AN IMPROVED TECHNIQUE FOR RANKING SEMANTIC ASSOCIATIONS

S Narayana[1], Dr. G P S Varma[2], and Dr. A Govardhan[3]

[1]Department of Computer Science and Engineering, Gudlavalleru Engineering College, AP, India

[2]Department of Information Technology, SRKR Engineering College, Bhimavaram, AP, India

[3]Department of Computer Science and Engineering, JNT University, Hyderabad, AP, India

ABSTRACT

The primary focus of the search techniques in the first generation of the Web is accessing relevant documents from the Web. Though it satisfies user requirements, but it is insufficient as the user sometimes wishes to access actionable information involving complex relationships between two given entities. Finding such complex relationships (also known as semantic associations) is especially useful in applications such as National Security, Pharmacy, Business Intelligence etc. Therefore the next frontier is discovering relevant semantic associations between two entities present in large semantic metadata repositories. Given two entities, there exist a huge number of semantic associations between two entities. Hence ranking of these associations is required in order to find more relevant associations. For this Aleman Meza et al. proposed a method involving six metrics viz. context, subsumption, rarity, popularity, association length and trust. To compute the overall rank of the associations this method computes context, subsumption, rarity and popularity values for each component of the association and for all the associations. However it is obvious that, many components appears repeatedly in many associations therefore it is not necessary to compute context, subsumption, rarity, popularity, and trust values of the components every time for each association rather the previously computed values may be used while computing the overall rank of the associations. This paper proposes a method to reuse the previously computed values using a hash data structure thus reduce the execution time. To demonstrate the effectiveness of the proposed method, experiments were conducted on SWETO ontology. Results show that the proposed method is more efficient than the other existing methods.

1. INTRODUCTION

In today's world, a rapid increase in the accumulation and addition of huge information is found enormously. With this rapid change it is essential and demanding to store and retrieve the relevant information from the web. While accessing of relevant documents is the major focus of the Information Retrieval Systems of the first generation Web, where as accessing the relevant entities and the relationships that exist among these entities is the primary goal of the next generation Web. The current World Wide Web has huge amount of data that is often unstructured and usually human understandable but not machine understandable. Further, the current Web infrastructure does not allow the identification of entities and their relationships. This has lead to the development of next generation Web called the Semantic Web [8]. The Semantic Web makes use of machine interpretable semantics to address this problem by providing machine support to the user. The traditional search engine based systems find the relevant documents based on the

given keywords or key phrases where as the Semantic Web uses machine interpretable semantics to access the relevant information. The main tools that are currently being used in the Semantic Web are ontologies based on RDF [9], RDFS [10], and OWL (Web Ontology Language) and its associated reasoners [15] [16].

Searching relationships among the entities like people, places and events from the Semantic Web is an essential component in the future. Many applications such as intelligence analysis, genetics and pharmaceutical research are concentrating more on complex relationships than simple direct relationships between entities. The ranking of documents has been a critical component of past search engines based systems and now ranking of complex relationships between entities is becoming an important component in today's Semantic Web analytics engines [1][7][18]. Upon building the recent work on specifying and discovering complex relationships in RDF data, a flexible ranking approach is presented which can be used to identify more interesting and relevant relationships in the Semantic Web [1][2]. To provide a different type of analysis based on semantic relationships, users are given potentially interesting complex relationships between entities, through a sequence of relationships between the metadata (annotations) of Web sources (or documents). These complex relationships between two entities are defined as semantic associations [7]. Possibly, these relationships are at the heart of semantics [14], lending meaning to information, making it understandable and actionable and providing new and possibly unexpected insights. Semantic associations constitute one of most important actionable knowledge. As an example Table 1 shows some semantic associations between two entities *Arnold Schwarzenegger* and *Jeb Bush*.

