



RSS: A framework enabling ranked search on the semantic web

Xiaomin Ning, Hai Jin ^{*}, Hao Wu

Services Computing Technology and System Laboratory, Cluster and Grid Computing Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China

Received 27 December 2006; received in revised form 2 March 2007; accepted 3 March 2007

Abstract

The semantic web not only contains resources but also includes the heterogeneous relationships among them, which is sharply distinguished from the current web. As the growth of the semantic web, specialized search techniques are of significance. In this paper, we present RSS—a framework for enabling ranked semantic search on the semantic web. In this framework, the heterogeneity of relationships is 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. The experimental results show that the framework is feasible and can produce better ordering of semantic search results than directly applying the standard PageRank algorithm on the semantic web.

© 2007 Elsevier Ltd. All rights reserved.

Keywords: Relationship analysis; Rank; Semantic search; Semantic web

1. Introduction

The semantic web (Berners-Lee, Hendler, & Lassila, 2001) is an extension of the current web, based on the idea of exchanging information with explicit, formal and machine-accessible descriptions of meaning. As the scale of the web grows, search engines have played important roles in the web infrastructure. It is reasonable to say that the specialized search engines will be indispensable to find resources encoded in semantic web languages such as RDF(S) and OWL with the growth of the semantic web.

In the current web, web pages are connected with each other to form a so-called web graph. Traditional web search technologies such as PageRank (Brin & Page, 1998) and HITS (Kleinberg, 1999) are typically based on hyperlink analysis in the web graph and query keywords processing. Ranking of search results is critical for these technologies since a web search usually returns too many results. The success of the Google

^{*} Corresponding author. Tel.: +86 27 8754 3529; fax: +86 27 8755 7354.

E-mail addresses: ningxm@hust.edu.cn (X. Ning), hjin@hust.edu.cn (H. Jin), haowu@hust.edu.cn (H. Wu).

search engine is mainly due to the very effective PageRank technique which calculates the importance of web pages on the base of the idea that a page is important if it is pointed to by other important pages.

In the semantic web, the information space is complex since it contains not only resources but also the relationships among them. We can see the major differences between the semantic web and the current web from the following W3C description about the semantic web (Herman, 2006):

“The Semantic Web is about two things. It is about common formats for interchange of data, where on the original Web we only had interchange of documents. Also it is about language for recording how the data relates to real world objects.”

Some query languages have been proposed to access the semantic web data, e.g., RQL, RDQL, SeRQL, SPARQL, etc., and most of them use a SQL-like declarative syntax to retrieve query results as a set of RDF triples. However, these SQL-like query languages may impose cognitive overload (Nielsen, 1990) on users since they usually are not familiar with domain ontologies or semantic web languages. Moreover, these techniques do not provide any ranking strategies about ordering the query results. As the number of ontologies available in the semantic web increases rapidly, it is necessary to find specialized ranking mechanisms to properly order search results.

Unfortunately, it is not appropriate to directly transplant techniques successfully used in the web or information retrieval area, such as PageRank (Brin & Page, 1998), HITS (Kleinberg, 1999), VSM (Berry, Drmac, & Jessup, 1999), LSI (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990), to the semantic web search and ranking. First, all links between web pages are considered equally in standard PageRank. However, one unique characteristic of the semantic web is the heterogeneity of relationships between resources. Different types of relationships imply different importance for the connected resources. For example, in the scientific research domain a research paper may be written by several authors and cited by other papers. There are two different types of relationships: *has-author* and *cited-by*, which have different semantics. It is obvious that the number of citations is more important than the number of authors for a paper. However, the PageRank algorithm assumes that authors and citations have equal impacts on the evaluation of the paper. Therefore, the ranking mechanisms on the semantic web should depend on not only the number of relationships connecting resources but also the heterogeneity of relationships. Second, traditional search techniques usually focus on finding documents/pages with the query keywords, while the semantic features of relationships among resources are ignored. Anyanwu, Maduko, and Sheth (2005) propose that relationships should play an important role in the semantic web because a single object is intensely uninteresting. Therefore, it is necessary to provide users with results most semantically related to the query requests, even if the search results do not contain the query keywords explicitly. For example, it is reasonable to get papers which titles or full-texts contain the term “semantic” when we query with the keyword “semantic” in the computer science domain. However, it is very helpful to provide answers with entities typed “Journal”, e.g., “*Artificial Intelligence Review*”, “*IEEE Transactions on Knowledge and Data Engineering*”, or entities typed “Conference”, e.g., “*WWW 2006*”, “*AAAI 2006*”, etc., since many publications on “semantic” appear in these journals or conferences though their names do not directly contain the term “semantic”. Also, it is useful to retrieve entities typed “Author”, e.g., “*Tim Berners-Lee*”, “*Eric Miller*”, etc., who are famous in the “semantic” research area.

This paper provides RSS—a framework which enables ranked semantic search in the semantic web. In this framework, the search results can be greatly expanded with entities which are most semantically related to the query. Moreover, the heterogeneity of relationships is fully exploited to determine the importance of resources, thus supporting semantic search and providing users with properly ordered search results. An extended spreading activation algorithm (Cohen & Kjeldsen, 1987; Crestani, 1997; Rumelhart & Norman, 1983; Salton & Buckley, 1988) is proposed to retrieve resources most semantically related to the query, thus supporting inference while searching the semantic web. We evaluate the proposed algorithms over real-world data from CiteSeer metadata which covers literatures in the field of computer science and computer technology with about 800,000 publications (<http://citeseer.ist.psu.edu/oai.html>).

The rest of this paper is organized as follows. Section 2 discusses related works. We define the data model and describe the ranking formulation in Section 3. In Section 4, we present our ranked semantic search method. Section 5 describes experimental results. We conclude the paper and point out future directions in Section 6.

2. Related work

Technologies such as PageRank (Brin & Page, 1998) and (Kleinberg, 1999) have been successfully applied to get good ranked search results in the web. The heuristic underlying these approaches is that pages with many inlinks are more likely to be of high quality than pages with few inlinks. The PageRank (Brin & Page, 1998) algorithm which has led to the popular Google search engine is based on the random surfer model which follows the hyperlinks of web pages by clicking on them or sometimes gets bored and jumps to another web page not hyperlinked by the current web page. Based on the random surfer model, the PageRank algorithm computes the rank (indicating popularity rather than relevance) of each web page by iteratively propagating the rank until convergence. The HITS algorithm proposed by Kleinberg (1999) invokes a traditional search engine to obtain a set of pages relevant to the query, expands this set with its inlinks and outlinks, and then attempts to find two types of pages, hubs (pages that point to many pages of high quality) and authorities (pages of high quality). The problem common to both PageRank and HITS is topic drift (Langville & Meyer, 2005) because they give the same weight to all edges.

The spreading activation model, originated from the field of psychology (Rumelhart & Norman, 1983) and widely used in the area of artificial intelligence as a processing framework for semantic or associative networks (Lehmann, 1992), has spread to other areas, especially in information retrieval (Cohen & Kjeldsen, 1987; Crestani, 1997; Salton & Buckley, 1988). The pure spreading activation model (Crestani, 1997) is made up of a network data structure similar to the semantic network upon which simple processing techniques are applied. The activation process starts by placing a specified activation weight at some starting nodes. The initial activation weight spreads through the network along the links originating at the starting node. The activation weight of a node is a function of the weighted sum of the inputs to that node from directly connected nodes. This activation process iterates until a termination condition is achieved. The results of the algorithm contain the nodes ordered by their activation values. The pure spreading activation model has some serious drawbacks (Crestani, 1997) and many heuristics or inference rules are proposed to enhance the basic model, e.g., the constrained spreading activation (Cohen & Kjeldsen, 1987), thus it can support inference better. Some rules-based methods have been proposed to support inference on the semantic web. Mayfield and Finin (2003) presents a framework for tightly integrating search and inference which supports both retrieval-driven and inference-driven processing on the semantic web. The inference and reasoning functionality of this framework is based on the use of DAMLJessKB (Kopena & Regli, 2003) which reads each DAML+OIL file as a collection of RDF triples and Jess as a forward chaining production system to apply rules to triples.

