

# SemreX: a Semantic Association based Scientific Literature Sharing System

Pingpeng Yuan, Hai Jin, Yi Li, Binlin Chang, Xiaomin Ning, Wen Ni, Li Huang,  
Hao Wu

Service Computing Technology and System Lab  
Cluster and Grid Computing Lab  
School of Computer Science and Technology  
Huazhong University of Science and Technology, Wuhan 430074, China,  
hjin@hust.edu.cn

**Abstract.** Access to scientific literature information is a very important, as well as time-consuming daily work for scientific researchers. However, more and more literatures are available. It imposes a challenge to literature database. Current literature system mainly pays attention to a few explicit relationship among literature entities. In this paper, we present SemreX – a semantic association based literature sharing system with a single access point based on semantic web technologies. The concept of Semantic Association is proposed to reveal explicit or implicit relationships between semantic entities so as to facilitate researchers retrieving semantically relevant information. For the purpose of expression of semantic association, we propose a semantic association data model. Since it is very important for identification of semantic association to identify entities correctly, we develop some methods to identify entity name correctly. To real some implicit semantic association, we propose Wikipedia based classification. Based on semantic association among entities, we propose a entity ranking approach which help users find useful literature entities quickly.

**Keywords:** Semantic Association, Literature

## 1. Introduction

With the advances of science and technology, more and more literatures are available. Thus, it is very important for researchers to retrieve scientific literature efficiently and easily. However, Currently, there are many literature retrieval systems available, ACM [1], IEEE Xplore [2], CiteSeerX [3], Libra [4], CiteULike [5], DBLP [6], etc. However, those systems are generally based on explicit relationship among literature, such as author-publications, publication - sources. Those resulted in losing large amount of relevant information implied in literatures. Users can not find information, for example, publications belonging to same categories, relationship among researchers with same research interest, importance of publications and

publication sources and so on. Those implicit relationships available will help users find interest information more quickly.

The semantic web not only contains resources but also includes the heterogeneous relationships among them. Here, we propose a Semantic Association based literature sharing system – SemreX. The concept of Semantic Association is proposed to reveal explicit or implicit relationships among literature entities, such as authors, papers, publications, categories so as to facilitate users retrieving semantically relevant information, as well as context of literature entities.

The rest of the paper is organized as follows. The brief introduction of SemreX is given in section 2. Section 3 presents our data model: Semantic Association based Data Model. Section 4 presents the semantic data storage system of SemreX. It is very important to identify entities correctly, in section 5, we propose some of our methods to reach the goal. Classifying literatures can indicate many implicit relationship, we present a Wikipedia based classification approach in section 6. Section 7 gives the approach to rank entities which is also based semantic association. Finally, conclusions are given in Section 8.

## 2. System overview

The eldest SemreX was a Semantic Peer-to-Peer Scientific References Sharing System and developed in 2004 [7]. Later, we also developed a Client/Server based SemreX [8]. Now SemreX is Web based literature retrieval system which are different from previous versions.

Since there exist semantic association among any entities and semantic association indicate abundant information, SemreX need to discover and manage those semantic association efficiently. To reach the goal, firstly, SemreX automatically extracts and provides metadata and their explicit association from literatures. Moreover, SemreX also discover implicit relationship of literature entities, which implied in content and format of paper. Those explicit and implicit relationships are used to evaluate and cluster literature entities. Evaluated entities and their relationships are important criterions to match query term and rank search result or show interesting results to users.

Now, SemreX ([www.semrex.cn](http://www.semrex.cn)) is equipped with the following functions: automation extraction of metadata of literature, semantic associate based storage, literature classification and ranking, literature retrieval, and semantic data visualization etc. We also develop a high efficiency storage system for semantic data. Currently, SemreX manages metadata of more than 1 million literatures, which is integrated from public available datasets including SwetoDblp [9], Dbpedia [10], and metadata extracted from personal profiles etc. The dataset has approximately 20M triples. The detail information about the data set is listed in table 1. Moreover, the data set are still growing.