Table 1. Semantic Associations

S. No.	Semantic Association
1	**Arnold Schwarzenegger** -member of- **National Governors Association** -member of- **Mitt Romney** -member of- **Republican Party** -member of- **Jeb Bush**
2	**Arnold Schwarzenegger** -member of- **Republican Party** -member of- **Mel Martinez** -represents- **Florida** -represents- **Jeb Bush**
3	**Arnold Schwarzenegger** -spoke at- **2004 Republican National Convention** -nominated at- **George W. Bush** -relative of- **George H.W. Bush** -member of- **Republican Party** -member of- **Jeb Bush**
4	**Arnold Schwarzenegger** -invested in- **Planet Hollywood** -invested in- **Bruce Willis** -affiliated with- **Republican Party** -member of- **Jeb Bush**
5	**Arnold Schwarzenegger** -member of- **George H.W. Bush's Council of Physical Fitness** -affiliated with- **George H.W. Bush** -relative of- **George W. Bush** -member of- **Republican Party** -member of- **Jeb Bush**
6	**Arnold Schwarzenegger** -spoke at- **2004 Republican National Convention** -spoke at- **Laura Bush** -spouse of- **George W. Bush** -member of- **Republican Party** -member of- **Jeb Bush**
7	**Arnold Schwarzenegger** -member of- **George H.W. Bush's Council of Physical Fitness** -affiliated with- **George H.W. Bush** -member of- **Republican Party** -member of- **George W. Bush** -relative of- **Jeb Bush**

Given two entities, there exist a huge number of semantic associations between the entities. Therefore ranking of associations is required in order to get relevant associations. The Semantic Web not only consist resources but also consist heterogeneous relationships that exist between resources. With the size and complexity of ontologies growing rapidly, the number of semantic associations between a pair of entities is becoming increasingly overwhelming. Moreover these

associations pass through one or more intermediate entities. The resulting research benefits many areas of industry such as "e-activities", health care, privacy and security, knowledge management and information retrieval [16]. The development of large scale semantic metadata repositories such as SWETO [11], TAP [12], OpenCyc [13] etc. provide a platform to discover semantic associations between the entities. To discover relevant semantic associations, Aleman Meza et al. [1] [2] proposed a flexible ranking approach. This approach is based on six metrics viz. context, subsumption, popularity, rarity, trust, and association length. To compute the ranking score, this approach computes the values for context, subsumption, popularity and rarity for each component of the association every time it scans the association from the database. However, some components may occur repeatedly in two or more associations. For example, as shown in Table 1, the relationships 'member of', 'spoke at', 'affiliated with', 'relative of' and the entities 'Republican Party', 'George H.W. Bush', '2004 Republican National Convention', 'George H.W. Bush's Council of Physical Fitness' occurs repeatedly in many associations. Since, it is evident that many components of the associations occur repeatedly in many associations therefore reusing the values of the components which were already computed may reduce the execution time. This paper proposes an approach to reuse the previously computed values viz. context, subsumption, popularity, rarity and trust of the components while computing the ranking score of the associations thus reduce the execution time. The rest of the paper is organized as follows; Section 2 reviews related work, Section 3 describes the data model and basic definitions of semantic associations, Section 4 explains the proposed method, the experimental results are presented in Section 5 and Section 6 draws some conclusions and possible future work.

2. RELATED WORK

Several methods [1]-[7] have been proposed to discover and rank semantic associations. Anyanwu and Sheth et al. [7] propose a method to discover and rank semantic associations using ρ-operator. The ρ-operator performs a traversal in the knowledge base to determine whether or not an association is possible. If an association is possible then the authors used the notion of context to capture the relevant region(s) which contains potential paths. In addition to the context, user may assign ranks to important properties in the order of importance. This allows the display of relevant associations at the top.

Shahdad Shariatmadari et al. [6] propose a technique to find semantic associations using Semantic Similarity. Anyanwu et al. [4] proposed a method called *SemRank* to rank semantic associations. In this method, semantic associations are ranked based on their predictability. The rank model which it uses is a rich blend of both semantic and information-theoretic techniques with heuristics that support the search process. It provides a sliding bar using which, a user can easily vary the search mode from conventional search mode to discovery search mode. The relevancy of a semantic association is measured based on the information content of the association which is computed based on the occurrence of edge as an event and RDF properties as outcomes. In other word, it measures property's uniqueness with respect to the other properties in the knowledge base to decide association relevancy.

Aleman Meza et al.[1] [2] propose a method to rank semantic associations using six types of criteria called Subsumption(items that occur at lower level in the hierarchy convey more information than the items that occur at higher level), Path length(allows the user to select longer or shorter paths), Popularity(allows the user to prefer popular entities or unpopular entities based on number of incoming and outgoing edges of entities), Rarity(allows the user to prefer rarely occurring or commonly occurring associations), Trust(decides the reliability of the association based on its origin) and Context (allows the user to select concepts in an RDF graph to define his domain of interest. This method also ranks semantic associations using user preferences such as

favour rare or common associations, popular or unpopular associations and shorter or longer associations.