Link analysis and ranking technologies in the web graph have been generalized for analyzing structural databases (Balmin, Hristidis, & Papakonstantinou, 2004; Cohen, Mamou, Kanza, & Sagiv, 2003; Guo, Shao, Botev, & Shanmugasundaram, 2003). ObjectRank (Balmin et al., 2004) applies authority-based ranking to keyword search in databases modeled as labeled graphs. In ObjectRank, authority originates at the objects containing the keywords and flows to objects according to their semantic connections. Each node is ranked according to its authority w.r.t. the particular keywords. However, ObjectRank creates ranking vectors for each keyword on each object, which imposes heavy computation and storage overhead. XSearch (Cohen et al., 2003) is a semantic search engine for XML which has a simple query language and returns semantically related document fragments that satisfy the query. Query answers are ranked using extended information retrieval techniques and are generated in an order similar to the ranking. XRANK (Guo et al., 2003) proposes a method to rank XML elements using the link structure of the database. XRANK computes rankings at the granularity of an element because XML keyword search queries return elements. Also, XRANK considers element-to-element links in addition to document-to-document links.

Some ranking techniques (Alani, Brewster, & Shadbolt, 2006; Anyanwu et al., 2005; Ding et al., 2004; Stojanovic, Studer, & Stojanovic, 2003; Vallet, Castells, Fernandez, Mylonas, & Avrithis, 2007) for the semantic web have been proposed. AKTiveRank (Alani et al., 2006) is an experimental system for ranking ontologies based on some measures that assess the ontology in terms of how well it represents the concepts of interest. Anyanwu et al. (2005) propose an approach and framework for ranking complex relationships resulting from a relationship search. However, their ranking method mainly focuses on relationships other than entities. Vallet et al. (2007) presents a novel personalized content management system which is to improve the retrieval process by taking into account user preferences on the semantic context of ongoing user activities. In the

context of the personalized retrieval, the personal relevance measure is combined with query-dependent and user-neutral search result rank values to produce the final rank score for a document. In addition, they apply a form of constrained spreading activation to explore semantic paths using a semantic expansion of both user preferences and the context. Swoogle (Ding et al., 2004) is a crawler-based indexing and retrieval system for the semantic web documents, i.e., RDF or OWL documents. It analyzes the documents to compute useful metadata properties and relationships between them. However, the ranking in Swoogle is just at the document level. Stojanovic et al. (2003) present an ontology-based ranking scheme for ranking entities in the semantic web. This scheme determines the relevance of the entities based on their specificity. However, the importance of resources based on the global information space is not considered during the ranking.

Semantic search (Castells, Fernandez, & Vallet, 2007; Guha, McCool, & Miller, 2003; Rocha, Schwabe, & Aragao, 2004) plays an important role in the vision of the semantic web. Guha et al. (2003) present the idea of semantic search aimed at improving searches of documents in the semantic web by augmenting the results of traditional searches with relevant data obtained from multiple sources in the semantic web. Rocha et al. (2004) seek to find important related entities to a given set of keywords using a spreading activation mechanism with the domain knowledge provided by experts. However, the initial values of starting nodes mainly depend on the relevance between the query and nodes, without the consideration of overall importance. In addition, all types of relationships are considered to have the same relative weight in the measures. Castells et al. (2007) presents an ontology-based retrieval approach which is based on an adaptation of the classic vector-space model, including an annotation weighting algorithm, and a ranking algorithm to overcome the limitations of purely boolean ontology-based retrieval models and many prior semantic search proposals. The approach shows clear improvements w.r.t. keyword-based search. The RSS model differs from this approach most in that RSS considers the heterogeneity of relationships between resources for supporting ranked semantic search while this approach applies finer-grained domain ontologies to improve keyword-based full-text search. In addition, semantic annotation of the web content addressed by Castells et al. (2007) is not involved in this paper. KIM (Kiryakov, Popov, Terziev, Manov, & Ognyanoff, 2004; Popov, Kiryakov, Ognyanoff, Manov, & Kirilov, 2004) gives a good work on automatic semantic annotation in the semantic web. Moreover, the KIM platform implements IR-like indexing and retrieval which is further extended using the ontology and knowledge about the specific entities based on these annotations.

3. Ranking

In this section, we first formally define the data model that our work builds upon. Then, we discuss how to apply relationships analysis and edge weights to rank the global importance of resources in the data model. Finally, the formulation of global ranking on resources is presented.

3.1. Data model

Some formalisms have been proposed for representing semantic web data, e.g., RDF/RDFS, DAML+OIL, OWL, etc. This paper focuses on the RDF/RDFS family, but it is possible to extend our work to other formalisms as well (Patel-Schneider & Horrocks, 2006).

RDF represents statements about web resources with the form (*subject*, *property*, *object*) named an RDF triple which asserts that a resource, the *subject*, has a *property* whose value is the *object*. This model can be represented as a labeled directed graph (Hayes, 2004). We extend the RDF semantics and model the semantic web data as a weighted directed graph $G = (V, E, f)$, where $V = \{v_i : i \in \mathbb{N}\}$ is a finite set of nodes representing the resources or literals, $E \subseteq V \times V$ a set of edges representing properties, and $f : E \rightarrow \mathbb{R}^+$ a edge weight function indicating the relationship strengths of properties. Therefore, an RDF triple is extended to a 4-tuple (s, p, o, w) where s , p and o are *subject*, *property* and *object*, respectively, and $w = f(p) \in \mathbb{R}^+$ is the weight of p . RDF Schema defines classes and properties that describe groups of related resources and relationships between resources. *Classes* are sets of resources and elements of a class are instances of that class. The initial edge weight assignment depends on the schema graph, i.e., the types of properties, other than instances. For example, the properties *has-author* and *cited-by* have different edge weight functions, however, for the specific

property *has-author*, the edge weight function is determined though there are many different instances of publications and authors related to the property *has-author*.

In this model, each resource has a global ranking value which determines how important the resource is in the graph. The ranking mechanism to be detailed is based on a novel ranking algorithm which takes relationships analysis and the edge weight functions into account.

3.2. Edge weights

In the data model, edges represent properties and the weights of edges indicate the strength of relationships. An extended random surfer has different transition probabilities along different types of edges in this data model. Therefore, edge weights should rationally be determined to support the Markovian walk. Let us consider the data schema graph example of Fig. 1 in the scientific literature domain.

In Fig. 1, there are three kinds of Class, i.e., Author, Publication and Conference. These Classes are interconnected through heterogeneous relationships, e.g., a publication is written by a set of authors or cited by other papers or similar to many papers in contents or presented in conferences. These relationships have different semantics, thus the transition probabilities may not be the same. Each type of relationship T in the schema graph is assigned a weight $w(T)$. For example, the impact of a paper is affected much by papers citing it other than papers it cites, therefore, the edges of *cited-by* and *cite* could have transition probabilities 0 and 0.6, respectively. The assignment of edge weights is a continuous trial and may be complemented by domain experts/engineers. There have been various attempts (Agarwal, Branson, & Belongie, 2006; Burges et al., 2005; Diligenti, Gori, & Maggini, 2005) to assign edge weights automatically, though still complex and time-consuming. This work is out of the scope of this paper. Fortunately, we only need to manually assign edge weights on the schema level other than on the instance level where the assignment is accomplished automatically according to the schema.

