Discovering and Ranking Semantic Associations over a Large RDF Metabase

Chris Halaschek, Boanerges Aleman-Meza, I. Budak Arpinar, Amit P. Sheth

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

Abstract

Information retrieval over semantic metadata has recently received a great amount of interest in both industry and academia. In particular, discovering complex and meaningful relationships among this data is becoming an active research topic. Just as ranking of documents is a critical component of today's search engines, the ranking of relationships will be essential in tomorrow's semantic analytics engines. Building upon our recent work on specifying these semantic relationships, which we refer to as Semantic Associations, we demonstrate a system where these associations are discovered among a large semantic metabase represented in RDF. Additionally we employ ranking techniques to provide users with the most interesting and relevant results.

1. Introduction

The focus of contemporary data and information retrieval systems has been to provide efficient support for the querying and retrieval of data. Significant academic and industrial research has now transitioned to mainstream search engines, such as Google, Vivisimo, and Teoma. With the increasing move from data to knowledge and the rising popularity of the Semantic Web vision [4], there is also significant interest and ongoing work in automatically extracting and representing metadata as semantic annotations to documents and services on the Web (e.g., [7]).

Given these developments, the stage is now set for the next generation of information systems that will facilitate getting actionable information from massive data sources. Through our NSF funded research project, SemDIS: Discovering Complex Relationships in the Semantic Web[1], we are developing such a system. Automatic metadata extraction resulting in semantically annotated Web entities via RDF[2], allows us to use ontologies and diverse knowledge-bases to "understand" in a limited way what a document is about (i.e. meaning of data). These knowledge-bases can then used for discovering previously unknown and potentially interesting relations between the entities, through a set of relationships between the metadata/annotations of the documents. Arguably, relationships are at the heart of semantics (e.g., [9]), lending meaning to information, making it understandable and actionable and providing new and possibly unexpected insights. One interesting type of complex relationships that we call Semantic Associations is defined next [3].

Definition 1 (ρ–Semantic Association): Two entities e_1 and e_n are ρ–semantically associated if there exists a sequence $e_1, P_1, e_2, P_2, e_3, \ldots e_{n-1}, P_{n-1}, e_n$ in an RDF graph where e_i, $1 \le i \le n$, are entities, P_j, $1 \le j < n$, are properties, and entities e_i and e_{i+1} are in relationship P_i.

Such Semantic Associations are illustrated in Figure 1. Other types of more complex Semantic Associations involve finding similarity patterns and are not discussed here for brevity. For simplicity, in the remaining sections of this document we will refer to ρ–Semantic Associations as Semantic Associations and leave the presentation of other types of associations to further papers.

Proceedings of the 30th VLDB Conference, Toronto, Canada, 2004

[1] http://lsdis.cs.uga.edu/Projects/SemDis/
[2] http://www.w3.org/TR/REC-rdfsyntax/

In the quest for finding Semantic Associations, users are frequently overwhelmed with too many results. For example, in our current semantic test-bed developed for open access and use by the Semantic Web research community, SWETO[3] (Semantic Web Technology Evaluation Ontology detailed in [2]), there are over 800,000 entities and 1.5 million explicit relationships among them. Simple Semantic Association queries between two entities result in hundreds of results and understanding the relevance of these associations requires comparable intellectual effort to understanding the relevance of a document in response to keyword queries.

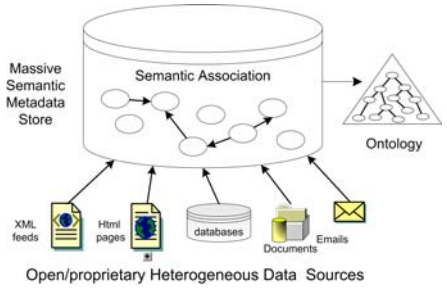

Figure 1: Semantic Associations

Therefore, it is important to locate interesting and meaningful relations and to rank them before presenting them to the user. The main goal of SemDIS is to demonstrate the discovery of Semantic Associations, as well as our ranking formalization presented in [1]. Thus, this shows the system's capabilities for semantic analytics across different sources of data through enabling users to uncover the most interesting associations by discovery and then ranking them in a relevant fashion.