Lee M et al. [5] propose a method to rank semantic associations based on information theory and spreading activation to expand the semantic network. In this method, the results are provided based on relations between search keyword and other resources in a semantic network.
Viswanathan and Ilango et al. [3] propose a personalization approach for ranking semantic associations between two entities. They capture user's interest level in different domains based on their Web browsing history. The value of the user's interest level is stored in a table and based on these values the context weight of the associations is calculated and ranked.

To the best of my knowledge, all these approaches may not incorporate the reusability of previously computed values while computing the ranking score of semantic associations.

3. BACK GROUND
3.1. Data Model

On the Semantic Web, information is represented as a set of assertions called statements made up of three parts: subject, predicate, and object. The subject and the object of a statement is the resource that a statement describes, and the predicate describes a relationship between the subject and the object. The relationship is labelled with the name of the property and resource is labelled with the URI of the resource. A resource can be an entity or a literal. Object can be another resource or a literal. Assertions of this form a directed graph, with subjects and objects of each statement as nodes, and predicates as edges. This is the data model used by the Semantic Web, and it is formalized in the language called the Resource Description Framework (RDF) [9]. RDF is a World Wide Web Consortium (W3C) standard for describing Web resources (also called entities) by specifying how these resources are related with other resources (or class of entities). The class hierarchy of resources and property hierarchy are described in an RDF Schema (RDFS) [10] which acts the standard vocabulary for RDF. The Web Ontology Language (OWL) extends the RDFS vocabulary with additional features.

3.2. Semantic Associations

The complex relationships between two entities are known as semantic associations [7]. Semantic associations are meaningful and relevant complex relationships between the entities. They lend meaning to information, making it understandable and actionable, and provide new and possibly unexpected insights. Different entities can be related in multiple ways. For example, a Professor may be related to a University, students, courses, and publications; but s/he can also be related to other entities by different relations like *hobbies*, *religion*, *politics*, etc. Relationships that span several entities may be very important in domains such as National Security, because they may enable analysts to see the connections between seemingly disparate people, places and events. To define semantic associations, the formalism specified by Anyanwu et al. [7] is followed.

3.2.1. Defination1 (Semantic Connectivity)

Two entities e1 and en are semantically connected if there exists a sequence $e_1, P_1, e_2, P_2 \ldots e_{n-1}, P_{n-1}, e_n$ in an RDF graph where e_i ($1 \le i \le n$) are entities and P_j ($1 \le j \le n$) are properties. Figure 1 shows the semantic connectivity between e_i and e_n.
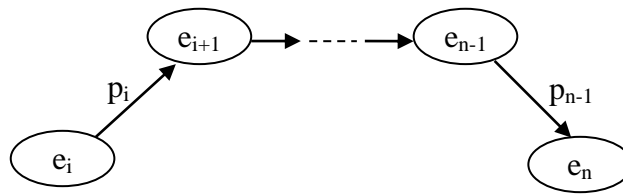
Figure 1. Semantic association between entities e_i and e_n

3.2.2. Defination2 (Semantic Similarity)

Two entities e_1 and f_1 are semantically similar if there exist two semantic paths e_1, P_1, e_2, P_2, …., e_{n-1}, P_{n-1}, e_n and f_1, Q_1, f_2, Q_2, …, f_{n-1}, Q_{n-1}, f_n semantically connecting e_1 with e_n and f_1 with f_n respectively, and that for every pair of properties P_i and Q_i, $1 \leq i \leq n$, either of the following conditions holds; $P_i = Q_i$ or P_i is *rdf:subPropertyOf* Q_i or Q_i is *rdf:subPropertyOf* P_i, then two paths originating at e_1 and f_1, respectively, are semantically similar.

3.2.3. Defination3 (Semantic Association)

Two entities e_x and e_y are semantically associated if e_x and e_y are semantically connected or semantically similar.

4. PROPOSED METHOD

This section describes the proposed approach to reuse the previously computed values of the components to compute the overall rank of each association. Before to this, it first explains the criteria proposed by Aleman Meza et al. to rank the semantic associations.