For a Markovian walk, the edge weights given a source node v_i in the data schema graph should form the following probability distribution:

$$\sum_{j \in S_i} w_{ij} = 1, \quad \text{where } S_i = \{\text{target nodes from } v_i\} \quad (1)$$

However, there is a problem with the transition probability distribution because the sum of a schema node may not be strictly equal to 1. For example in Fig. 1, the schema node *Author* has only a transition probability 0.4 to the node *Publication*. Similar to the PageRank algorithm, we add a perturbation factor α which models a user's "teleportation" tendency to randomly select a new resource other than following connected relationships which is depicted in Section 3.3.

A data instance graph should adhere to the data schema graph. The ranking focuses on the instances level other than the schema level. In the data instance graph, given the source node v_i and the edge $e : v_i \rightarrow v_j$ with the relationship type T , the edge weight $w(e)$ is computed as follows:

$$w(e) = \begin{cases} \frac{w(T)}{\text{OutDegree}(v_i, T)} & \text{if } \text{OutDegree}(v_i, T) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

In Eq. (2), $w(T)$ is the weight of the relationship typed T in the schema graph, and $\text{OutDegree}(v_i, T)$ is the number of outlinks typed T from v_i .

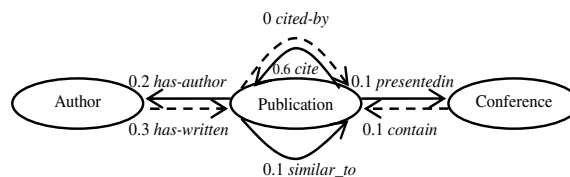


Fig. 1. The data schema graph.

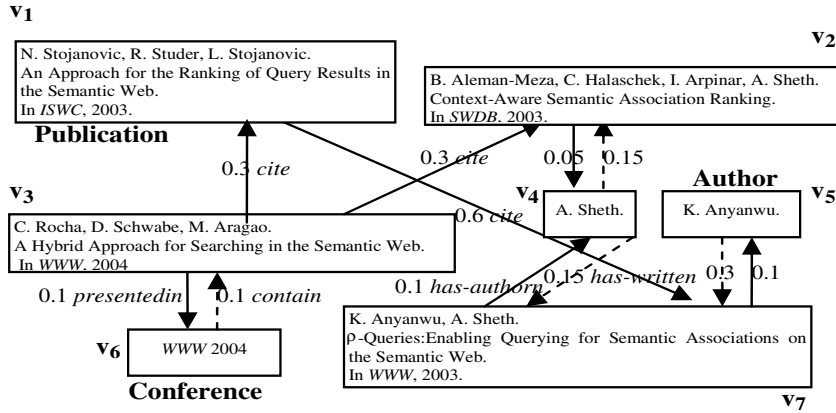


Fig. 2. The data instance graph.

Fig. 2 gives an example of the data instance graph. From Fig. 2, it is clear that the edges connecting the paper v_3 to the cited papers, i.e., v_1 and v_2 , have the weight 0.3 while in Fig. 1 the weight of the relationship *cite* is 0.6. This is because the paper cites two papers in the instance graph, thus dividing the edge weight of *cite* in the schema graph by $OutDegree(v_i, T) = 2$ for Eq. (2). For simplicity, some nodes and edges are omitted in the graph. Therefore, the edge weight of the relationship “*contain*” connecting the conference “*WWW 2004*” to the paper v_3 should be actually much less than 0.1 as in Fig. 2 since the conference contains many other papers which are not shown in Fig. 2. The same holds for the author nodes because they may have written many other publications.

3.3. Global ranking on resources

To globally rank the importance of resources in the data instance graph, the relationships analysis and edge weights are both taken into account. 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, for example, preferring prestige conferences or favorite authors in the scientific literature domain. The dampening factor α , 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 value $g(v)$ is the stationary probability for this Markovian walk. The vector $g = [g(v_1), \dots, g(v_i), \dots, g(v_n)]^T$ is computed as follows:

$$g^{(m)} = \alpha P g^{(m-1)} + \frac{1 - \alpha}{|V|} r \quad (3)$$

Considering the edge weights, the matrix P is calculated as follows:

$$P_{ij} = w(e), \quad \text{where } e : v_j \rightarrow v_i \in E \quad (4)$$

In Eq. (4), $w(e)$ is the edge weight of $e : v_j \rightarrow v_i$ which is computed through Eq. (2). To converge the computation, the following measurement is used:

$$\sigma = \|g^{(k+1)} - g^{(k)}\|_2 = \sqrt{\sum_{1 \leq i \leq n} |g(v_i)^{(k+1)} - g(v_i)^{(k)}|^2} < \epsilon \quad (5)$$

In Eq. (5), ϵ is the convergence threshold and the smaller value slows down the convergence rate, thus taking more computation time. We give the following example to illustrate the global ranking mechanism:

Example 1. Consider the example of Fig. 2. First we create the matrix P as follows:

$$P = \begin{matrix} & \begin{matrix} v1 & v2 & v3 & v4 & v5 & v6 & v7 \end{matrix} \\ \begin{matrix} v1 \\ v2 \\ v3 \\ v4 \\ v5 \\ v6 \\ v7 \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0.6 \\ 0 & 0 & 0 & 0.05 & 0 & 0 & 0 \\ 0.3 & 0.3 & 0 & 0 & 0 & 0.1 & 0 \\ 0 & 0.15 & 0 & 0 & 0 & 0 & 0.15 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.3 \\ 0 & 0 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.1 & 0 & 0 \end{pmatrix} \end{matrix}$$

Let the initial global ranking values g be the vector $g^{(0)} = [1, 1, \dots, 1]^T$ and $r = [1, 1, \dots, 1]^T$. We choose $\alpha = 0.85$ and form

$$g^{(1)} = \alpha P g^{(0)} + \frac{1 - \alpha}{|V|} r = 0.85 * P * g^{(0)} + \frac{0.15}{7} r$$

Thus, we get

$$g^{(1)} = [.5314, .5739, .6164, .2764, .2764, .1064, .1914]^T$$

Then we set $\epsilon = 0.001$ and iterate the computation. After 40 iterations, the final global ranking values g is computed:

$$g = [0.5335, 1.2099, 0.3011, 2.1370, 1.1091, 0.2004, 2.2580]^T$$

We can see that v_7 has the highest value 2.2580 since it is connected by v_1 via *cite*. In addition, v_7 is connected by v_4 via *has-written* who is also one author of another paper v_2 . The node v_6 , i.e., “*WWW 2004*”, has the lowest value 0.200384 since it just contains one paper v_3 in the example. In reality, the ranking value of v_6 should be much more higher as the conference contains many other papers about the web which are not shown in Fig. 2.

4. Search method

In this section we present our search method for generating ordered search results w.r.t. a query request. Section 4.1 defines the query and answer model. Section 4.2 describes the general semantic search process which extends the spreading activation algorithm. The generation of the final ordered semantic search results is also described.