The rest of the paper is organized as follows: Section 2 describes an overview of the ranking criterion; Section 3 presents the system implementation; lastly, Section 4 details the plans for the demonstration.

2. Ranking Criterion Overview

As mentioned earlier, ρ–Semantic Associations are essentially paths connecting two entities that can span across multiple domains and may involve any number of intermediate entities and relations. In this section, we describe various criteria for ranking these associations such that higher ranked associations are more relevant. Our approach defines an association rank as a function of various intermediate ranking components. For brevity, the following descriptions of these criteria exclude in depth examples and actual formulas; however, the details regarding the ranking approach can be found in [1]. We also note here that we are developing additional ranking criteria, which consider the popularity (number of incoming and outgoing edges) of the entities within a resulting association, as well as the frequency occurrence of the entities and relationships in an association.

2.1 Context

An association between two entities can pass through a variety of *regions*. By *regions*, we refer to areas of interest of a user in an ontology with respect to a specific query. For example, a user may be interested in the way two 'Persons' are related to one another in the domain of 'Computer Science Publications'. Taken from the SWETO ontology, the user would be most interested in associations that included concepts such as 'Scientific Publication', 'Computer Science Professor', etc. To capture this, we define the notion of a query *context*. This *context* is made up of various *regions* specified by the user. A context specification (discussed in Section 3.3) is thus used to capture a user's interest in order to provide him/her with the relevant knowledge among the numerous indirect relationships between the entities. Since the types of the entities are described using RDF Schema[4], we can use the associated class and relationship types to restrict our attention to the entities and relations of interest. Thus, by defining regions (or sub-graphs) of the RDF Schema, the areas of interest of the user are captured. Lastly, because a user can be interested in a variety of different regions with differing degrees on interest, we associate a weight with each region specified.

2.2 Subsumption

When considering entities in an ontology, those that are lower in the hierarchy can be thought of as more specialized instances of those further up in the hierarchy [6] (i.e. entities have more specific meaning). For example, in the SWETO ontology, the class 'Computer Science Professor' is a subclass of class 'Professor', which in turn is a subclass of class 'Person'. It is clear that a 'Professor' is a more specific type of 'Person'. Similarly, a 'Computer Science Professor' conveys more meaning than both 'Person' and a 'Professor'. This notion is captured through a criterion we refer to as 'Subsumption'. The intuition is assigning a higher rank to more "specific" entities in Semantic Associations.

2.3 Path Length

In some queries, a user may be interested in the most direct paths (i.e., short paths). This may infer a stronger relationship between two entities. Yet in other cases a user may wish to find possibly hidden, indirect, or discrete paths (i.e., long paths). The latter may be more significant in certain domains, for example, potential terrorist cells remain distant and avoid direct contact with one another in order to defer possible detection [5]. Hence, the user determines which *Path Length* influence, if any, should be used (this is largely domain dependent).

[3] http://lsdis.cs.uga.edu/proj/Sweto

[4] http://www.w3.org/TR/rdf-schema/

2.4 Trust

Due to the distributed nature of the data sources in a system of this type, various relationships and entities in a path originate from different sources. Some of these sources may be trusted while others may not. For example, Reuters could be regarded as a more trusted source on international news than some of the other news organizations. Thus, trust values are assigned by the user to relationships depending on the source. With trust values assigned to the knowledge, the system can rank paths coming from more trustworthy sources over those that are less trustworthy.

3. System Implementation

The SemDIS system has been designed so that it can be interacted with and almost entirely administered through a Web interface. The main components of the system are illustrated in Figure 2. The entire system, except for the Knowledge Extraction Module, is Web-accessible, and all code was written in Java. The following sections will detail the main components of the system architecture.

