6] or the SCS score [10]. They can be profitably applied in this context to generate a ranked list of features (nodes) that represents a certain entity.

In both approaches (*query-based* and *graph-exploration*) the module performs SPARQL queries over the Linked datasets. The main difference is that in the query-based approach, the query already retrieves the ranked features, i.e., it orders the features and the answer of the query is the ranked list of features itself, while in the graph-exploration approach, the SPARQL query is used to retrieve an RDF graph which the module will use to measure the importance of each entity.

B. Computing entity similarity

The *entity similarity processor* module takes as input the lists of features, and compares them to measure how similar they are, using a rank correlation metric. The module, therefore, outputs the similarity score for the pair of entities.

The module can choose one of the similarity measures introduced in Section II to compute the similarity between the entities, according to the nature of the ranked features. If the lists have the same items (i.e., if they are conjoint), the module may choose Kendall, Spearman's p, among other metrics [7]. Otherwise, the module may choose AO (Average Overlap) [2] or its top-weighted parameterized extension, RBO [11].

The module outputs a similarity score that measures how similar the entities are. When the entities do not have any feature in common, the correlation coefficient is 0, and when they have the same features and in the same order (and the same weights for weighted ranks), the output is 1.

IV. Experiments

In this section, we instantiate the framework and evaluate our approach comparing the similarity measure obtained using SELEcTor with a ground truth. We adopt the museums domain as an example.

A. Museums on DBpedia

DBpedia provides entities that represent museums around the world, which are instances of `dbo:Museum` class. An example is the triple `<dbr:Louvre, rdf:type, dbo:Museum>` (`dbr` is the prefix of `http://dbpedia.org/resource/`*)* stating that the Louvre is an entity of type museum.

The museum instances can be linked to other entities through the `dct:subject` property, often used to represent the topic of the entity. Some of these entities are hierarchically related to each other through the `skos:broader` property and may have a direct link to the `dbc:Museum_by_type` class. We call *categories* all the entities linked to `dbc:Museum_by_type` directly or indirectly through the `skos:broader` property.

Figure 2 shows some categories associated with the Louvre. A category directly related to Louvre is `dbc:Museums_of_Ancient_Greece`. The indirectly related categories are `dbc:History_museum` and `dbc:Civilization_museums.`

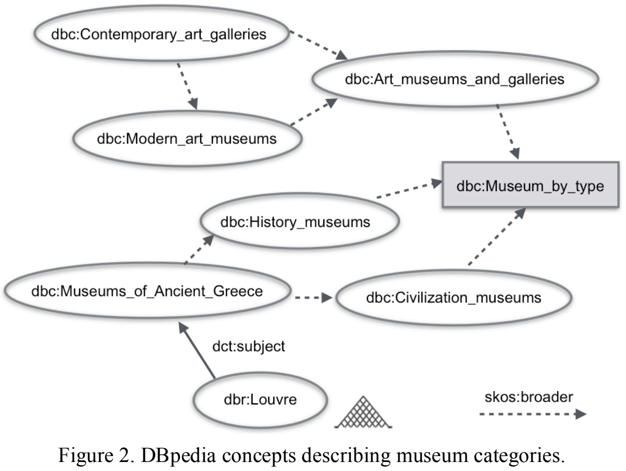

Figure 2. DBpedia concepts describing museum categories.

