

# Ranking Complex Relationships on the Semantic Web

Boanerges Aleman-Meza, Christian Halaschek-Wiener, I. Budak Arpinar,  
Cartic Ramakrishnan, and Amit Sheth

Large Scale Distributed Information Systems (LSDIS) Lab,  
Computer Science Department, University of Georgia,  
Athens, GA 30602-7404, USA  
boanerg@cs.uga.edu, ch@cs.umd.edu,  
{budak,cartic,amit}@cs.uga.edu

**Abstract.** The focus of contemporary Web information retrieval systems has been to provide efficient support for the querying and retrieval of relevant documents. More recently, information retrieval over semantic metadata extracted from the Web has received an increasing amount of interest in both industry and academia. In particular, discovering complex and meaningful relationships among this metadata is an interesting and challenging research topic. Just as ranking of documents is a critical component of today's search engines, the ranking of complex relationships will be an important component in tomorrow's Semantic Web analytics engines. Building upon our recent work on specifying and discovering complex relationships in RDF data, called Semantic Associations, we present a flexible ranking approach which can be used to identify more interesting and relevant relationships in the Semantic Web. Additionally, we demonstrate our ranking scheme's effectiveness through an empirical evaluation over a real-world dataset.

**Keywords.** H.3.3.d Metadata, H.3.5.f XML/XSL/RDF

## 1 Introduction

The focus of contemporary Web information retrieval systems has been to provide efficient support for the querying and retrieval of documents. There has been significant academic and industrial research in mainstream search engines, such as Google<sup>1</sup>, Vivisimo<sup>2</sup>, Teoma<sup>3</sup>, etc. These systems have made considerable progress in the ability to locate relevant pieces of data among the vast numbers of documents on the Web.

Currently, due to the increasing move from data to knowledge and the rising popularity of the Semantic Web vision, there is significant interest and ongoing research in automatically extracting and representing semantic metadata as annotations to both documents and services on the Web. Several communities such as the Gene Ontology Consortium, Federal Aviation Administration (Aviation

---

<sup>1</sup> <http://www.google.com>

<sup>2</sup> <http://www.vivisimo.com>

<sup>3</sup> <http://www.teoma.com>

Ontology), Molecular Biology Ontology Working Group, Stanford University's Knowledge Systems Lab, etc. are also effectively conceptualizing domain knowledge and enabling standards for exchanging, managing and integrating data more efficiently. Additionally, research in the Semantic Web has spawned several commercially viable products through companies such as Semagix<sup>4</sup>, Ontoprise<sup>5</sup>, and Network Inference<sup>6</sup> to name a few.

Due to this ongoing work, large scale repositories of semantic metadata extracted from Web pages have been created and are publicly available. For example, TAP [1] is a fairly broad but not very deep knowledge base annotated in Resource Description Framework (RDF)<sup>7</sup> that contains information pertaining to authors, sports, companies, etc. Additionally, SWETO (Semantic Web Technology Evaluation Ontology [2]) is a comparatively narrower but deep knowledge base annotated in RDF.

Given these developments, the stage is now set for the next generation of technologies, which will facilitate getting actionable knowledge and information from semantic metadata extracted from Web documents, the deep Web and large enterprise repositories. Traditionally, many users analyze information by either browsing the Web, or using search engines to locate Web content based on keywords or phrases. Conventional search engines return a ranked list of documents that are expected to contain information corresponding to the keywords used in the search. The user is left with the task of sifting through these results. These approaches therefore do not directly give the end user actionable knowledge, that is, searching the documents is not a goal yet an intermediate step to discover it. The actionable knowledge is usually directed at decision or progress making in business, science etc., and has to be gleaned by the user from the documents. We aim to provide a different type of analysis based on semantic relationships, in which users are given potentially interesting complex relationships between entities, through a sequence of relationships between the metadata (annotations) of Web sources (or documents). We have defined these complex relationships between two entities as Semantic Associations [3]. Arguably, these relationships are at the heart of semantics, lending meaning to information, making it understandable and actionable and providing new and possibly unexpected insights.

When querying for Semantic Associations, users are frequently overwhelmed with too many results. For example, a typical Semantic Association query involving two '*Computer Science Researchers*' over the SWETO test-bed, results in tens, hundreds or thousands of associations varying from co-authorship through their publications, to relationships through the geographic locations they live in. A user cannot be expected to sift through this large number of results in search of those that are highly relevant to his/her interest.

This article is an extension of initial efforts on ranking Semantic Associations [4]. Specifically, we propose a flexible ranking approach with new criteria that allows the identification of the most interesting Semantic Associations between two entities. Additionally, we provide details of the current system implementation and

---

<sup>4</sup> Semagix Inc., <http://www.semagix.com>

<sup>5</sup> Ontoprise GmbH, <http://www.ontoprise.com>

<sup>6</sup> Network Inference Ltd., <http://www.networkinference.com>

<sup>7</sup> <http://www.w3.org/RDF/>

demonstrate the effectiveness of the ranking approach through an evaluation over the SWETO test-bed.