4.1. Query and answer model

Generally, a query request $R = \{Q; C\}$ comprises two parts where Q denotes the query requirement and C specifies the answer constraint. Q consists of $n \geq 1$ search items $s_1, \dots, s_i, \dots, s_n$, where s_i takes the form $t_i:k_i$ which requires the keyword k_i to be in the target resource connected by the property t_i . C explicitly specifies the expected type of search results. If C is set to *null*, any type of search results can be returned. The query model can be extended to a complete Google-like keyword search interface if Q takes the form $k_1, \dots, k_i, \dots, k_n$ and C is set to *null*. In this paper, we mainly focus on the former query model.

Let us give examples to explain the above model. When we search publications written by the author “Hai Jin” and presented in the conference “*WWW 2006*”, the query can be denoted by $R = \{has-author:“Hai Jin”, presentedin:“WWW 2006”; Publication\}$. Consider the next example, finding most related resources about the author “Hai Jin”. In this example no particular type of results is specified, therefore, any resources most semantically related could be returned. The query is represented as $R = \{has-author:“Hai Jin”; null\}$. The second example is a typical semantic search. The answer model $A = Results(R) \subseteq V$ has been partly described in R . The final answer results are a sequence of resources matching the query R . If C is set to *null*, any type of most semantically related resources could be returned. However, the results should be properly ordered

according to a mechanism which can measure the importance and the relevance to R . The mechanism, named ranked semantic search, is the emphasis of this paper.

4.2. General search algorithm

Let the query $R = \{Q; C\}$ and the data instance graph $G = (V, E, f)$ be given. The search algorithm attempts to find ordered semantic results $A = Results(R) \subseteq V$. We first outline the key procedures of the general search algorithm:

- We mainly extend the spreading activation to guide the semantic search in G . To determine starting nodes of the spreading process, we combine the nodes global importance and the relevance between R and nodes to compute the activation values. The process explores G from the starting nodes and edge weights are taken into account. This process is iterative.
- According to whether C specifies the expected type of results, the spreading process can be categorized into concept constrained search (CCS) and none-concept constrained search (NCCS). Constraints, e.g., activation threshold constraint, distance constraint, concept constraint, etc., are applied to prevent the spreading process propagating across the entire data instance graph.
- After the finish of the spreading process, the nodes processed are ordered non-increasingly according to their activation values, thus generating the final ranked semantic search results.

Next, we illustrate the above procedures. The activation starts with an initial set of starting nodes from which the process explores G . This can be computed, e.g., as in the Google search engine (Brin & Page, 1998; Richardson & Domingos, 2002) in which the final ranking score of a web page depends on not only the global PageRank value but also the relevance of the page to a query. We combine the global importance and the query-dependent relevance to determine the initial activation values. The resulting activation value a_i of node $v_i \in V$ to the query R is as follows:

$$a_i = \lambda \frac{g(v_i)}{\max_{v_j \in V} g(v_j)} + (1 - \lambda) Rel(R, v_i) \quad (\lambda \in [0, 1]) \quad (6)$$

In Eq. (6), $g(v_i)$ is the global ranking value of node v_i and $Rel(R, v_i)$ is a relevance measure of node v_i to R which is in the range $[0, 1]$. The ranking value $g(v_i)$ is normalized in the range $[0, 1]$ divided by the maximum ranking values for all nodes in the graph. The parameter λ (default 0.5) is a weight factor between $g(v_i)$ and $Rel(R, v_i)$. The choice of relevance measure $Rel(R, v_i)$ is arbitrary. For example, let a search item be $t_i : k_i \in Q$. If the property t_i is *has-author* or *presentedin*, the relevance measure could simply be $Rel(R, v_i) = 1$ if k_i appears in v_i , and 0 otherwise. If t_i is *has-fulltext*, more complex measurement could be used, e.g., TFIDF, cosine similarity, proximity distance. The nodes with higher activation values, e.g., setting a threshold value or a total number limit, are placed into the set of starting nodes.

The starting nodes are inserted into a non-increasing priority queue according to the activation values. During the propagation, the nodes in the queue are topped and other nodes are inferred along different types of edges. The activated nodes which are not in the priority queue are inserted and topped again. This process iterates until termination conditions are achieved, or the priority queue is empty. After this process, the nodes processed are ordered non-increasingly according to their activation values, and then search results are generated.

We discuss the output function of activation in the propagation process. During the activation, the activation threshold and distance constraints are used to prevent propagating across the entire graph. The output function of node v_i is formulated as follows:

$$a'_i = \begin{cases} (1 - d \cdot \theta) I_i, & \text{where } I_i = \sum_{j \in \text{nodes connected to } i} w_{ji} \cdot a_j \text{ if } I_i \geq \text{Threshold} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

In Eq. (7), d is the propagation distance from the initial set of nodes, θ is the decay factor which decreases the activation value during every level of propagation, I_i is the input value of v_i and w_{ji} is the weight of the edge from nodes j to i , and *Threshold* is the defined input threshold below which the activation would not propagate from the node. The reason for setting the decay factor θ (default 0.3) is that the strength of association between nodes decreases with increasing propagation distance. It is important to restrict the maximum propagation distance d in the spreading process. As Crestani (1997) indicates, setting the maximum value as three steps is usually enough, however this is application specific. In our model, we set the default maximum value as 2. To compute the input value I_i , edge weights which reflect relationship strengths of connected nodes, are taken into account. Relationships with higher edge weights imply more importance to the activation process. The assignment of edge weights is formulated by Eq. (2).

4.2.1. Concept constrained search

For the query $R = \{Q; C\}$, if C explicitly specifies the expected type of search results, only resources of the given concept type C are allowed to be returned, e.g., $C = \text{"Publication"}$ or "Author" . This search is similar to SQL-like declarative languages which provide RDF-based precise answers. For example, the query, finding publications written by the author "Hai Jin", is denoted by $R = \{\text{has-author: "Hai Jin"; Publication}\}$. The results could be given by the following RDQL query:

```
SELECT    ?pub
WHERE     (?pub <rdf:type> <rss:Publication>)
          (?pub <rss:has-author> ?author)
AND       ?author = 'Hai Jin'
USING    rdf FOR <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
          rss FOR <http://grid.hust.edu.cn/rss#>
```

However, the RDQL technique does not provide any ranking strategies to order the search results. In the general search process, the search strategy could be naturally extended to support concept constraint and resources ranking. For concept constraint, only nodes of the given concept type are allowed to be placed into the output priority queue during the activation. In addition, the global importance of resources and the relevance are taken into account, thus producing ranked results with the given concept type.

4.2.2. None-concept constrained search

If C in the query R is set to *null*, any type of search results could be returned. This search is a typical semantic search (Guha et al., 2003) since no particular concept types are specified. The semantic relationship between the query and returned resources may be explicit or implicit. For example, the query, finding most related resources about the author "Hai Jin", is represented as $R = \{\text{has-author: "Hai Jin"; null}\}$. The resources which are explicitly connected to the author, e.g., important publications he wrote, conferences he often presented in, are reasonable to be returned. It is also useful to return those resources which are implicitly semantically related to the author, e.g., authors he has co-authored with, authors who may have similar research interests to him since they usually presented in the same conferences, publications through the chain of citation relationships, etc. At present, there are no appropriate techniques to directly support this type of search as we do.

However, this type of search imposes the problem that any two resources in the graph might be semantically related since the small-world phenomenon (Kleinberg, 2000). It is necessary to give metrics to measure the degree of semantic relationships among resources and determine the importance of returned resources. Those results with semantic relationships below a predefined threshold may be discarded.