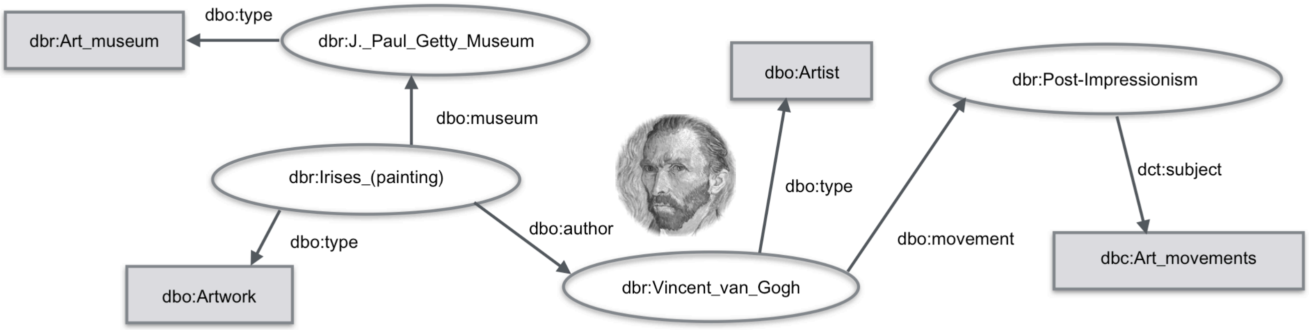

Figure 3. DBpedia links describing J. Paul Getty museum features.

Figure 3 illustrates other museum properties to be explored in DBpedia. The `dbo:museum` property links a museum to its artworks, instances of `dbo:Artwork`. In turn, each artwork may be linked to its creator/artist through the `dbo:author` property. Finally, the artists may be related to one or more art movements by the `dbo:movement` property. The RDF graph shown in Figure 3 represents that the J Paul Getty Museum has as artwork the *Irises* painting by Vincent van Gogh, an impressionist (art movement) artist.

In the experiments, we explored both these DBpedia graphs (shown in Figure 2 and Figure 3), as follows.

B. *Ranking the features*

Following the first step of the framework, the extraction of ranked features, we applied to the DBpedia graph both the *graph-exploration* and the *query-based* approaches, described in Section III (a).

When applying the *graph-exploration* approach, we explored DBpedia entities and properties as shown in Figure 2. Given a Linked Data entity *e* that represents a museum, the *ranked features extractor* module queries DBpedia via its SPARQL endpoint following only the properties `dct:subject` and `dct:broader`. We navigate the graph from the root entity (the museum) reaching the entities that represent their categories.

We consider such museum categories (the entities having direct or indirect links to the `dbc:Museum_by_type` class) as the features to be ranked by this module. We used the depth-first approach with depth distance 4, as adopted in [19], which means that we considered entities from the root until all its 4-hop neighbors.

To measure the relevance of each feature with respect to the museum, we computed the distance from the museum to all the features (the categories) using the SCS score (see Section II) [10]. We then ordered the features according to the score, generating the ranked features list.

Table I shows Louvre's ranked features using the graph-exploration approach, ordered in descending order by SCS score. The feature `dbc:Museums_of_Ancient_Near_East`, at the first position, and the feature `dbc:Museums_of_Ancient_Greece`, at the second position, are the more relevant to describe the Louvre (both with SCS score 0.5), while the less relevant is `dbc:Civilization_museums`, with SCS score 0.25.

TABLE I. LOUVRE FEATURES BY THE GRAPH-EXPLORATION APPROACH

Louvre categories	SCS score
dbc:Museums_of_Ancient_Near_East	0.5
dbc:Museums_of_Ancient_Greece	0.5
dbc:History_museums	0.48
dbc:Civilization_museums	0.25

When applying the graph-exploration approach to a group of museums, we found that the ranked features did not represent the museums suitably for two reasons: (i) some of the most famous museum have few DBpedia categories that represent them, such as the Louvre; and (ii) in some cases the categories found are very generic and thus do not represent the museums appropriately (such as `dbc:Society_museums`, `dbc:Art_museums_and_Galleries` and `dbc:Civilization_museums`).

We therefore also applied the *query-based* approach, exploiting the DBpedia graph paths shown in Figure 3. Instead of exploiting the museum categories, we focused on the art movements of the artworks that can be reached using the artists, shown in Figure 3.