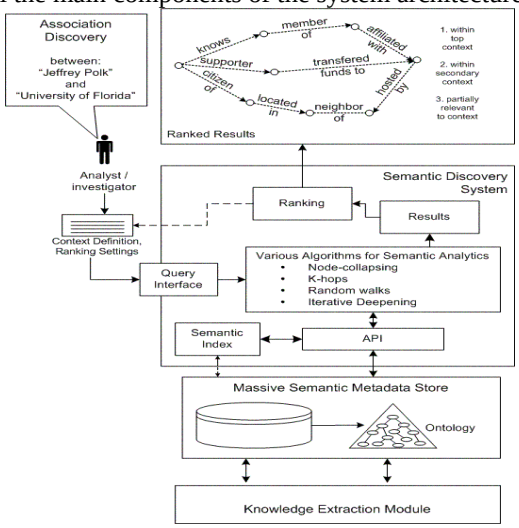

Figure 2: SemDIS System Architecture

3.1 Knowledge Extraction Module

In the SemDIS system, the knowledge extraction module (including the metadata extraction and storage) is implemented using Semagix[5] Freedom, a commercial product which evolved from the LSDIS lab's past research in semantic interoperability and SCORE technology [8]. Using this technology, we have created SWETO, a populated ontology with a large number of instances. It includes organizations, countries, people, researchers, conference, publications, etc., that are related by named relationships. Extractors are created within the Freedom environment, in which regular expressions are written to extract text from standard html, semi-structured (XML), and database-driven Web pages. As the Web pages are 'scraped' and analyzed by the extractors, the extracted entities are disambiguated and stored in the appropriate classes in the ontology. Additionally, provenance information, including source, time and date of extraction, etc., is maintained for all extracted data. We later utilize Freedom's API for exporting both the ontology and its instances in either RDF or OWL[6] syntax.

In order to query the knowledge base, we have implemented a Java API that allows for loading the ontology and its instances into main memory. Thus the system is provided with fast access to the data.

3.2 Knowledge Discovery

The query processing algorithms for discovery of Semantic Associations include adapted ideas based on k-hops, random walks, iterative deepening and node collapsing. The inputs for the query engine are two entities in the dataset. The query engine then finds all Semantic Associations between the entities of interest and forwards the results to the ranking module. We are developing heuristics to prune the search space based on semantics (e.g. through context), as well as index structures in order to reduce the time to perform a search.

3.3 Context Definition Module

As described in Section 2.1, the intuition behind a query context is that a user can specify at a high level the relevant types of data and relationships according to his/her needs; for example the 'Financial' or 'Scientific Publication' domains.

In this system, we utilize a modified version of Touchgraph[7], a Java applet for visual interaction with a graph, to define a query context. Essentially, a user can define regions, with their associated weights, classes, and properties using this interface. Figure 3 shows a screenshot of the context definition interface used within the system.

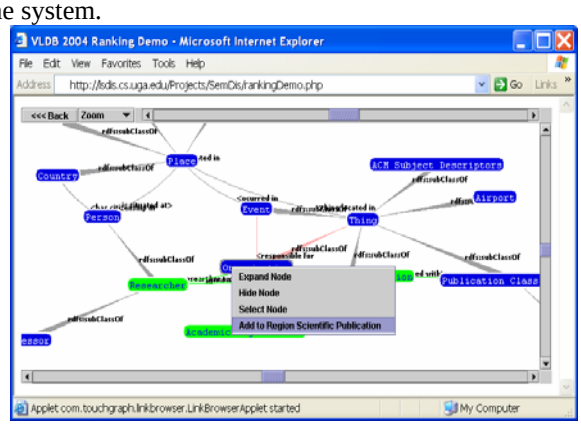

Figure 3: Context Definition Interface

[5] http://www.semagix.com

[6] http://www.w3.org/TR/owl-ref/

[7] http://www.touchgraph.com

3.4 Ranking Module

The ranking module is a Java implementation of the previously mentioned ranking approach. Unranked associations are passed from the query processor to the ranking module. The paths are then traversed and ranked according to the ranking criteria defined earlier. In the current implementation of the system, a user can interact with the ranking module so that s/he can specify context, whether to favor long or short paths, and which sources are the trustworthiest. Additionally, the user is also able to assign a weight to each of these individual ranking criteria.