Finally, we briefly discuss the generation of results since most of this work has been implied during the search process. The answer results $A = \text{Results}(R) \subseteq V$ are a sequence of resources which are ordered according to the semantic relationships with the query R . The results are generated after the search process. The resources processed in the search process are ordered non-increasingly according to their activation values. For *CCS*, the results A would be a sequence of resources with the same type C . In *NCCS*, A could be a mixture of ordered resources with any kinds of concept types. In particular, the former can be taken as a special case of

the latter. For example, to better support results display and browsing, we can categorize the mixed resources of A in the latter into several separate sequences according to different concept types they belong to. Hence, each sequence would consist of resources with the same concept type.

5. Experiments

In this section we experimentally evaluate the ranked semantic search framework. The goals of the experiments are: (1) to validate the feasibility of global ranking on resources; (2) to compare the quality of our ranking mechanism with that of the PageRank in the data graph; and (3) to demonstrate the effect of the ranked semantic search framework.

5.1. Datasets and experimental setup

The datasets are real-word data collected from CiteSeer metadata (<http://citeseer.ist.psu.edu/oai.html>), including the CiteSeer BibTex records and metadata archive `oai_citeseer.tar.gz` which is compliant with the Dublin Core standard with additional metadata fields (e.g., abstract, citation relationships, author affiliations, and author addresses). The metadata covers literatures in the field of computer science and computer technology with about 800,000 publications and totals approximately 2 GB.

To verify the feasibility and efficiency of our ranked semantic search framework, we mainly choose those data in the database domain. We use keywords “data”, “database”, “SIGMOD”, “VLDB”, “ICDE”, “ICDT”, “TKDE”, “TODS” to match the title or the booktitle or the journal fields of BibTex entries. Most publications within the database area are filtered and parts of publications on other research areas, such as artificial intelligence, data mining, information retrieval, are also included in the datasets. For the above filtering procedure, some BibTex entries are discarded for errors, e.g., url or authors missing. Next we combine the filtered BibTex entries with the metadata archive and generate detail information on publications, including unique id, title, author, publishing location, abstract and citation relationship. The information is then encoded into a RDF graph using the Sesame storage server. In this graph, there are totally 27,488 distinct instances and 199,517 relationships. These data instances belong to four different kinds of Class, i.e., Author, Publication, Conference and Journal. These instances are inter-connected through heterogeneous relationships, such as *has-author* and its inverse *has-written*, *cited-by* and *cite*, *presented in* and *contain*, *published in* and *published*. In addition, there are relationships connecting instances to literals, e.g., *has-title*, *conference-name*, *journal-name*, etc. To simplify the problem, all authors of a paper are regarded as having the same importance to this paper in the experiments, which might not be always true. These schema relationships are assigned different weights as in Fig. 1. For simplicity, we do not apply the relationship *similar_to* which could be implemented through VSM or the document cosine similarity measurement. The weight assignment of the relationship *published in/published* is the same as that of *presented in/contain*.

5.2. Results

Before illustrating the experimental results, we should declare that our ranking results in this paper DO NOT imply who are the most important researchers or which papers are the most valuable literatures or which journals/conferences should be No. 1 in the database research area at all. The experimental results just indicate the importance of resources and the relevance to the query in the data instance graph based on the CiteSeer metadata in which many database researchers and publications are absent because they could not be crawled on the web by the CiteSeer search engine.

The experiments are divided into two classes. Firstly, we verify the quality of the ranking mechanism. Since there are no standard metrics to measure the quality of ranking ontologies or instances in the semantic web at present, we mainly compare the quality of the mechanism with that of the PageRank through analyzing the global ranking results. Secondly, we present the ranked search results to demonstrate the effect of the ranked semantic search, especially the none-concept constrained search. Also, the results are compared to that of the PageRank which takes global PageRanks values and the query-dependent relevance measure.

5.2.1. Quality of global ranking

Table 1 illustrates the global ranking results using RSS ranking mechanism which is independent of queries. We replace the URIs of instances with their readable comments in Table 1. The *Class* field indicates concept types of instances and the *Ranking Value* field gives global ranking values of instances. From this table, we can see that the conference “The SIGMOD Conference” is ranked 1st with the ranking value 27.064753. This is due to the fact that the conference contains many publications and these publications are cited by others frequently. The journal “IEEE Transactions on Knowledge and Data Engineering” is ranked 2nd. We notice that the 3rd position is “Benchmarking Database Systems: A Systematic Approach” with the high ranking value 17.808655. This is because three famous database researchers: “Dina Bitton”, “David J. Dewitt”, and “Carolyn Turbyfill”, wrote this publication. The author “David J. Dewitt” is ranked 16th and is ranked 2nd among all authors in the datasets. For the following authors ranking results in Table 4, we will see that all three authors are of Top-10. In addition, this publication is cited by many other publications. From the Google Scholar (<http://scholar.google.com>), the publication is cited 247 times. The Top-1 ranked author among all authors in the datasets is “Rakesh Agrawal”. From his homepage (<http://rakesh.agrawal-family.com/>), we can find that “he is the recipient of the ACM-SIGKDD First Innovation Award, ACM-SIGMOD Edgar F. Codd Innovations Award, ACM-SIGMOD Test of Time Award, VLDB 10-Yr Most Influential Paper Award”. Moreover, he has written the 1st as well as 2nd highest cited of all papers in the fields of databases and data mining. As can be seen from Table 1, the results of our ranking mechanism well reflect the global importance of instances.

Table 2 shows the ranking results of the PageRank model. The two top ranked instances are the same as those of Table 1. However, we can find there are some important differences between Tables 1 and 2. First, our ranking model pays more attention on publications since the edge weight of the relationship *cite* between publications is a high value (0.6 in this experiment). There are 8 publications in Table 1 while there are only 4 publications in Table 2. Second, the ranking results of Table 1 more concentrate on the database domain than those of Table 2. In Table 2, the conferences “Workshop on Algorithms and Data Structures” and “Storage and Retrieval for Image and Video” are ranked 4th and 9th, respectively, since they contain many publications or authors on the keyword “data”. However, it might be more appropriate to place the two conferences into the algorithms and theory or the multimedia applications research area. Third, several conferences in Table 2, e.g., “Knowledge Representation Meets Databases”, “IFIP Workshop on Database Security”, are highly ranked because these conferences contain many publications and authors in the datasets.

Table 1
Global ranking results using RSS ranking model

Rank	Instance (instance comment)	Class	Ranking value
1	The SIGMOD Conference	Conference	27.064753
2	IEEE Transactions on Knowledge and Data Engineering	Journal	19.142874
3	Benchmarking Database Systems: A Systematic Approach	Publication	17.808655
4	Knowledge Discovery and Data Mining	Conference	16.527643
5	Mining Association Rules between Sets of Items in Large Databases	Publication	14.375960
6	An Interval Classifier for Database Mining Applications	Publication	9.990716
7	The Object-Oriented Database System Manifesto	Publication	9.925074
8	Fast Algorithms for Mining Association Rules	Publication	9.001670
9	Deriving Production Rules for Constraint Maintenance	Publication	7.653728
10	VLDB Journal: Very Large Data Bases	Journal	7.047528
11	ACM Transactions on Database Systems	Journal	7.034485
12	Efficient Similarity Search In Sequence Databases	Publication	6.223272
13	Symposium on Principles of Database Systems	Conference	6.096932
14	Data Cube: A Relational Aggregation Operator Generalizing	Publication	5.873787
15	Rakesh Agrawal	Author	5.818371
16	David J. Dewitt	Author	5.683524
17	Performing Group-By Before Join	Publication	5.079914
18	The R-Tree: A Dynamic Index for Multi-Dimensional Objects	Publication	5.020757
19	Methods and Tools for Equivalent Data Model Mapping Construction	Publication	4.670882
20	SIGMOD Record	Journal	4.667948