The *ranked features extractor* module performed the following SPARQL query, which aggregates the art movements and orders them by the artwork frequency. The `?museum` parameter represents the input entity.

```
SELECT ?artMovement
WHERE {
  ?artWork <dbo:museum> ?museum.
  ?artWork <dbo:author> ?artist.
  ?artist <dbo:movement> ?artMovement.
}
GROUP BY ?art_movement
ORDER BY DESC(count(?artWork))
```

Table II shows the SPARQL query results for the Getty Museum. Note that, since DBpedia is constantly updated, the results may vary. In some cases, the SPARQL query may return items with the same feature value, i.e., two or more art movements with the same artwork frequency. In this case, one may choose another criterion to untie these features, such as the *out* or the *in* degree of the feature, which respectively represent the number of RDF out-links leaving from the entity and the number of in-links pointing to the entity.

TABLE II. GETTY FEATURES EXTRACTION BY THE QUERY-BASED APPROACH.

J Paul Getty ranked features
dbr:Symbolism_(arts)
dbr:Baroque
dbr:Expressionism
dbr:Romanticism
dbr:High_Renaissance
dbr:Dutch_Golden_Age_painting
dbr:Academic_art
dbr:Post-Impressionism
dbr:Mannerism

According to the query-based approach and by using the SPARQL query shown above, the feature that best describes the Getty Museum is the *Symbolism* art movement while the less relevant feature is the *Mannerism* art movement, which means that the museum has more artworks of the Symbolist art period than of the Romanticism art period.

Comparing the graph-exploration approach with the query-based approach for the museums scenario, we found that the later strategy can extract better features, both in quantity (the ranked lists have more items) and in quality (the art movements are more domain-specific than the museum categories available on DBpedia).

C. Computing entity similarity

As explained in Section III, the *entity similarity processor* module takes as input the ranked features, representing the entities to compare, and outputs a similarity score.

Table III shows the ranked features that describe the Getty and the Louvre museums (for a matter of space, some Louvre features have been omitted). They have been generated in the previous step using the query-based approach by performing the SPARQL query previously presented.

To compare the two ranked features, the *entity similarity processor* module choses the RBO measure (see Section II) [11], since it handles non-conjoint lists (the museums are not described by the same art movements) and weights high ranks more heavily than low ranks (their top-k art movement is more relevant then the top-k+1, and so on).

When computing the similarity between the Getty and the Louvre museums, with the top-weighted parameter *p* equal to 0.95, the RBO score is 0.437. In fact, from a total of 4 common features (the art movements), the first art movement that occurs in the Getty and the Louvre lists (`dbr:Baroque`) appears in the 4nd position in the Louvre list and the last art movement to match (`dbr:Dutch_Golden_Age_Paiting`) appears in the 6th position in the Louvre list.

When comparing the Getty Museum with the Museum Of Modern Art, in New York, the RBO similarity score with p = 0.95 is 0.117. In fact, they only have 2 common art movements, the first art movement matches in the 8th position (`dbr:Post-Impressionism`) and the last art movement matches in the 14th position (`dbr:Expressionism`). Considering our museums dataset shown in Table IV, the Museum Of Modern Art is the museum least similar to the Getty Museum.

TABLE III. COMPARING THE RANKED FEATURES.

J Paul Getty ranked features	Louvre ranked features
dbr:Symbolism_(arts)	dbr:Romanticism
dbr:Baroque	dbr:High_Renaissance
dbr:Expressionism	dbr:Neoclassicism
dbr:Romanticism	dbr:Baroque
dbr:High_Renaissance	dbr:Italian_Renaissance
dbr:Dutch_Golden_Age_painting	dbr:Dutch_Golden_Age_painting
dbr:Academic_art	dbr:The_Renaissance
dbr:Post-Impressionism	dbr:Classicism
dbr:Mannerism	dbr:Realism_(arts)
	dbr:Flemish_Baroque_painting
	dbr:Early_Netherlandish_painting
	dbr:Caravaggisti

We computed the similarity between all museums of our dataset (presented in Table V) using RBO. Then, we generated, for each museum, the list of the most similar museums. Table IV shows the most similar museums to the Getty Museum, with *p* = 0.95 and *p* = 0.98.