4.1. Ranking Semantic Associations
4.1.1. Context Weight Cs

Consider the scenario where user wishes to find semantic associations between two persons in the domain of *'Politics'*. Then concepts such as *'Politician'*, *'Political Organization'*, *'Government Organization'* and *'Legislation'* are considered to be more relevant whereas the concepts such as *'Financial Organization'* and *'Terrorist Organization'* are considered to be less relevant. So, user is provided facilities to define his context by selecting his interested regions from the ontology, and based on this context the associations are ranked. As an example consider the RDF graph shown in Figure 2. It shows that, the user has selected three regions belonging to *'Political Organization'*, *'Politician'*, and *'Legislation'*. It also shows that there are three associations viz. the top-most association (call it as S1), the middle association (call it as S2) and the bottom-most association (call it as S3). Since all the entities of S1 are belonging to the user selected regions so S1 should be ranked high. Similarly three entities of S2 are fit in the user selected regions so S2 should be ranked next and none of the entities of S3 belong to user selected regions, therefore S3 should be ranked lower.
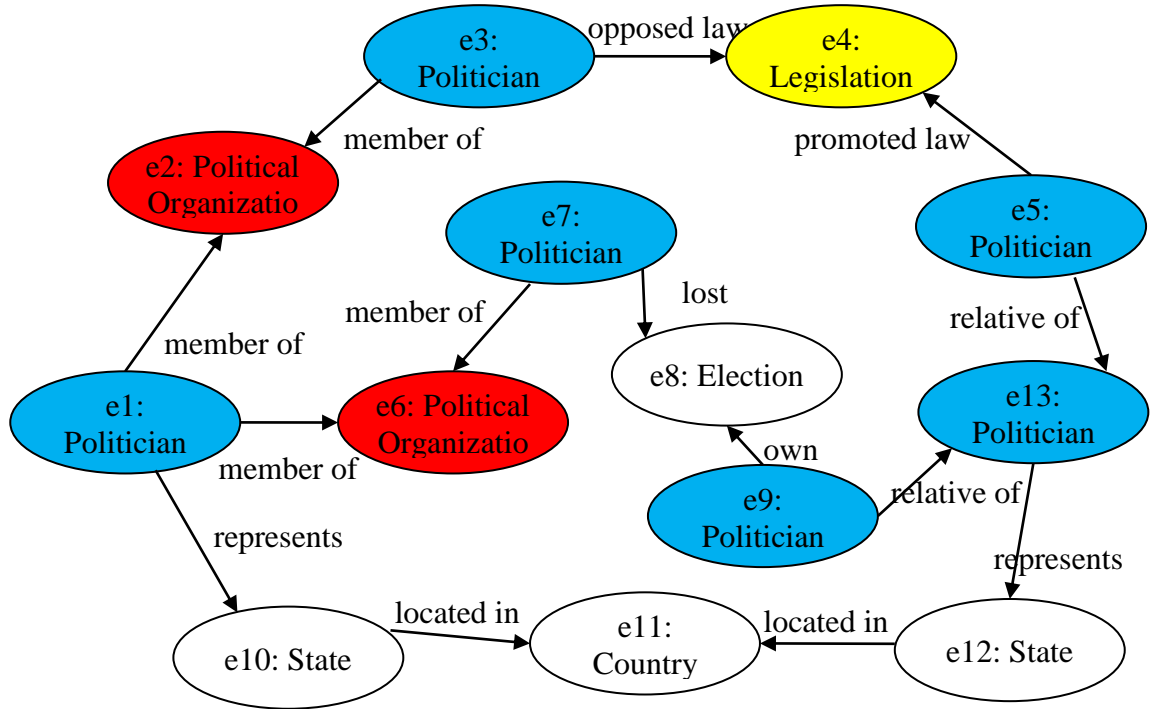
Figure 2. Part of an RDF graph showing semantic associations between two entities e1 and e13

Let S be a semantic association, c be a component of S which is either an entity or a relationship. Let *length(S)* denotes length of S which is the number of components present in S. Let R_i denote region i, which is collection of entities and relationships. Let X_i denotes the set of components belonging to region 'i' and is given as;

$$X_i = \{c | c \in Ci \wedge c \in S\} \tag{1}$$

Let Y_i denotes the set of components that does not belong to any of the regions and is given as;

$$Y_i = \{c | c \notin Ci (\forall i | 1 \le i \le n) \wedge c \in S\} \tag{2}$$

Where 'n' is the number of regions selected by the user.