Table 2
Global ranking results using PageRank model

Rank	Instance (instance comment)	Class	Ranking value
1	The SIGMOD Conference	Conference	27.697193
2	IEEE Transactions on Knowledge and Data Engineering	Journal	22.710367
3	Knowledge Discovery and Data Mining	Conference	21.498678
4	Workshop on Algorithms and Data Structures	Conference	13.680820
5	Mining Association Rules between Sets of Items in Large Databases	Publication	13.146843
6	VLDB Journal: Very Large Data Bases	Journal	9.558816
7	Knowledge Representation Meets Databases	Conference	9.558816
8	Rakesh Agrawal	Author	9.196095
9	Storage and Retrieval for Image and Video	Conference	9.025708
10	IFIP Workshop on Database Security	Conference	8.942515
11	An Interval Classifier for Database Mining Applications	Publication	8.446479
12	Benchmarking Database Systems: A Systematic Approach	Publication	8.200334
13	ACM Transactions on Database Systems	Journal	7.651512
14	Fast Algorithms for Mining Association Rules	Publication	7.205092
15	Extending Database Technology	Conference	7.015524
16	Statistical and Scientific Database Management	Conference	6.889505
17	Data Cube: A Relational Aggregation Operator Generalizing	Publication	6.841758
18	The Object-Oriented Database System Manifesto	Publication	6.108785
19	Deriving Production Rules for Constraint Maintenance	Publication	5.869477
20	Symposium on Principles of Database Systems	Conference	5.822423

Table 3 gives the journals ranking results using RSS ranking model. These results are selected from the global ranking results with the concept type “Journal”.

Table 4 gives ranking results of authors using RSS ranking model. We especially point out the ranked 5th author “Dina Bitton”. He has just a single publication in the datasets. However, the publication is “Benchmarking Database Systems: A Systematic Approach” which is ranked 1st among all publications (see Table 1). In addition, his co-authors of the publication are “David J. Dewitt” and “Carolyn Turbyfill” and they are highly ranked 2nd and 4th among all authors, respectively. However, he is just ranked 21st among all authors in the PageRank model.

From the above results, we can conclude that our global ranking model is feasible and performs well. The model better measures the global importance of resources than the PageRank model w.r.t. the selected database application domain.

5.2.2. Effect of changing the edge weights

The edge weights of relationships have influence on the global ranking results. We should evaluate the effect of changing edge weights to compare the distance between different ranking lists. One popular distance measure between two full ranking lists is the *Kendall's τ* (Dwork, Kumar, Naor, & Sivakumar, 2001) which counts

Table 3
Journals ranking results using RSS ranking model

Rank	Journal name	Ranking value
1	IEEE Transactions on Knowledge and Data Engineering	19.142874
2	VLDB Journal: Very Large Data Bases	7.047528
3	ACM Transactions on Database Systems	7.034485
4	SIGMOD Record	4.667948
5	Distributed and Parallel Databases	3.100313
6	Journal of Data Mining and Knowledge Discovery	1.805155
7	IEEE Data Engineering Bulletin	1.308101
8	IEEE Quarterly Bulletin on Data Engineering	1.084234
9	Computational Statistics and Data Analysis	1.067780
10	EDI Forum: The Journal of Electronic Data Interchange	0.686978

Table 4
Authors ranking results using RSS ranking model

Rank	Author name	Ranking value
1	Rakesh Agrawal	5.818371
2	David J. Dewitt	5.683524
3	Christos Faloutsos	3.601633
4	Carolyn Turbyfill	3.513857
5	Dina Bitton	3.513857
6	Jennifer Widom	3.361753
7	Michael Stonebraker	3.267235
8	Hector Garcia-Molina	2.431980
9	H.V. Jagadish	2.415681
10	Ramakrishnan Srikant	2.275021

the number of pairwise disagreements between two lists. We use a variant of *Kendall's* τ as similarity measurement which measures the degree to which the relative orderings of the top k entries of two ranking lists are in agreement.

Given two top k lists τ_1 and τ_2 , let U be the union set of distinct elements in τ_1 or τ_2 . Let τ'_1 contains $U - \tau_1$ which appears after all the elements in τ_1 . The τ'_2 is defined similarly. Then we define the similarity measure as follows:

$$S(\tau_1, \tau_2) = \frac{|(u, v) : \tau'_1(u) < \tau'_1(v), \tau'_2(u) < \tau'_2(v)|}{(|U - 1|)|U|/2} \quad (8)$$

In Eq. (8), $S(\tau_1, \tau_2)$ is in the range $[0, 1]$, and the larger value indicates τ_1 is more similar to τ_2 . For $u, v \in U$, if $\tau'_1(u) < \tau'_1(v)$, we say u is ahead of v in τ'_1 . The same holds for $\tau'_2(u) < \tau'_2(v)$. We set the edge weight of relationship *cite* ranging from 0.1 to 0.7, to compute the similarity measures between their ranking lists with that of setting the default weight 0.6 (see Section 3.2, $w(T = cite) = 0.6$), respectively. The similarity measure with the PageRank is also computed. The length of ranking lists is limited to top $k = 1000$.

Fig. 3 shows the similarity measure results. For comparison, we also place the similarity measure of the PageRank model though in which the assignment of edge weights is not needed in Fig. 3. From the figure, we can see that the similarity measure has the largest value 1 when the edge weight is set to 0.6 which is certain since the top k lists $\tau_1 \equiv \tau_2$. The similarity value between the ranking list of the PageRank model and that of setting $w(T = cite) = 0.6$ is 0.44 which is close to that (i.e., 0.46) of setting the weight 0.2. The reason is that when setting $w(T = cite) = 0.2$ in Fig. 1, the edge weights of relationships are almost equal which is similar to the PageRank. Fig. 3 indicates that the appropriate assignment of edge weights is important for the global ranking results and very high or low edge weights might cause the RSS ranking mechanism degenerate to the standard PageRank model.

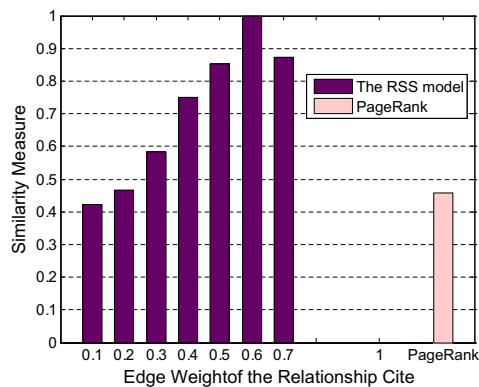


Fig. 3. Similarity measure between different ranking lists.

5.2.3. Effect of ranked semantic search

In this subsection, the query-dependent ranked semantic search results are presented, especially the none-concept constrained search since it is a typical semantic search (Guha et al., 2003). We also compare the effect of our method with that of the PageRank.

Table 5 shows the top results for the query $R = \{\text{"has-title":\text{"spatial"}; \text{null}\}$ using the RSS search method. The query is to find resources which are most semantically related to the keyword “spatial” via the property “has-title”. The *Dist.* field gives the propagation distance from the initial set of nodes. The results with the *Dist.* value 0 are the initial activation nodes. We can see that the final search results are a mixture of different classes. The 3rd ranked instance is the publication “The R-Tree: A Dynamic Index for Multi-Dimensional Objects” which title does not contains the keyword “spatial”. This paper, written by Christos Faloutsos (see Table 4) in 1987, is an important publication on spatial database and has been cited 754 times (according to Google Scholar). The most highly ranked author on spatial database is “Jiawei Han”. He is fruitful on the database and data mining research area, especially data mining in spatial database.