## 2 Background

**Metadata Extraction Techniques.** Ontology driven metadata extraction techniques have been an active research area over the past years. Both semi-automatic and automatic techniques and tools have been developed and significant work continues [5]. Various tools exist, including CREAM [6], Semagix Freedom<sup>4</sup>, SemTag [7], etc. Semagix Freedom has typically been used to populate ontologies that average more than one million instances. SemTag, part of IBM's WebFountain project, has used a smaller ontology but has demonstrated Web scale metadata extraction from well over a billion pages. In particular, the Freedom toolkit has been used as the infrastructure technology to create the data set for our evaluations. Essentially, metadata extractors use regular expressions to extract entities from data sources. As the sources are 'scraped' and analyzed by the extractors, the extracted entities are disambiguated and stored in appropriate classes in an ontology.

**Data Model Used to Represent Metadata.** RDF is a W3C standard used for describing resources using a model based on named relationships between resources. Relationships in RDF, known as *Properties*, are binary relationships between resources (or between a resource and a literal) which take on the roles of *Subject* and *Object* respectively. The *Subject*, *Predicate* and *Object* compose an RDF statement. This model can be represented as a directed labeled graph with typed edges and nodes where a labeled edge connects the *Subject* to the *Object*. Let a *property sequence* be a finite sequence of relationships, that is, a path in the directed graph. A property sequence is therefore a sequence of links between two entities.

**Semantic Associations.** Semantic Associations are complex relationships between two entities. A query for Semantic Associations takes as input two entities,  $e_1$  and  $e_n$ . Adapted from the formal definition [3], we define Semantic Associations as follows: Two entities  $e_1$  and  $e_n$  are semantically associated if there exists one or more property sequence  $e_1, p_1, e_2, p_2, e_3, \dots, e_{n-1}, p_{n-1}, e_n$  in an RDF graph where  $e_i, 1 \leq i \leq n$ , are entities and each  $p_j, 1 \leq j < n$ , is a relationship (property) between entities  $e_j$  and  $e_{j+1}$ . Note that Semantic Associations are complex relationships spanning over heterogeneous schemas (consequently heterogeneous properties and entities). For example, in national security applications, Semantic Associations may enable analysts to see the interconnections between different people, places and events.

Discovery of Semantic Associations between two entities can be viewed as a graph traversal problem over the graph data model of RDF. We implemented and tested various algorithms based on k-hops, random walks and iterative deepening. A discussion of these algorithms is out of the scope of this paper.

### 3 Ranking Semantic Associations

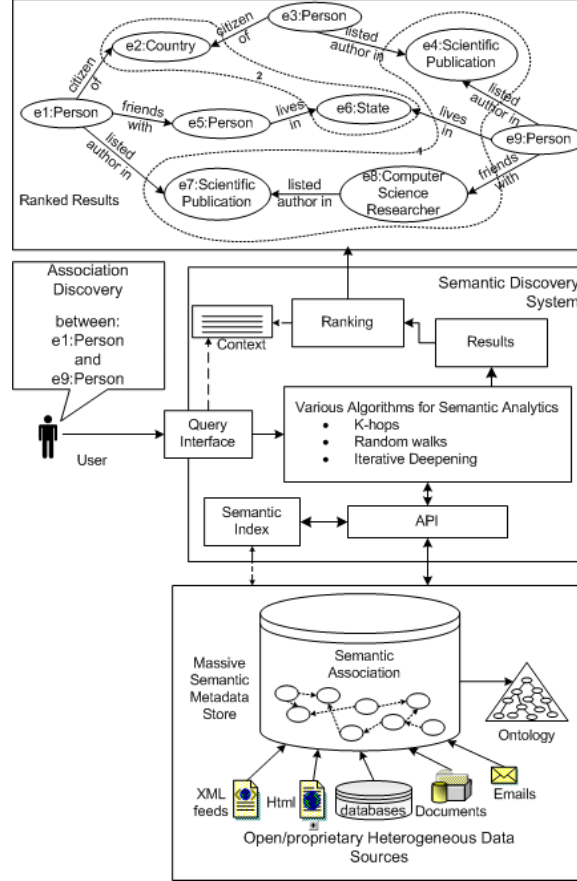
Our goal to rank results of a query involves two entities (e.g., `e1:Person` and `e9:Person` in Fig. 1). Due to the small world phenomenon it is conceivable that there are a myriad of paths connecting two entities. Ranking these paths in order of relevance is required. We identify flexible criteria to rank Semantic Associations. We classify the ranking criteria into *Semantic* and *Statistical* metrics. Semantic metrics are based on semantic aspects of an ontology. Statistical metrics are based on statistical aspects of the ontology, particularly on number and connectivity aspects of entities and relationships.