Table 1. Data of SemreX

Item	Number
Articles in Journal	386481

Articles in Proceeding	613760
Book chapters	4718
Conference/Journal/Book	13009
Persons	615416
Citations	3665053
Terms	108983
Categories	6676
URIs	1750283

### 3. Semantic Association Data model

In this section, we first formally define the data model that our work builds upon. The data model is based on the RDF/RDFS, but it is possible to extend our work to other formalisms as well.

As we say in section 1, there exists explicit and implicit relationship among entities. Although those relationships can be represented by RDF, it is not enough and requires the ability to transparently represent some data associated with RDF triples. For example, what is the probability that the relationship is correct or valid? We extend the RDF semantics and model the semantic web data as a directed graph. Before the presentation of semantic association data model, we firstly introduce the graph definition for RDF data model.

Suppose  $T$  is a RDF graph,  $T=(V^t, E^t, l_v^t, l_e^t)$ , where:  $V^t=\{v_x: x \in subj(T) \cup obj(T)\}$ ;  $l_v^t$  is the node labeling function,  $l_v^t(v_x) = x$ ;  $E^t=\{(s, o)|(s, p, o) \in T\}$  is the edge labeling function,  $l_e^t(s, o) = p$ .

Now, we will define Semantic Association Data model formally in the following:

Suppose labeling function set  $F=\{f_i|i \in N, i=0, 1, \dots, n-1\}$  are statement labeling functions. Now, the triples are consisted of RDF graph  $L=(V^l, E^l, l_v^l, l_e^l)$ , where:  $V^l=S_t \cup A_v$ ,  $S_t=\{st_t: t \in T\}$ ,  $A_v=\{a_x: x=f_i(t), t \in T\}$ ,  $l_v^l(v_x)=x$ ,  $E^l=\{(t, f_i(t))|t \in T\}$ ,  $l_e^l(t, f_i(t))=f_i$ .  $f_i$  can be any functions, for example, may be functions which indicate the possibility of statement or weight functions.

In SemreX, when ranking resources globally,  $f_i$  is defined as a weight function indicating the association strengths of subject and object (or property value) in a statement. The ranking mechanism takes relationship analysis and the edge weight functions into account. Since an extended random surfer has different transition probabilities along different types of association in this data model. Therefore, weights should rationally be determined to support the Markovian walk [11].

### 4. High Efficiency Semantic Data Storage

In order to manage RDF data efficiently and support RDF data analytical processing applications better, we make the following design decisions when we design semantic data storage system for SemreX:

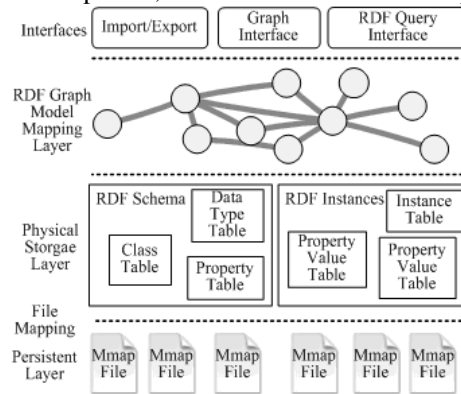
**Store, operate and query RDF data in main memory,** since accessing data directly in main memory is much faster than disk. Because memory is volatile, we use memory mapping mechanism to persist data in file system.

**Use a compact representation of RDF data.** Although memory is cheaper and computer has a large size memory, memory is shared among the processes running in the computer. By the help of compact representation of RDF data, SemreX can store a large scale RDF data set while it requires a small memory usage. Moreover, some optimizations, including URI encoding, native data type and so on are adopted to achieve this decision.

**Grid based partition and regroup RDF data.** Different from triple store or vertically partitioning, SemreX partition RDF triples according to schema information, such as type of class and property. The approach partitions the RDF data vertically and horizontally. Then SemreX regroups triples of multi-valued property type. By doing that, redundancy can be reduced, and certain triple retrieval and aggregation operations can be executed much faster.