Table 6 gives the search results of the PageRank for the same query. The results are publications which title contain the keyword “spatial” and are ordered none-increasingly according to the PageRank values. From the results of Table 6, we find some important publications on spatial database, however, we could not obtain much more useful information on this research area as our search method does.

Table 7 gives the top search results for the query $R = \{\text{"has-author":\text{"N.H. Gehani"}; \text{null}\}$ using the RSS search method. The query is to find resources which are most semantically related to the author “N.H. Gehani” via the property “has-author”. The results with the *Dist.* value 0 are those publications written by him. The 9th is ranked instance is the author “H.V. Jagadish” who co-authored with him for many times. Table 8 gives the search results of the PageRank which have only seven entries. These results are all publications written by “N.H. Gehani”. *Recall* and *Precision* are usually used to evaluate the quality of retrieval systems. At present, there are no standard metrics to evaluate the quality of semantic search on the semantic web, there-

Table 5
Top-10 results for the query $\{\text{"has-title":\text{"spatial"}; \text{null}\}$ using RSS

Rank	Results (instance comment)	Class	Distance
1	Efficient and Effective Clustering Methods for Spatial Data Mining	Publication	0
2	The VLDB Journal	Journal	1
3	The R-Tree: A Dynamic Index for Multi-Dimensional Objects	Publication	2
4	Symposium on Large Spatial Databases	Conference	1
5	Jiawei Han	Author	1
6	Spatial Databases – Accomplishments and Research Needs	Publication	0
7	Efficient Processing of Spatial Queries in Line Segment Databases	Publication	0
8	Raymond T. Ng	Author	1
9	A Storage and Access Architecture for Efficient Query Processing in Spatial Database Systems	Publication	0
10	Knowledge Discovery in Large Spatial Databases: Focusing Techniques for Efficient Class Identification	Publication	0

Table 6
Top-10 results for the query $\{\text{"has-title":\text{"spatial"}; \text{null}\}$ using PageRank search

Rank	Results (instance comment)	Class
1	Spatial Databases – Accomplishments and Research Needs	Publication
2	Qualitative Representation of Spatial Knowledge in Two-Dimensional Space	Publication
3	k -Nearest Neighbor Classification on Spatial Data Streams Using P-trees	Publication
4	Spatial Queries on a Hierarchical Terrain Model	Publication
5	Spatial Join Processing Using Corner Transformation	Publication
6	Generalized Relational Algebra: Modeling Spatial Queries in Constraint Databases	Publication
7	Spatial Joins Using R-trees: Breadth-First Traversal with Global Optimizations	Publication
8	DEDALE, A Spatial Constraint Database	Publication
9	Calibrating the Meanings of Spatial Predicates from Natural Language: Line-Region Relations	Publication
10	Optimal Redundancy in Spatial Database Systems	Publication

Table 7
Top-10 results for the query {"has-author": "N.H. Gehani"; null} using RSS

Rank	Results (instance comment)	Class	Distance
1	ODE as an Active Database: Constraints and Triggers	Publication	0
2	Composite Event Specification in Active Databases: Model and Implementation	Publication	0
3	Event Specification in an Active Object-Oriented Database	Publication	0
4	N.H. Gehani	Author	1
5	Rationale for the Design in the Database Programming Language O++	Publication	0
6	Object Versioning in ODE	Publication	0
7	Efficient Processing of Spatial Queries in Line Segment Databases	Publication	0
8	Integrity Maintenance in Object-Oriented Databases	Publication	1
9	H.V. Jagadish	Author	2
10	The VLDB Conference	Conference	1

fore, the metrics *Recall* and *Precision* in IR might not be appropriate for the evaluation. However, from Tables 7 and 8, we can still find that the RSS model can retrieve more semantically related answers than the PageRank search method, thus increasing the *recall*. There may be difficulties for comparing the *precision*. The main reason is that RSS retrieves entities most semantically related to queries while PageRank is based on keyword matching. For example, for the query $R = \{\text{"has-title": "spatial"; null}\}$ (see Tables 5 and 6), the publication "The R-Tree: A Dynamic Index for Multi-Dimensional Objects" and the author "Jiawei Han" are not retrieved by the PageRank model, though the two results are very relevant to this query. Therefore, we resorted to user participation.

We choose 10 test queries which were all on the database subject, and separately sent them to our RSS search model and the PageRank search model. The two models returned result lists (totally 20). Then we invited 10 students who are PhD or master candidates. Among them, 7 students are from our lab and the others who major in spatial databases are from Wuhan University. Each student rated the top-15 (occasionally, less than 15) results of each returned list using scores from 0 to 1. The scoring of relevance was regulated as follows: 0 for irrelevant result, 0.3 for a slightly relevant one, 0.6 for a fairly relevant one, and 1 for a highly relevant one. We averaged their scores and evaluated the effectiveness of two models using the scored precision. The scored precision for the result list τ_i is defined:

$$sp(\tau_i) = \frac{\sum_{s=1}^{10} \sum_{j=1}^k score(\tau_i, s, j)}{10 * k} \quad (9)$$

In Eq. (9), $score(\tau_i, s, j)$ denotes the score that the s th student marked for the j th entry of the list τ_i in which only the top k ($k \leq 15$) entries were selected. For each result list τ_i , we calculated $sp(\tau_i)$ which denotes the average score that 10 students marked the top k entries of τ_i . The result we obtained is shown in Fig. 4.

For RSS and PageRank, the average scored precisions over all 10 test queries are 0.90 and 0.71, respectively. The above user evaluation is simple and subjective, however, the results can still show our RSS search model outperforms the PageRank model approximately 0.19 w.r.t. the average scored precision.

From the experimental results, we can draw the conclusion that our ranked semantic search framework is effective and can produce favorable ordering quality of semantic search results.

Table 8
All results for the query {"has-author": "N.H. Gehani"; null} using PageRank search

Rank	Results (instance comment)	Class
1	CQL++: An SQL for a C++-based Object-Oriented DBMS	Publication
2	Event Specification in an Active Object-Oriented Database	Publication
3	Composite Event Specification in Active Databases: Model and Implementation	Publication
4	Object Versioning in ODE	Publication
5	Rationale for the Design in the Database Programming Language O++	Publication
6	ODE as an Active Database: Constraints and Triggers	Publication
7	Temporal Queries for Active Database Support	Publication

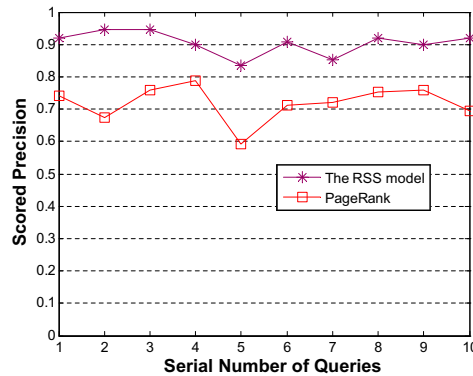


Fig. 4. Comparison of the scored precision.