Traditional keyword based search engines use either the content of resources (words in a Web page) or the link structure between pages to return a ranked set of resources in response to a query. TF-IDF could also be used to judge the relevance of a document with respect to a query term. Our ranking problem however does not aim to rank documents, yet Semantic Associations, which are essentially sequences of properties linking entities. Therefore, the rank of a specific Semantic Association is determined using each property in the property sequence which corresponds to a single relationship between entities. Hence we believe that conventional ranking mechanisms do not apply to the problem we are faced with.

#### 3.1 Semantic Metrics

**Context.** Consider a scenario in which a user is interested in discovering how two ‘Persons’ are related to each other in the domain of ‘Computer Science Publications’. Concepts such as ‘Scientific Publication’, ‘Computer Science Professor’, etc. would be most relevant, whereas concepts such as ‘Financial Organization’ would not. Thus, to capture the relevance of a (complex) relationship, the notion of a *query context* captures various ontological *regions* specified by the user. Since the types of the entities are described using RDF, we use the class and relationship types to restrict our attention to the entities and relationships of interest (query context). The user interacts with a graphical visualization of the ontology to specify the query context (see Fig. 2). A user interested in different domains can manually assign weights to each *region* of the query context so that specific regions of the context can be given more preference than others.

To illustrate our approach, consider three associations depicted at the top of Fig. 1, where a user has specified a contextual *region 1* containing classes ‘Scientific Publication’ and ‘Computer Science Researcher’. Additionally, assume the user specified *region 2* containing classes ‘Country’ and ‘State’. The resulting *regions*, 1 and 2, refer to the computer science research and geographic domains, respectively. For the associations at the top of Fig. 1, (with say, weights 0.8 and 0.2 for regions 1 and 2, respectively), the bottom-most association would have the highest rank because all of its entities and relationships are in the region with highest weight. The second ranked association would be the association at the top of the figure because it has an entity in *region 1*, but (unlike the association in the middle) also has an entity in *region 2*.



**Fig. 1.** System architecture and context example

Before formally presenting the ranking criteria, we introduce notation used throughout the paper. Let  $A$  represent a Semantic Association, that is, a path sequence consisting of nodes (entities) and edges (relationships) that connects the two entities. Let  $length(A)$  be the number of entities and relationships of  $A$ . Let  $R_i$  represent the *region*  $i$ , that is, the set of classes and relationships that capture a domain of interest. Given that both entities and relationships contribute to ranking, let  $c$  be a *component* of  $A$  (either an entity or a relationship). For example,  $c_1$  and  $c_{length(A)}$  correspond to the entities used in a query where  $A$  is one of the Semantic Associations results of the query. We define the following sets for convenience, using the notation  $c \in R_i$  to represent whether the type (rdf:type) of  $c$  belongs to *region*  $R_i$ :

$$X_i = \{c \mid c \in R_i \wedge c \in A\} \quad (1), \quad Z = \{c \mid (\forall i \mid 1 \leq i \leq n) c \notin R_i \wedge c \in A\} \quad (2)$$

where  $n$  is the number of *regions* in the query context. Thus,  $X_i$  is the set of components of  $A$  in the  $i^{th}$  *region* and  $Z$  is the set of components of  $A$  not in any

contextual region. We now define the *Context* weight of a given association  $A$ ,  $C_A$ , such that

$$C_A = \frac{1}{\text{length}(A)} \left( \sum_{i=1}^n (W_{R_i} \times |X_i|) \right) \times \left( 1 - \frac{|Z|}{\text{length}(A)} \right), \quad (3)$$

where  $n$  is the number of *regions*,  $W_{R_i}$  is the weight for the  $i^{\text{th}}$  region.

**Subsumption.** Classes in an ontology that are lower in the hierarchy can be considered to be more specialized instances of those further up in the hierarchy. That is, they convey more detailed information and have more specific meaning. For example, an entity of type “*Professor*” conveys more meaning than an entity of type “*Person*”. Hence, the intuition is to assign higher relevance based on *subsumption*. For example, in Fig. 1, entity ‘e8’ will be given higher rank than entity ‘e5’.