TABLE IV. THE MOST SIMILAR MUSEUMS TO J. PAUL GETTY.

Getty similars	RBO score p = 0.95	RBO score p = 0.98
dbr:Metropolitan_Museum_of_Art	0.437	0.491
dbr:Louvre	0.404	0.429
dbr:Kunsthistorisches_Museum	0.385	0.419
dbr:Museum_of_Fine_Arts,_Boston	0.360	0.381
dbr:Vatican_Museums	0.351	0.380
dbr:Uffizi	0.261	0.302
dbr:National_Gallery_of_Art	0.247	0.281
dbr:Musée_d'Orsay	0.161	0.195
dbr:Philadelphia_Museum_of_Art	0.161	0.195
dbr:Museum_of_Modern_Art	0.117	0.139
dbr:Art_Institute_of_Chicago	0.103	0.130

D. Evaluation

1) Constructing the ground truth

Since there is no specific ground truth containing museums similarity data to validate our approach, we built it using a well-known Web site about art history. SmartHistory[1] is a non-profit organization that makes art history learning content freely available and provides several articles discussing the most important masterpieces, ranging from ancient to contemporary art.

We chose SmartHistory as the ground truth because it is a rich source of museums data entirely authored by human domain experts, its creation process is totally independent from the DBpedia (or similar) data, and it is not affected by popularity bias.

Each SmartHistory article (very often an article is about an artwork) is categorized according to a hierarchical taxonomy, which includes time periods, art movements, and other relevant facets. For each artwork, the hosting museum is also mentioned.

[1] http://smarthistory.org

Given the SmartHistory data, we defined the similarity between two museums based on the categories found in the articles mentioning the museums. To avoid sparsity, we limited to the top-2 levels of the category hierarchy. Museum similarity was then computed as the cosine similarity of the museum's categories. Cosine similarity was adopted as it allows to properly weight the richness of a given museum in a specific category, but it also avoids boosting large museums with many artworks.

2) Choosing the set of museums

First, we pre-filtered 32 museums in DBpedia with at least 8 artworks and 5 art movements to avoid poorly described museums. Then, we filtered the museums that have also categories in the ground truth, SmartHistory, resulting in the 12 richest museums in both datasets, shown in Table V.

TABLE V. CHOSEN MUSEUMS FOR THE EXPERIMENT

Museum	#DBpedia art movements	#SmartHistory art movements
dbr:Metropolitan_Museum_of_Art	28	84
dbr:Louvre	17	37
dbr:Museum_of_Modern_Art	36	17
dbr:National_Gallery_of_Art	29	17
dbr:J._Paul_Getty_Museum	9	14
dbr:Uffizi	11	12
dbr:Museum_of_Fine_Arts,_Boston	7	16
dbr:Musée_d'Orsay	11	10
dbr:Art_Institute_of_Chicago	26	9
dbr:Philadelphia_Museum_of_Art	15	13
dbr:Kunsthistorisches_Museum	9	11
dbr:Vatican_Museums	16	13

We chose as baseline the semantic relatedness measure proposed by Milne and Witten, WLM [8] (see Section II). Even if WLM is intended to be a generic approach, we chose it as a baseline, since it also measures the semantic relatedness of two Linked Data entities. To compute the WLM similarity, we used Dexter, an Open Source Framework for Entity Linking [1].

Our strategy to evaluate the results is based on the comparison of the lists of similar museums (such as in Table IV) generated by the three different approaches: (i) our approach, namely the SELEcTor framework, (ii) the WLN measure, which represents our baseline and (iii) the SmartHistory data, from which we have built the ground truth.