6. Conclusions and future work

The semantic web is not only about common formats for interchange of data but also about relationships among data which sharply distinguishes from the web. In addition, many current methods of accessing the semantic web data usually use a SQL-like syntax to retrieve exactly matching results in disorder. Therefore, specialized search mechanisms to support inference and properly order search results are important for the success of the semantic web. In this paper we introduce the framework RSS which implements ranked semantic search on the semantic web. In this framework, a novel ranking algorithm is proposed to measure the global importance of resources in the data graph which takes relationships analysis and the edge weights account. In addition, the search results can be greatly expanded with entities which are most semantically related to the query through an extended spreading activation process, thus supporting semantic search and providing uses with properly ordered search results in terms of a combination of global ranking values and the relevance between the resources and the query. The experimental results show that the framework is feasible and leads to favorable semantic search results.

For our future work, we will try to automatically assign the edge weights in the schema graph using machine learning methods and the relevance feedback mechanism. In addition, the datasets in the experiments mainly focus on scientific research domain. We are currently trying to evaluate our model in other application areas though we believe the framework is applicable to broader domains.

Acknowledgement

This paper is supported by the National 973 Key Basic Research Program under Grant No. 2003CB317003, and the Cultivation Fund of the Key Scientific and Technical Innovation Project, Ministry of Education of China under Grant No. 705034.

We are grateful to anonymous reviewers for their useful comments and suggestions which contribute to substantially improving this paper.

References

- Agarwal, S., Branson, K., & Belongie, S. (2006). Higher order learning with graphs. In *Proceedings of the 23rd international conference on machine learning (ICML)*.
- Alani, H., Brewster, C., & Shadbolt, N. (2006). Ranking ontologies with AKTiveRank. In *Proceedings of the fifth international semantic web conference (ISWC)*.
- Anyanwu, K., Maduko, A., & Sheth, A. (2005). SemRank: ranking complex semantic relationship search results on the semantic web. In *Proceedings of the 14th international conference on world wide web (WWW), Chiba, Japan* (pp. 117–127).
- Balmin, A., Hristidis, V., & Papakonstantinou, Y. (2004). ObjectRank: authority-based keyword search in databases. In *Proceedings of the 30th international conference on very large data bases (VLDB), Toronto, Canada* (pp. 564–575).
- Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. *Scientific American*, 284(5), 34–43.

- Berry, M., Drmac, Z., & Jessup, E. (1999). Matrices, vector spaces, and information retrieval. *SIAM Review*, 41(2), 335–362.
- Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the seventh international conference on world wide web (WWW)*, Brisbane, Australia (pp. 107–117).
- Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., et al. (2005). Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on machine learning (ICML)*, Bonn, Germany (pp. 89–96).
- Castells, P., Fernandez, M., & Vallet, D. (2007). An adaptation of the vector-space model for ontology-based information retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 19(2), 261–272.
- Cohen, P., & Kjeldsen, R. (1987). Information retrieval by constrained spreading activation on semantic networks. *Information Processing and Management*, 23(4), 255–268.
- Cohen, S., Mamou, J., Kanza, Y., & Sagiv, Y. (2003). XSEarch: a semantic search engine for XML. In *Proceedings of the 30th international conference on very large data bases (VLDB)*, Berlin, Germany (pp. 45–56).
- Crestani, F. (1997). Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, 11(6), 453–482.
- Deerwester, M. S., Dumais, S., Furnas, G., Landauer, T., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6), 391–407.
- Diligenti, M., Gori, M., & Maggini, M. (2005). Learning web page scores by error back-propagation. In *Proceedings of the 9th international joint conference on artificial intelligence (IJCAI)*, Edinburgh, Scotland (pp. 684–689).
- Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., et al. (2004). Swoogle: a search and metadata engine for the semantic web. In *Proceedings of the 13th ACM conference on information and knowledge management (CIKM)*, Washington DC, USA (pp. 652–659).
- Dwork, C., Kumar, R., Naor, M., & Sivakumar, D. (2001). Rank aggregation methods for the web. In *Proceedings of the 10th international conference on world wide web (WWW)*, Hong Kong (pp. 613–622).
- Guha, R., McCool, R., & Miller, E. (2003). Semantic search. In *Proceedings of the 12th international conference on world wide web (WWW)*, Budapest, Hungary (pp. 700–709).
- Guo, L., Shao, F., Botev, C., & Shanmugasundaram, J. (2003). XRank: ranked keyword search over XML documents. In *Proceedings of the 22th ACM SIGMOD international conference on management of data (SIGMOD)*, California, USA (pp. 16–27).
- Hayes, P. (2004). RDF semantics. <<http://www.w3.org/TR/rdf-mt/>>.
- Herman, I. (2006). Semantic web. <<http://www.w3.org/2001/sw/>>.
- Kiryakov, A., Popov, B., Terziev, I., Manov, D., & Ognyanoff, D. (2004). Semantic annotation, indexing, and retrieval. *Journal of Web Semantics*, 2(1), 49–79.
- Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5), 604–632.
- Kleinberg, J. (2000). The small-world phenomenon: an algorithm perspective. In *Proceedings of the 32nd ACM symposium on theory of computing (STOC)*, New York, USA (pp. 163–170).
- Kopena, J., & Regli, W. (2003). DAMLJessKB: a tool for reasoning with the semantic web. *IEEE Intelligent System*, 18(3), 74–77.
- Langville, A. N., & Meyer, C. D. (2005). A survey of eigenvector methods of web information retrieval. *SIAM Review*, 47(1), 135–161.
- Lehmann, F. (1992). Semantic networks. *Journal of Computers and Mathematics with Applications*, 23(22), 1–50.
- Mayfield, J., & Finin, T. (2003). Information retrieval on the semantic web: integrating inference and retrieval. In *Proceedings of the workshop on the semantic web at the 26th international ACM SIGIR conference on research and development in information retrieval*, Toronto, Canada.
- Nielsen, J. (1990). The art of navigating through hypertext. *Communications of the ACM*, 33(3), 296–310.
- Patel-Schneider, P., & Horrocks, I. (2006). Mapping to RDF graphs for OWL. <<http://www.w3.org/TR/owl-semantics/mapping.html>>.
- Popov, B., Kiryakov, A., Ognyanoff, D., Manov, D., & Kirilov, A. (2004). KIM-a semantic platform for information extraction and retrieval. *Journal of Natural Language Engineering*, 10(3–4), 375–392.
- Richardson, M., & Domingos, P. (2002). The intelligent surfer: probabilistic combination of link and content information in PageRank. *Advances in Neural Information Processing Systems*, 14, 1441–1448.
- Rocha, C., Schwabe, D., & Aragao, M. (2004). A hybrid approach for searching in the semantic web. In *Proceedings of the 13rd international conference on world wide web (WWW)*, New York, USA (pp. 374–383).
- Rumelhart, D., & Norman, D. (1983). Representation in memory. Technical Report, Department of Psychology and Institute of Cognitive Science, UCSD La Jolla, USA.
- Salton, G., & Buckley, C. (1988). On the use of spreading activation methods in automatic information retrieval. In *Proceedings of the 11th ACM SIGIR international conference on research and development in information retrieval (SIGIR)*, Grenoble, France (pp. 147–160).
- Stojanovic, N., Studer, R., & Stojanovic, L. (2003). An approach for the ranking of query results in the semantic web. In *Proceedings of the 2nd international semantic web conference (ISWC)*, Florida (pp. 500–516).
- Vallet, D., Castells, P., Fernandez, M., Mylonas, P., & Avrithis, Y. (2007). Personalized content retrieval in context using ontological knowledge [Special issue on the convergence of knowledge engineering, semantics and signal processing in audiovisual information retrieval]. *IEEE Transactions on Circuits and Systems for Video Technology*.