We define the *component subsumption weight* ( $csw$ ) of the  $i^{\text{th}}$  component,  $c_i$ , in an association  $A$  such that

$$csw_i = \frac{H_{c_i}}{H_{\text{depth}}}, \quad (4)$$

where  $H_{c_i}$  is the position of component  $c_i$  in hierarchy  $H$  (the topmost class has a value of 1) and  $H_{\text{depth}}$  is the total height of the class/relationships hierarchy of the current branch. We now define the overall *Subsumption* weight of an association  $A$  such that

$$S_A = \prod_{i=1}^{\text{length}(A)} csw_i \quad (5)$$

**Trust.** Various entities and their relationships in a Semantic Association originate from different sources. Some of these sources may be more trusted than others (e.g., Reuters could be regarded as a more trusted source on international news than some other news organization). Thus, trust values need to be assigned to the meta-data extracted depending on its source. For the dataset we used, trust values were empirically assigned. When computing *Trust* weights of a Semantic Association, we follow this intuition: the strength of an association is only as strong as its weakest link. This approach has been commonly used in various security models and scenarios [8]. Let  $t_{c_i}$  represent the assigned trust value (depending on its data source) of a component  $c_i$ . We define the *Trust* weight of an overall association  $A$  as

$$T_A = \min(t_{c_i}). \quad (6)$$

### 3.2 Statistical Metrics

**Rarity.** Given the increasing size of Semantic Web test-beds, many relationships and entities of the same type exist. We believe that in some queries, rarely occurring entities and relationships can be considered more interesting. This is similar to the ideas presented in [9], where infrequently occurring relationships (i.e., rare events) are considered to be more interesting than commonly occurring ones. In some queries however, the opposite may be true. For example, in the context of money laundering, often individuals engage in common case transactions as to avoid detection. In this case, common looking (not rare) transactions are used to launder funds so that the financial movements will go overlooked [10]. Thus the user should determine, depending upon the query, which *Rarity* weight preference s/he has.

We define the *Rarity* rank of an association  $A$ , in terms of the rarity of the *components* within  $A$ . First, let  $K$  represent the knowledge base (all entities and relationships). Now, we define the *component rarity* of the  $i^{\text{th}}$  component,  $c_i$ , in  $A$  as  $rar_i$  such that

$$rar_i = \frac{|M| - |N|}{|M|}, \text{ where} \quad (7)$$

$$M = \{res \mid res \in K\} \text{ (all entities and relationships in } K\text{), and} \quad (8)$$

$$N = \{res_j \mid res_j \in K \wedge typeOf(res_j) = typeOf(c_i)\}, \quad (9)$$

with the restriction that in the case  $res_j$  and  $c_i$  are both of type `rdf:Property`, the subject and object of  $c_i$  and  $res_j$  must have the same `rdf:type`. Thus  $rar_i$  captures the frequency of occurrence of *component*  $c_i$ , with respect to the entire knowledge base. We can now define the overall *Rarity* weight,  $R$ , of an association,  $A$ , as a function of all the *components* in  $A$ , such that

$$R_A = \frac{1}{length(A)} \times \sum_{i=1}^{length(A)} rar_i \text{ (a);} \quad R_A = 1 - \frac{1}{length(A)} \times \sum_{i=1}^{length(A)} rar_i \text{ (b)}, \quad (10)$$

where  $length(A)$  is the number of *components* in  $A$ . If a user wants to favor rare associations, **(10a)** is used; in contrast, if a user wants to favor more common associations **(10b)** is used. Thus,  $R_A$  is essentially the average *Rarity* (or commonality) of all *components* in  $A$ .

**Popularity.** When investigating the entities in an association, it is apparent that some entities have more incoming and outgoing relationships than others. Somewhat similar to Kleinberg's Web page ranking algorithm [11], as well as the PageRank [12] algorithm used by Google, our approach takes into consideration the number incoming and outgoing relationships of entities. In our approach, we view the number of incoming and outgoing edges of an entity as its *Popularity*. In some queries, associations with entities that have a high *Popularity* may be more relevant. These entities can be thought of as *hotspots* in the knowledge base. For example, authors

with many publications would have high popularity. In certain queries, associations that pass through these *hotspots* could be considered very relevant. Yet, in other queries, one may want to rank very popular entities lower. For example, entities of type ‘Country’ may have an extremely high number of incoming and outgoing relationships.