Then, for each museum, we compared the SELEcTor list with the ground truth list. As an example, Table VI shows the lists of museums similar to the Getty Museum generated by SELEcTor and SmartHistory. As shown in Table VI, according to SELEcToR, the MET (Metropolitan Museum of Art, in New York) is the museum most similar to the Getty Museum, the second most similar is the Louvre Museum and the third one is the Kunsthistorisches Museum, an art museum in Vienna, and so on.

TABLE VI. COMPARING SELECTOR WITH THE GROUND TRUTH

SELEcTor Getty similars	SmartHistory Getty similars
dbr:Metropolitan_Museum_of_Art	dbr:Metropolitan_Museum_of_Art
dbr:Louvre	dbr:Vatican_Museums
dbr:Kunsthistorisches_Museum	dbr:Louvre
dbr:Museum_of_Fine_Arts,Boston	dbr:National_Gallery_of_Art
dbr:Vatican_Museums	dbr:Art_Institute_of_Chicago
dbr:Uffizi	dbr:Museum_of_Fine_Arts,Boston
dbr:National_Gallery_of_Art	dbr:Musée_d'Orsay
dbr:Musée_d'Orsay	dbr:Philadelphia_Museum_of_Art
dbr:Philadelphia_Museum_of_Art	dbr:Kunsthistorisches_Museum
dbr:Museum_of_Modern_Art	dbr:Uffizi
dbr:Art_Institute_of_Chicago	dbr:Museum_of_Modern_Art

We also compared the lists of similar museums generated by WLM (our baseline) and SmartHistory. As an example, Table VII shows of the lists of museums similar to the Getty Museum generated by WLM and SmartHistory.

TABLE VII. COMPARING WLM WITH THE GROUND TRUTH

WLM Getty similars	SmartHistory Getty similars
dbr:National_Gallery_of_Art	dbr:Metropolitan_Museum_of_Art
dbr:Musée_d'Orsay	dbr:Vatican_Museums
dbr:Philadelphia_Museum_of_Art	dbr:Louvre
dbr:Museum_of_Fine_Arts,Boston	dbr:National_Gallery_of_Art
dbr:Kunsthistorisches_Museum	dbr:Art_Institute_of_Chicago
dbr:Art_Institute_of_Chicago	dbr:Museum_of_Fine_Arts,Boston
dbr:Metropolitan_Museum_of_Art	dbr:Musée_d'Orsay
dbr:Uffizi	dbr:Philadelphia_Museum_of_Art
dbr:Museum_of_Modern_Art	dbr:Kunsthistorisches_Museum
dbr:Vatican_Museums	dbr:Uffizi
dbr:Louvre	dbr:Museum_of_Modern_Art

According to the baseline WLM (Table VII), the most similar museum to the Getty Museum is the National Gallery of Art, an art museum in Washington D.C. The second most similar museum is the Musée d'Orsey, in Paris, and the third most similar is the Philadelphia Museum of Art, and so on.

Analyzing the results, we note that the geographic proximity between two museums influences the similarity score between them, according to WLM, as expected. This is because this measure considers all links found in their respective Wikipedia articles, including some geographic-related links. An example is the link <dbc:Modern_art_museums_in_the_United_States>, connected to the museum through the property <dct:subject>, that can be found both in the Metropolitan Museum page and in the National Gallery of Art page (dbc is a prefix for http://dbpedia.org/page/Category and dct is a prefix for http://purl.org/dc/ terms/). This explains why, according to WLM, some museums – for instance the Louvre and the Vatican – are in the last positions in the similarity list, while in the SELEcTor lists of similar museums, they appear in the first-half of the list. Analogously, WLM considered as similar some museums that SELEcTor does not – for instance, the

Philadelphia Museum of Art, which is also located in the United States.

Lastly, we calculated the accuracy of the SELEcToR lists and the WLN lists by comparing both with the ground truth. We compared the lists using NDCG (Normalized Discounted Cumulative Gain) [5], a well-known metric used in Information Retrieval to measure ranking quality. This measure accumulates the gain from the top of the list to the bottom, penalizing lower ranks. It may be parameterized to consider the top k elements of the lists.