**Provide a high-level query interface,** which supports structured queries for extracting data from the RDF repository, and provide some special syntaxes for search function and graph based retrieval or analytical queries.

The architecture of the storage system of SemreX is depicted in Figure 1. The system is consisted of 4 layers: persistent layer, physical storage layer, RDF graph model layer and interface layer. Due to volatility of memory, the persistent layer maps buffers in main memory to files, therefore provide a data persistent mechanism for SemreX. Thus, the data in memory can be preserved after the system shut down. Every table managed by physical storage layer is mapped to a file. During startup of the storage system of SemreX, it reads map files and constructs tables of physical storage layer. When data are updated, the data are written into map files.



**Figure 1.** System architecture of Storage System for SemreX

Physical storage layer manages tables and indexes. The tables and indexes are stored in buffers. Buffers are the basic component exchanged between all layers and components in the system. Each buffer need to be persistent are mapped to a file in file system, and upper level tables, indexes and temporary query execution results are stored in buffers The tables which physical storage layer manages include variable

length table, fixed length table, temporary table. The physical storage layer manages two kinds of RDF data: RDF instance and RDF schema.

The RDF graph model layer is a logical model layer, which provides a global view of RDF data. The view is implemented by union all the decomposed data stored in the tables of physical storage layer. On the top of the RDF graph model layer, there are several interfaces for the upper application, including import/export utilities, graph manipulation interface and query interface.

## **5. Named Entity Identification**

Before discovering semantic association among entities, it is very important to identify entities correctly. However, due to abbreviation and other reasons, it is very difficult to identify entities correctly. There are several reasons for ambiguity. One reason is abbreviations. Named entities in scientific literatures are generally abbreviations. One abbreviation may have multiple full names. Ambiguity also occurs in names of authors, organizations. Identifying entities wrongly lead to wrong semantic association.

For the purpose of identifying named entities correctly, firstly, we adopt Finite automata to identify named entities from papers. Considering the semantic completeness of content, we adopt the left-most matching approach to identify named entities. Some identified named entities are abbreviations. According to the analysis on occurrence of full names and their abbreviations, SemreX uses regular expressions to extract the mapping between full name of entities and their abbreviation from paper and their description. Moreover, considering some special cases, for example, Web Ontology Language is commonly abbreviated to OWL, we use the identified abbreviations of literature as local context to identify other abbreviations of literature.

Since there may exist several full names which have a same abbreviation, it is necessary for name disambiguation to consider adjacent entities, which are called as global context. Global context include category of adjacent entities, co-occurrence frequency of entities etc. SemreX combines global context with Hidden Markov Model to achieve the goal.

## **6. Wikipedia based Classification**

Since semantic association includes explicit relationship and implicit relationships, it is required to capture implicit semantic association such as texts which belongs to same categories. However, literatures generally belong to thousands of categories, it is difficult to reach the goal using traditional classification algorithms, such as SVM, kNN. Here, we adopt a Wikipedia-based approach to achieve a deep and multiple category classification for literature. Each article in Wikipedia describes a topic (or concept). Each article belongs to at least one category. Besides, categories are nested in a directed acyclic graph [12].

To adopt Wikipedia as knowledge bases to classify texts, we firstly need to extract just Wikipedia terminologies from documents, and then adjust the representation of document according to Wikipedia. We search each word of document  $d$  in the thesaurus of Wikipedia and decide whether the word matched a category or concept of Wikipedia. After matched categories and concepts occurring in document  $d$  are identified, document  $d$  can be represented as vector  $\varphi(d) = (<category\ tags>, <candidate\ concepts>, <related\ concepts>)$ , Where *category tags* means those words of  $d$  are category tags of Wikipedia, *candidate concepts* and *related concepts* indicate those words of  $d$  are concepts of Wikipedia.