Similar to our assessment of *Rarity*, we define the *Popularity* of an association in terms of the popularity of its entities. We now define the *entity popularity*,  $p_i$ , of the  $i^{\text{th}}$  entity,  $e_i$ , in association  $A$  as:

$$p_i = \frac{|pop_{e_i}|}{\max_{1 \leq j \leq n}(|pop_{e_j}|)} \text{ where } typeOf(e_i) = typeOf(e_j) \quad (11)$$

where  $n$  is the total number of entities in the knowledge base. Thus,  $pop_{e_i}$  is the set of incoming and outgoing relationships of  $e_i$  and  $\max_{1 \leq j \leq n}(|pop_{e_j}|)$  represents the size of the largest such set among all entities in the knowledge base of the same class as  $e_i$ . Thus  $p_i$  captures the *Popularity* of  $e_i$ , with respect to the all other entities of its same type in the knowledge base. We now define the overall *Popularity* weight,  $P$ , of an association  $A$ , such that

$$P_A = \frac{1}{n} \times \sum_{i=1}^n p_i \quad (\mathbf{a}); \quad P_A = 1 - \frac{1}{n} \times \sum_{i=1}^n p_i \quad (\mathbf{b}), \quad (12)$$

where  $n$  is the number of entities (nodes) in  $A$  and  $p_i$  is the *entity popularity* of the  $i^{\text{th}}$  entity in  $A$ . If a user wants to favor popular associations,  $\mathbf{a}$  is used; in contrast, if a user wants to favor less popular associations  $\mathbf{b}$  is used. Thus,  $P_A$  is essentially the average *Popularity* or *non-Popularity* of all entities in  $A$ .

**Association Length.** In some queries, a user may be interested in more direct associations (i.e., shorter associations). Yet in other cases a user may wish to find indirect or longer associations. For example, money laundering involves deliberate innocuous looking transactions that may change several hands.

We define the *Association Length* weight,  $L_A$ , of an association  $A$ . If a user wants to favor shorter associations,  $\mathbf{a}$  is used, otherwise  $\mathbf{b}$  is used.

$$L_A = \frac{1}{length(A)} \quad (\mathbf{a}); \quad L_A = 1 - \frac{1}{length(A)} \quad (\mathbf{b}). \quad (13)$$

### 3.3 Overall Ranking Criterion

In the above sections, we have defined various association ranking criteria. We now define the overall association rank, using these criteria as

$$W_A = k_1 \times C_A + k_2 \times S_A + k_3 \times T_A + k_4 \times R_A + k_5 \times P_A + k_6 \times L_A, \quad (14)$$

where  $k_i$  ( $1 \leq i \leq 6$ ) add up to 1. Depending on the type of search, a user can change these weights to fine-tune the ranking criteria. In our experiments, we found useful to highly weight the context component and use the other ranking components as secondary criteria (Section 4.1 provides more details).

## 4 Experimental Results

The ranking approach presented in this work has been implemented and tested within SemDIS (Semantic DIScovery: Discovering Complex Relationships in the Semantic Web) project. The main components are illustrated in Fig. 1. The ranking prototype<sup>8</sup> utilized a modified version of TouchGraph<sup>9</sup> (applet for visual interaction with a graph) to define a query context. Prior to a query, a user can define contextual regions of the visualized ontology, with their associated weights using this graphical interface (see Fig. 2). Unranked associations are passed from the query processor to the ranking module. The associations are then ranked according to the ranking criteria defined by the user. The Web-based user interface allows the user to specify entities on which Semantic Association queries are performed. Optionally, the user can customize the ranking criteria by assigning weights to each individual ranking criterion.

The version of SWETO used for the evaluation contains instances including 2,902 cities, countries and states, 1,515 airports, 30,948 companies and banks, 1,511 terrorist attacks and organizations, 307,417 persons including researchers [2]. A large part of the dataset (extracted from DBLP<sup>10</sup>) contains 463,270 scientific publications, 4,256 conferences, journals, books etc. In total, SWETO contains approximately 800,000 entities and 1.5 million explicit relationships between them in RDF (e.g., 30,809 “located in” relationships), extracted from various Web sources (see SWETO Web site<sup>11</sup> for a complete listing of Web sources). The knowledge extraction is performed using Semagix Freedom, a commercial product which evolved from the LSDIS lab’s past research. The regular expressions are written to extract text from standard html, semi-structured (XML), and database-driven Web pages.

### 4.1 Ranking Evaluation

Due to the various ways to interpret Semantic Associations, we evaluated our ranking results with respect to those obtained by a panel of five human subjects who are graduate students at the LSDIS Lab. Although some of these students are members of SemDIS project and have background knowledge on Semantic Associations none of them are involved in the development of the ranking approach. The non-SemDIS students are introduced to concept of Semantic Associations briefly. For evaluation a quiet meeting room is selected and the interaction between the human subjects is not allowed. Note that no time constraint is specified for completion of the experiment.