Figure 4 shows the results considering only the Getty Museum, with *k* from 3 to 8. Considering the top 3 items, the SELEcTor accuracy score is 0.886, while the WLM score is 0.697. Considering the top 4 items, the SELEcTor score decreases to 0.876, but it is still higher than the WLM score, 0.714. The higher SELEcTor accuracy was 0.924, when *k* = 8.

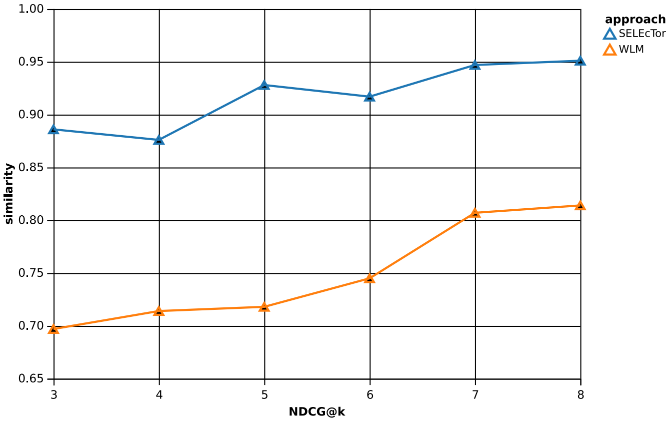

Figure 4. NDGC results for Getty museum

Finally, we compared the similarity lists of all museums using the same idea, again with *k* ranging from 3 to 8. Figure 5 shows the results.

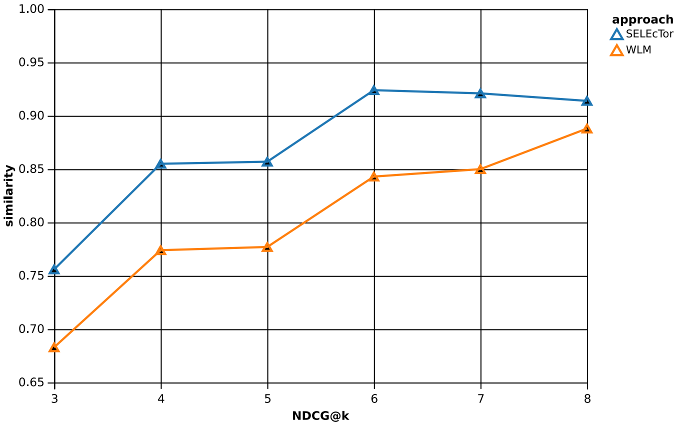

Figure 5. The average NDGC top k items

Considering all museums, SELEcTor performs significantly better than WLM, for any *k*, with the highest accuracy being 0.924, when *k* = 6.

Since SELEcTor filters the entities that better describe a museum, it can be considered more selective than WLM, in the sense that it focuses on the specific features that describe the museum, such as its art movements.

We stress that we have chosen the museum scenario for the experiment, but our approach can be tuned or generalized to other domains. Another interesting scenario is the similarity computation between university departments or institutes. When comparing two computer science departments (entities), say, one could profitably use the approach we presented in this work: first generate ranked lists of published works (features) in each conference/journal, and then compare the ranked lists to compute the similarity between the departments. By analogy with the museums scenario, the department represents the museum, the published works – journal or conference papers – represent the museum artworks and the work authors – professors, researches and students – represent the artists.

V. Related work

We divide related work in three groups, as follows.

Similarity measures for rankings approaches investigate measures of similarity between ranked lists of items. When the top ranks are considered more important than the lower ranks the measures are often refereed as *Weighted Rank Correlation* measures.

Fagin et al. [2][3] proposed the intersection metric to deal with non-conjoint lists. The authors extended the set of intersection idea, considering the cumulative overlap at increasing depths.