3.5 User Interface

The user interface for the system is entirely Web accessible. The current implementation is servlet based (using Apache Tomcat), thus allowing the user to interact with the various system modules. A snapshot of the ranked results of a query to find Semantic Associations between Amit Sheth (one of the authors of this paper) and University of Georgia (UGA) is provided in Figure 4.

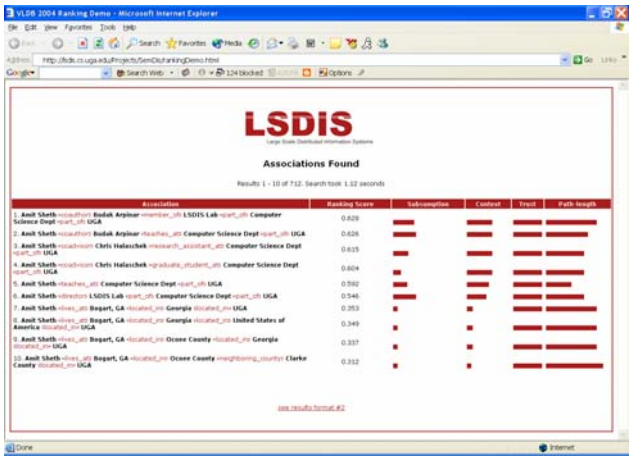

Figure 4: User Interface

4. Plan for Demonstration

The demonstration will be shown using a Web interface (driven by Apache Web server and Tomcat), in which users will be able to specify two entities and the system will return Semantic Associations between them. Additionally, the user will be allowed to define a query context through the Context Definition Interface, as well as customize the additional ranking criteria. Once the query is performed the ranked results will be displayed to the user through the Web interface. Alternatively, the results can be viewed in a random order to provide a comparison with ranked results. Hence, the demonstration will be a seamless integration of these facets of the systems.

5. Acknowledgements

We would like to thank other project members J. Miller, and K. Kochut, for their valuable comments, as well as Semagix, Inc. for providing its Freedom product. Additionally, we would like to acknowledge Meenakshi Nagarajan, Jason Lynes, and William Milnor for their work on the user interface. This project is funded by NSF-ITR-IDM Award # 0325464 titled 'SemDIS: Discovering Complex Relationships in the Semantic Web.'

6. References

[1] B. Aleman-Meza, C. Halaschek, I. B. Arpinar, and A. Sheth, "Context-Aware Semantic Association Ranking", 1st Intl. Workshop on Semantic Web and Databases, Berlin, Germany, September 7-8, 2003; pp. 33-50.

[2] B. Aleman-Meza, C. Halaschek, A. Sheth, I. B. Arpinar, and G. Sannapareddy, "SWETO: Large-Scale Semantic Web Test-bed", Intl. Workshop on Ontology in Action, Banff, Canada, June 20-24, 2004 (submitted).

[3] K. Anyanwu and A. Sheth, "ρ-Queries: Enabling Querying for Semantic Associations on the Semantic Web", 12th Intl. WWW Conference, Budapest, Hungary, 2003.

[4] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities", Scientific American, May 2001.

[5] V. Krebs, "Mapping Networks of Terrorist Cells". Connections, 24(3): 43-52, 2002.

[6] M. Rodriguez, and M. Egenhofer, "Determining Semantic Similarity among Entity Classes from Different Ontologies", IEEE Transactions on Knowledge and Data Engineering, 15(2): 442-456. 2003.

[7] U. Shah, T. Finin, A. Joshi, R. S. Cost, and J. Mayfield, "Information Retrieval on the Semantic Web", 10th Intl. Conference on Information and Knowledge Management, November 2002.

[8] A. Sheth, C. Bertram, D. Avant, B. Hammond, K. Kochut, and Y. Warke. (2002). "Managing semantic content for the Web." IEEE Internet Computing, 6(4), 80-87. 2002.

[9] A. Sheth, I. B. Arpinar, and V. Kashyap, "Relationships at the Heart of Semantic Web: Modeling, Discovering, and Exploiting Complex Semantic Relationships," Enhancing the Power of the Internet Studies in Fuzziness and Soft Computing, M. Nikravesh, B. Azvin, R. Yager and L. Zadeh, Springer-Verlag, 2003.