---

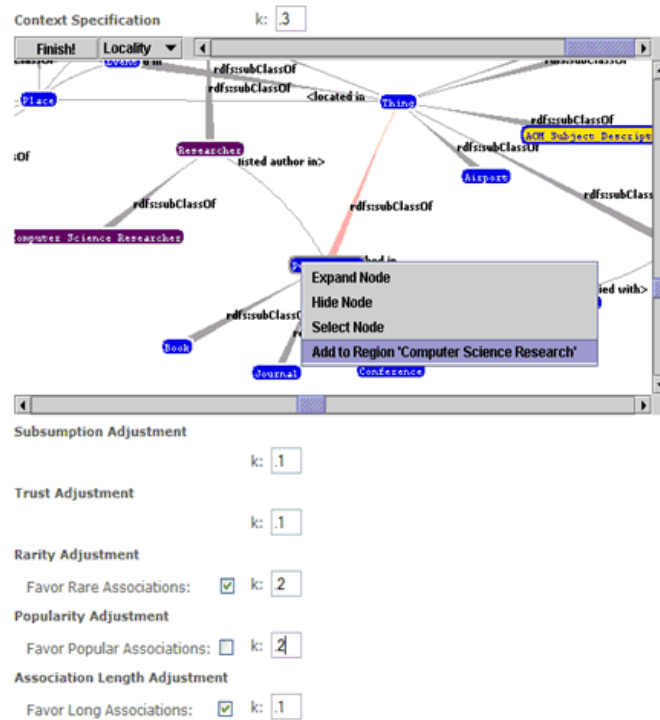
<sup>8</sup> <http://lsdis.cs.uga.edu:8080/vldbDemo2004/>

<sup>9</sup> TouchGraph, LLC <http://www.touchgraph.com>

<sup>10</sup> <http://www.informatik.uni-trier.de/~ley/db/>

<sup>11</sup> <http://lsdis.cs.uga.edu/Projects/SemDis/Sweto/>

The human subjects were given query results (randomly sorted) from different Semantic Association queries (each consisting of approximately 50 results where the longest associations were of length 12). Query results of each query included ranking criteria details, such as context, whether to favor short/long, rare/common associations, etc. as well as the type(s) of the entities and relationships in the associations in order to judge whether an association was relevant to the provided context. The human subjects ranked the associations based on this modeled interest and emphasized criterion. Given that different ranks were assigned to the same association, their average rank was used as a reference (target match).



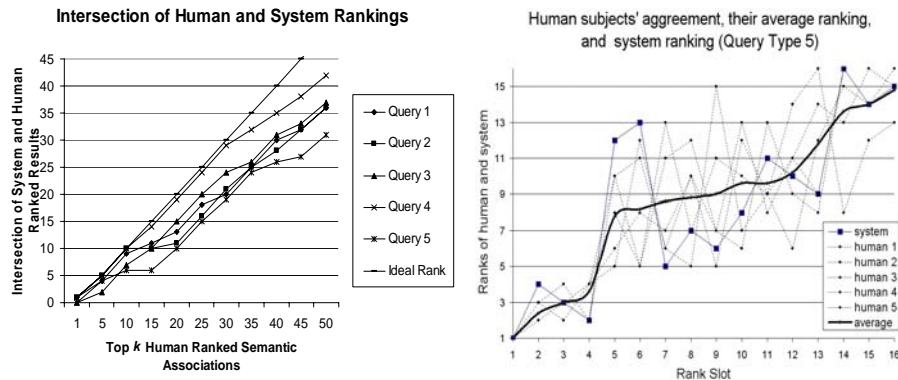
**Fig. 2.** User interface for context specification

Due to the large number of ways in which the criteria can be customized (e.g., favor long and rare vs. short and popular associations), we have evaluated five combinations. This is a small set, yet we feel it is a representative sample of the possible combinations. In each of the test queries, we have emphasized (weighted with  $k=0.4$ ) two of the criteria. The following list presents the ranking criteria and broader impact of each query.

Query	Query Details	Impact
1	Between two entities of type 'Person', with context of collegiate departments ('University', 'Academic Department', etc.); favors rare components.	Illustrates how the ranking approach can capture a user's interest in rare associations within a specific domain.
2	Between two entities of type 'Person'. Favors short associations in the context of computer science research.	Demonstrates the ability to capture the user interest in finding more direct connections (i.e., collaboration in a research project/area).
3	Between a 'Person' and a 'University', where common (not rare) associations are highly weighted and in the context of mathematics (departments and professors).	Shows the systems flexibility to highlight common relationships. This may be relevant, when trying to model the way a person relates to entities in a similar manner as the common public.
4	Between a 'Person' and a 'Financial Organization'; long associations and the financial domain context are favored.	Generally relevant for semantic analytics applications, such as those involving money laundering detection [10].
5	Between two 'Persons'; unpopular entities and the context of geographic locations are favored.	Demonstrates the system's capability to filter non relevant results which pass through highly connected entities, such as countries.