The intersection metric introduced by Fagin is referred as the *average overlap* (AO) in Webber et al. [11], that extended Fagin's idea to handle incomplete rankings when proposing, in 2010, a new measure named *rank-biased overlap* (RBO). RBO has a parameter that determines the strength of the weighting to top ranks. We used the RBO measure [11] in this work as the rank correlation metric to compare the ranked features that represents the entities we compared in the experiment.

Ranking Linked Data approaches rank resources exploiting the semantic aspects of Linked Data datasets.

ReConRank [13] proposed a ranking method that adapts the well-known PageRank HITS algorithms to Semantic Web data. Mirizzi et al. [14] exploited semantic tagging on Linked Open Data to rank resources. Their methodology combined the graph-based nature of the RDF structure, semantic relation in the graph and search engine results to rank Linked Data resources.

Mirizzi et al. [14] is the most relevant work to our approach. The main difference is that they do not exploit rank correlation metrics to rank resources; instead, they take advantage of results coming from search engines.

Semantic Relatedness of Linked Data approaches focus on measuring the similarity between Linked Data entities. The works on the literature can be focused into social network theory approaches, entity disambiguation [4], ontology-based approaches [12][15], Wikipedia structure-based approaches [8], among others. They can also combine these approaches to create a hybrid approach to take advantage of several strategies.

Milne et al. [8] proposed in 2008 an approach to measure semantic relatedness between entities using the hyperlink structure of Wikipedia (WLM). Nunes et al. [10] proposed a combined approach to compare disparate entities using graph analysis of datasets and entity co-occurrence on the Web. Ceccarelli et al. [1] proposed a machine-learned methodology to measure entity relatedness using reference datasets with annotated data. We used the WLM measure [11] in this work as the baseline for the experiment. As we discussed before, we chose WLM even if it is intended to be a generic approach, since it a well-known semantic relatedness measure.

Leal et al. [16] instead proposed a tool that extracts ontology for a given domain from DBpedia and use it to compute the semantic relatedness of terms. Grieser et al. [15] combined the Wikipedia category hierarchy with lexical similarity measures to estimate museum exhibit relatedness. Comparing to our work, although they also exploit the semantic relatedness in the cultural heritage domain, they take advantage of taxonomic and document-based methods to estimate museum exhibit relatedness. We exploit, instead, rank correlation metrics from the Information Retrieval field.

VI. Conclusions and Future Work

In this paper, we presented a novel approach to compute the similarity between Linked Data entities using ranked lists of features, also extracted from Linked Data sources. The ranked lists therefore act as representations of the entities we want to compare.

We then mapped the problem of estimating the similarity between two entities to the problem of comparing two ranked lists. We argued that, by ranking the features based on their relevance, instead of a more naïve approach that would consider the presence or absence of such features, we improved the accuracy of the similarity computation. Our experiments, using museums extracted from DBpedia, showed that SELEcTor achieves better accuracy than a chosen baseline.

Although presented as a self-contained approach to compare Linked Data entities, this work is part of an application-oriented-process in the trajectory domain, previously proposed in [20]. The experiment was oriented to the arts and culture domain, inspired by the fact that museums are a common stop of tourist routes.

We want, therefore, to compare trajectories as a whole, focusing on the semantic facet, and not only on the geo-spatial aspect. By comparing trajectory stops – trajectory parts, such as museums – we may then compare the tourist trajectories, which is especially useful for recommender systems on the tourism domain.

Acknowledgments

This work was partially supported by CNPq under grants 557128/2009-9, 303332/2013-1, 442338/2014-7, 248743/2013-9, 201690/2015-2, by FAPERJ under grants E-26/170028-2008 and E-26/201.337/2014.