There exist three kinds of relationship among concepts of Wikipedia: synonyms, hyponyms, and associative. Thus, we must adjust the representation of document  $d$ . We adopt the approach proposed in [13] to extend the vector space model for document  $d$ , namely:  $\varphi(d)' = \varphi(d) P \cdot \varphi(d)$ , where  $P$  is proximity matrix. Now, the feature vector of document  $d$  can be expressed as formula (1):

$$\varphi(d) = \{(c_1, tf(c_1, d)), \dots, (c_i, tf(c_i, d)), \dots, (t_j, tf(t_j, d))\} \quad (1)$$

where  $tf(c_i, d)$  is the term frequency of category  $c_i$  which occurs in document  $d$ ,  $tf(t_j, d)$  is the term frequency of terminology  $t_j$  occurring in document  $d$ .

After each document is expressed as the above forms, we can generate candidate categories based on the voting mechanism now. Generally, it is very common for a concept to subordinates to one or more categories. In these cases, we need to know how important a concept plays a role to decide a document belongs to a category. In other word, the concept vote how many percent of “grades” to a category. The weight that concept  $c_m$  votes on category  $y_k$  is calculated as formula (2):

$$contribution(y_k, c_m) = \frac{w(c_m)}{n} \times pw(y_k, c_m) \quad (2)$$

where  $w(c_m)$  is the weight of concept  $c_m$ ;  $n$  is the total of categories which  $c_m$  subordinated to;  $pw(y_k, c_m)$  indicates the importance that  $c_m$  to category  $y_k$ .

Since Wikipedia does not indicate importance of a concept to a category, namely weight. We adopt machine learning algorithm to determine that. After that, we get a candidate set of categories and corresponding weight and describe them as follows:

$$candidateSet: \{(y_1, w_c), \dots, (y_n, w_c)\} \quad (3)$$

There may exist many candidate categories. It is necessary to remove those categories with little possibility. Here, we filter the topic-unrelated categories from candidate set by clustering the sub-graph. For the purpose of convenient presentation, we define the undirected weight graph  $G=(V, E)$ , where candidate categories belong to  $V$  and theirs relationship belong to  $E$ , the weight indicates tightness of two vertexes. The tightness is defined as follows:

$$dis(y_i, y_j) = \begin{cases} \frac{w_{y_i} * w_{y_j}}{depth^2} & (y_i \text{ is the ancestor of } y_j) \\ \frac{w_{y_i} * w_{y_j}}{depth_1^2 + depth_2^2} & (otherwise) \end{cases} \quad (4)$$

Where  $w_{y_i}$  is the weight of  $y_i$ ,  $depth$  indicates the shortest distance between the two categories.  $depth_1$  and  $depth_2$  indicates the distance between category  $y_i$ ,  $y_j$  and the LCA (lowest common ancestor) respectively.

We do cluster analysis on  $G$  and may generate  $k$  group after clustering. We sum up the weights of all edges of a cluster, and the sum indicates the weight of the cluster. Now the set of the cluster are described as formula (5):

$$clusterSet = \{(r_1, w_1), (r_2, w_2), \dots, (r_k, w_k)\} \quad (5)$$

Now the weight of the category in the candidate set is calculated according to the contribution of the concepts that are directly subordinate to it. Except that, we must also consider the contribution of the sub-category. The contribution is related two factors: the weight of sub-categories and the distance between them in a path. Thus, the contribution is computed as formula (6):

$$contribution : Con(c_i, c_j) = weight(c_j) * \mu^{depth} \quad (6)$$

Where  $weight(c_j)$  is the weight of  $c_j$ ,  $depth$  is the shortest distance between  $c_i$  and  $c_j$ ,  $\mu$  is a back-off factor.

According to the contribution, we get the probability of every category and can determine which categories a document should belong to.

## 7. Semantic Association based Ranking and Search

In SemreX, we presented a framework for enabling ranked semantic search on the semantic web – RSS [11]. In this framework, the heterogeneity of semantic association was fully exploited to determine the global importance of resources. In addition, the search results can be greatly expanded with entities most semantically related to the query, thus able to provide users with properly ordered semantic search results by combining global ranking values and the relevance between the resources and the query.