In order to demonstrate the effectiveness of the ranking scheme, we illustrate in Fig. 3 (a), the number of Semantic Associations in the intersection of the top  $k$  system and human-ranked results. This shows the general relationship between the system and human-ranked associations. Note that the plot titled 'Ideal Rank' demonstrates the ideal relationship, in which the intersection equals  $k$  (e.g., all of the top five system-ranked associations are included within the top five human-ranked associations). In three out of the five queries, the top human-ranked association directly matched the system assigned rank. Additionally, the top human-ranked association fell within the top five system-ranked associations in all five queries. The results are promising, given that out of the top ten human-ranked results, the system averaged 8.4 matches. It is also interesting to note that the minimum average distance of the system assigned ranks from that of the human subject's for a query (considered in relative order) was 0.55, while the maximum never exceeded 4.

Additionally, Fig. 3 (b) illustrates disagreement between human-ranked results. The x-axis represents Semantic Associations which are ranked first, second, etc. according to average rank scores of human subjects. Note that the x-axis does not contain their actual rank scores, but instead their corresponding ordering. On the other hand, the y-axis represents rank scores given by the system and human subjects. It is evident in the figure that there are varying levels of disagreement in human subjects ranking. Note that the system rank falls in its majority within the range of ranking disagreement of human subjects. We calculated the *Spearman's Footrule distance* measure [13] between the system rankings and average users' rankings. A value of zero would have indicated a perfect match; a value of 0.125 would have indicated that the ordering of results by the system is off by one position on the average, etc. We obtained a value of 0.23, which indicates that the ordering of ranked results was off by less than two positions on the average.



**Fig. 3.** (a) Measure of rank intersections; (b) disagreement among human-ranked results

Another experiment was conducted to evaluate the effects that different weights on equation (14) have on the ranking results. This third evaluation considered combinations of short or long, rare or common and popular or unpopular paths together with three different weights for the *context* parameter (0.8, 0.4, and 0.2). Using these parameters, a set of ranked Semantic Associations was generated by the system for a given Semantic Association query. Then the human subjects (same settings as above) were asked to rank these sets according to usefulness in the context of geographic locations. In our findings, the five most useful results by all human subjects always included (i) those with context weight of 0.8; (ii) 63% of those with context weight of 0.4; and one with context of 0.2. Shorter paths were often ranked highly in the most useful result sets (the role of popularity and rarity was minor in this particular experiment).

While this is a limited, initial evaluation, we can conclude that these results demonstrate the potential of the ranking algorithm and suggest that the approach is flexible enough to capture a user's preference and relevantly rank these complex relationships. As a future work, a Web based experiment can facilitate collecting feedback from a large number of users with different queries, context specifications, and weight combinations over a long period of time.

## 5 Related Work

Ranking semantic relationships is fundamentally different from ranking of documents in search results as those addressed in contemporary information retrieval approaches. In general, contemporary ranking approaches focus on finding relevance with respect to keywords for which there is no formal semantics and primarily rely on statistical/IR, link analysis, social networking and lexical techniques. Our ranking approach does not aim to rank documents, yet Semantic Associations, which are essentially sequences of properties linking entities.

Research in the area of Semantic Web ranking techniques includes [14], where the notion of "semantic ranking" is presented to rank queries returned within portals. Their technique reinterprets query results as "query knowledge-bases", whose

similarity to the original knowledge-base provides the basis for ranking. In our approach, the relevance of results depends on the criteria defined by a user.

## 6 Conclusions

Next generation technologies that facilitate getting actionable knowledge and information from semantic metadata extracted from Web documents, the deep Web and large enterprise repositories are emerging. Through our past and ongoing work in metadata extraction, as well as the definition and discovery for complex relationships on the Semantic Web, which we call Semantic Associations, we see the need for new ranking techniques to assess the relevance of these associations due to the large number of results from queries.

Since Semantic Associations are based on metadata extracted from heterogeneous documents and a set of potentially complex relationships between these metadata, we have discovered that there is no one way to measure their relevance. Thus, we have defined a flexible, query dependant approach for automatically analyzing and relevantly ranking the resulting associations. Additionally, through empirical evaluation of the ranking scheme, we have found our ranking scheme to be promising in capturing the user's interest and rank results in a relevant fashion.

**Acknowledgement.** We would like to thank all SemDIS project members, Semagix, and our collaborators at UMBC. This project is funded by NSF-ITR-IDM Award#0325464 titled 'SemDIS: Discovering Complex Relationships in the Semantic Web' and NSF-ITR-IDM Award#0219649 titled 'Semantic Association Identification and Knowledge Discovery for National Security Applications.'