References

[1] Diego Ceccarelli, Claudio Lucchese, Salvatore Orlando, Raffaele Perego, and Salvatore Trani. Learning relatedness measures for entity linking. In Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, CIKM '13, pages 139-148, New York, NY, USA, 2013. ACM

[2] Ronald Fagin, Ravi Kumar, Mohammad Mahdian, D. Sivakumar, and Erik Vee. Comparing and aggregating rankings with ties. In Proceedings of the Twenty-third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS'04, pages 47-58, New York, NY, USA, 2004. ACM.

[3] Ronald Fagin, Ravi Kumar, and D Sivakumar. Comparing top k lists. SIAM Journal on Discrete Mathematics, 17(1):134-160, 2003.

[4] Ioana Hulpus, Narumol Prangnawarat, and ConorHayes. Path-Based Semantic Relatedness on Linked Data and Its Use to Word and Entity Disambiguation, pages 442-457. Springer International Publishing, Cham, 2015.

[5] Kalervo Jarvelin and Jaana Kekalainen. Cumulated gain-based evaluation of IR techniques.ACM Transactions on Information Systems (TOIS),20(4):422-446, 2002.

[6] Leo Katz. A new status index derived from sociometricanalysis. Psychometrika, 18(1):39-43, 1953.

[7] Maurice George Kendall. Rank correlation methods.1948.

[8] David Milne and Ian H. Witten. An efective, low-cost measure of semantic relatedness obtained from wikipedia links. In Proceedings of AAAI 2008, 2008.

[9] Kazuya Okamoto, Wei Chen, and Xiang-Yang Li. Ranking of closeness centrality for large-scale social networks. In International Workshop on Frontiers in Algorithmics, pages 186-195. Springer, 2008.

[10] Bernardo Pereira Nunes, Stefan Dietze, Marco Antonio Casanova, Ricardo Kawase, Besnik Fetahu, and Wolfgang Nejdl. Combining a Co-occurrence-Based and a Semantic Measure for Entity Linking, pages 548-562. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[11] William Webber, Alistair Moffat, and Justin Zobel. A similarity measure for indefinite rankings. ACMTrans. Inf. Syst., 28(4):20:1-20:38, November 2010 G.

[12] Abdolreza Hajmoosaei and Petra Skoric. Museum ontology-based metadata. In Tenth IEEE International Conference on Semantic Computing, ICSC 2016, Laguna Hills, CA, USA, February 4-6,2016, pages 100–103, 2016.

[13] Aidan Hogan, Stefan Decker, and Andreas Harth. Reconrank: A scalable ranking method for semantic web data with context. 2006.

[14] Roberto Mirizzi, Azzurra Ragone, Tommaso Di Noia, and Eugenio Di Sciascio. Ranking the Linked Data: The Case of DBpedia, pages 337–354. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[15] Karl Grieser, Timothy Baldwin, Fabian Bohnert, andLiz Sonenberg. Using ontological and document similarity to estimate museum exhibit relatedness. Journal on Computing and Cultural Heritage (JOCCH), 3(3):10, 2011.

[16] José Paulo Leal, Vânia Rodrigues, and Ricardo Queirós. Computing semantic relatedness using dbpedia. In OASIcs – Open Access Series in Informatics, volume 21. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2012.

[17] Maurice George Kendall. Rank correlation methods.1948.

[18] Sergey Ioffe. Improved consistent sampling, weighted minhash and l1 sketching. In 2010 IEEE International Conference on Data Mining, pages 246–255. IEEE, 2010.

[19] Lujun Fang, Anish Das Sarma, Cong Yu, and Philip Bohannon. Rex: explaining relationships between entity pairs. Proceedings of VLDB, 5(3):241–252, 2011.

[20] Livia Ruback, Marco Antonio Casanova, Alessandra Raffaetà, Chiara Renso, and Vania Vidal. Enriching mobility data with linked open data. In Proceedings of the 20th International Database Engineering & Applications Symposium, IDEAS '16, pages 173–182, New York, NY, USA, 2016. ACM.