The proposed semantic search model which supports inference is very different from traditional keyword-based search methods. Moreover, RSS also distinguishes from many current methods of accessing the semantic web data in that it applies novel ranking strategies to prevent returning search results in disorder.

This global ranking mechanism can be modeled as follows. In the data instance graph, a random surfer performs a Markovian walk which follows an edge  $e$  with the transition probability  $w(e)$  other than uniform probability applied in standard PageRank, or gets bored and randomly jumps to a new node. The personalization vector  $r$  can bias the surfer. The dampening factor  $\alpha$ , usually set to 0.85, is applied to perturb the matrix computation. Similar to the PageRank model where each web page has a global PageRank value, each node  $v \in V$  has a global ranking value  $g(v)$  which indicates the importance of  $v$  in the data instance graph. The vector  $g = [g(v_1), \dots, g(v_n)]^T$  is computed as follows:

$$g^{(m)} = \alpha P g^{(m-1)} + \frac{1 - \alpha}{|V|} r \quad (7)$$

Considering the edge weighs, the matrix  $P$  is calculated as follows:

$$p_{ij}=w(e), \text{ where } e: v_j \rightarrow v_i \in E$$

## 8. Conclusion

In this paper, we present SemreX – a semantic association based literature sharing system. The concept of Semantic Association is proposed to reveal explicit or implicit relationships between semantic entities so as to facilitate researchers retrieving semantically relevant information. For the purpose of expression of semantic association, we propose a semantic association data model. We develop methods to identify entity name correctly and classify literatures so as to reveal implicit semantic association. We also propose a semantic association based entity ranking approach which help users find useful literature entities quickly.

**Acknowledgement.** This work is funded by the National 973 Basic Research Program of China under grant No. 2003CB317003.

## References

1. The ACM Digital Library, <http://portal.acm.org/portal.cfm>.
2. IEEE Xplore, <http://ieeexplore.ieee.org/Xplore/home.jsp>
3. Microsoft Academic Search, <http://libra.msra.cn/>
4. CiteSeerX, <http://citeseerx.ist.psu.edu/>
5. CiteULike, <http://www.citeulike.org/>
6. DBLP Computer Science Bibliography, <http://dblp.uni-trier.de/>
7. H. H. Chen, H. Jin, X. M. Ning, P.P. Yuan, H.Wu, Z. X. Guo. SemreX: A Semantic Similarity Based P2P Overlay Network, Journal of Software, issue 5, 2006
8. X. M. Ning, H. Jin, H. Wu, *SemreX: towards large-scale literature information retrieval and browsing with semantic association*, In Proc. of IEEE International Conference on e-Business Engineering (ICEBE'06), Shanghai, China, October 24-26.
9. B. Aleman-Meza, F. Hakimpour, I. B. Arpinar, A.P. Sheth. *SwetoDblp Ontology of Computer Science Publications*. Journal of Web Semantics 5(3): 2007, pp: 151-155
10. S. Auer, C. Bizer, G. Kobilarov, et al. *DBpedia: A Nucleus for a Web of Open Data*. In Proc. of ISWC'07, 2007, pp: 715-728
11. X. M. Ning, H. Jin, H. Wu, *RSS: a framework enabling ranked search on the semantic web*, Information Processing and Management, Information Processing & Management, Vol. 44 (2), March 2008, pp: 893-909
12. E. Gabrilovich, S. Markovitch, *Overcoming the brittleness bottleneck using wikipedia: enhancing text categorization with encyclopedic knowledge*, In Proc. of AAAI, Boston, Massachusetts, 2006, pp. 517-524.
13. P. Wang, D. Carlotta. *Building semantic kernels for text classification using Wiki*. In: Proc. of KDD'08, Las Vegas, Nevada, USA, 2008. 713~721