## References

1. Guha, R., McCool, R.: TAP: An Semantic Web Test-bed. *Journal of Web Semantics*, 1(1) (2003)
2. Aleman-Meza, B., Halaschek, C., Sheth, A., Arpinar, B., and Sannapareddy, G.: SWETO: Large-Scale Semantic Web Test-bed, In *Proc. of the 16th International Conference on Software Engineering & Knowledge Engineering: Workshop on Ontology in Action*, Banff, Canada (2004)
3. Anyanwu, K., Sheth, A.:  $\rho$ -Queries: Enabling Querying for Semantic Associations on the Semantic Web. *The Twelfth Intl. World Wide Web Conference* (2003)
4. Aleman-Meza, B., Halaschek, C., Arpinar, I.B., Sheth, A.: Context-Aware Semantic Association Ranking, *First Intl. Workshop on Semantic Web and DBs*, Berlin, Germany (2003)
5. Vargas-Vera, M., Motta, E., Domingue, J., Lanzoni, M., Stutt, A., Ciravegna, F.: MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup. *13th Intl. Conf. on Knowledge Engineering and Management*, Sigüenza, Spain (2002)
6. Handschuh, S., Staab, S.: CREAM CREATING Metadata for the Semantic Web. *Computer Networks*. 42: 579-598, Elsevier (2003)
7. Dill, S., Eiron, N., Gibson, D., Gruhl, D., Guha, R., Jhingran, A., Kanungo, T., Rajagopalan, S., Tomkins, A., Tomlin, J. A., and Zien, J. Y.: SemTag and Seeker:

- Bootstrapping the semantic Web via automated semantic annotation. The Twelfth International World Wide Web Conference (2003)
8. Arce, I.: The Weakest Link Revisited. IEEE Security and Privacy, pp 72-76, March/April (2003)
  9. Lin, S., Chalupsky, H.: Unsupervised Link Discovery in Multi-relational Data via Rarity Analysis. The Third IEEE International Conference on Data Mining (2003)
  10. Anderson, R., Khattak, A.: The use of information retrieval techniques for intrusion detection. Proceedings of First International Workshop on the Recent Advances in Intrusion Detection, September (1998)
  11. Kleinberg, J.: Authoritative sources in a hyperlinked environment. Journal of ACM, 46(5) (1999)
  12. Brin, S., Page, L.: The Anatomy of a Large-Scale Hypertextual Web Search Engine, Computer Networks and ISDN Systems, 30(1-7):107-117 (1998)
  13. Diaconis, P., Graham, R.: Spearman's footrule as a measure of disarray, Journal of the Royal Statistical Society, Series B, 39(2):262-268 (1977)
  14. Stojanovic, S., Studer, R., Stojanovic, L.: An Approach for the Ranking of Query Results in the Semantic Web. 2nd Intl. Semantic Web Conference (2003)

## Biographies

- Boanerges Aleman-Meza is a PhD student in computer science at the University of Georgia. He received a master's degree in applied mathematics from the University of Georgia (2001). His bachelor's degree in computer science is from I.T.Ch. II, Mexico (1998). He is member of the IEEE Computer Society and the ACM. His research interests include databases, semantic technologies for search and analytics.
- Christian Halaschek-Wiener graduated with a MS degree from the University of Georgia. He is now a Ph.D. student in the Computer Science Dept. at the University of Maryland. Additionally, he received his BS in computer science from the University of Georgia (May 2002). He co-founded Semantic Innovations LLC and has served as CTO since its formation. His research interests include semantic technologies and interoperability, media content management and semantic applications in a variety of industrial sectors.
- I. Budak Arpinar is assistant professor of computer science at the University of Georgia (UGA), and is a member of the LSDIS lab. He received BSc, MSc, and PhD degrees from the Computer Engineering Department of Middle East Technical University (1991, 1993 and 1998, respectively). From 1999 to 2001, he was a post-doctoral research associate at UGA. His current areas of research include workflow management, semantic Web, semantic search and knowledge discovery and composition of Web services.
- Cartic Ramakrishnan is a PhD student in computer science at the University of Georgia. He graduated from the University of Pune, India with a BE in computer engineering. He is currently working with Amit Sheth at the LSDIS lab. He leads the research efforts in the ontology learning and semantic association discovery with other members at the LSDIS lab.

- Amit Sheth is a professor at the University of Georgia where he also directs the the LSDIS lab. Earlier, he served in R&D groups at Bellcore, Unisys, and Honeywell. In August 1999, Sheth founded Taalee, Inc., and managed as the CEO a Venture Capital funded a semantic web platform company based on technology licensed from at the LSDIS lab, and subsequently has served as the CTO of Semagix, Inc. which resulted from Taalee's merger/acquisition. Earlier he had founded Infocosm, Inc, which commercialized workflow technology. His research has led to several commercial products and applications. He has published over 180 papers and articles, given over 150 invited talks and colloquia including 20 keynotes, (co)-organized/chaired 15conferences/workshops, and served on over 100 program committees. He is on several journal editorial boards and is the EIC of the International Journal on Semantic Web and Information Systems.