Now the context weight C_S of semantic association, S, is defined as;

$$C_S = \frac{1}{length(S)} (\sum_{i=1}^{n} (W_{R_i} \times |X_i|)) \times (1 - \frac{|Y_i|}{length(S)}) \tag{3}$$

Where 'n' is the number of regions and W_{Ri} is the weight of ith region.

4.1.2. Subsumption Weight Ss

The RDF graph contains hierarchy of entities where entities that occur at lower level of hierarchy are considered to be more specialized entities and convey more useful information and the entities that occur at higher level of hierarchy are considered to be generalized entities and convey less useful information. Hence, Associations that consists specialized entities are more relevant. The subsumption weight of S is computed by computing the subsumption weights of its components. The subsumption weight of ith component of S is computed as;

$$comp_{sw_i} = \frac{H_{comp_i}}{H_{depth}} \tag{4}$$

Where H_{comp_i} is the position of the ith component in the hierarchy H (the root entity position is 1) and H_{depth} is the total depth of the hierarchy. Now the overall subsumption weight, S_S, of S is defined as;

$$S_S = \prod_{i=1}^{length(S)} comp_{sw_i} \tag{5}$$

4.1.3. Path Length Weight Ls

In some applications, user might be interested in finding shorter associations, yet in other applications he may wish to find longer associations. For example in money laundering, the money launderer intentionally performs several innocuous transitions to escape from law. So user can determine which association length influences most the relevancy of associations. The length weight, L_S, of S is computed as;

$$L_S = \frac{1}{length(S)} \tag{6a}$$

$$L_S = 1 - \frac{1}{length(S)} \tag{6b}$$

If the user wish to find shorter associations, then (6a) is used; otherwise (6b) is used.

4.1.4. Popularity Weight Ps

The number of incoming and outgoing edges of an entity is defined as its popularity. In some cases associations that contain many popular entities are considered to be more relevant. For example, in finding the associations between two persons in the domain of 'Scientific Publications', user is interested in popular associations since associations that contain 'Authors' with more citations are considered to be more relevant. Yet in some cases user wish to find less popular associations. For example, in finding the associations between two persons in the domain of 'Geographical Locations', user wish to find less popular associations because the entity of type 'Country' may contain many incoming and outgoing edges and convey less information. Hence, user has to select whether 'favour more popular associations or favour less popular associations' based on his interest.

The popularity weight of an association is computed in terms of the popularity weights of its entities. Entity popularity is defined as;

$$p_i = \frac{\left|pop_{e_i}\right|}{\max_{1 \leq j \leq n}\left(\left|pop_{e_j}\right|\right)} \tag{7}$$

Where *typeOf(e_i) = typeOf(e_j)*, 'n' is the total number of entities in the knowledge base, pop_{e_i} is the set of incoming and outgoing relationships of e_i, $max(pop_{e_j})$ denotes the size of the largest such set among all entities in the knowledge-base of the same class as e_i. Thus p_i captures the entity popularity of the entity e_i with respect to all other entities of the same type as e_i in the knowledge-base. Now the overall popularity weight, P_S, of S is defined as;

$$P_S = \frac{1}{n} \sum_{i=1}^{n} p_i \tag{8a}$$

$$P_S = 1 - \frac{1}{n} \sum_{i=1}^{n} p_i \tag{8b}$$

Where 'n' represents number of entities in S, and p_i represents the entity popularity of the ith entity in S. If user wish to find popular associations, then (8a) is used; on the other hand, if he wants less popular associations (8b) is used.

4.1.5. Rarity Weight Rs

In some applications, user might be interested in rarely occurring events and in other applications he might be interested in commonly occurring events. For example, in money laundering officials wish to find associations consisting commonly occurring events because money launderers' perform several common transactions to escape from law. So user is allowed to select 'favour rare or common associations according to his interest. The rarity weight of association S is computed

in terms of rarity weights of its components. Let K represent all entities and relationships present in the knowledge-base. The rarity weight rar_i, of a component c_i of S is computed as;

$$rar_i = \frac{|M| - |N|}{|N|} \tag{9}$$

Where $M = \{c \mid c \in K\}$ (all entities and relationships in K) and (10)
$$N = \{c_j \mid c_j \in K \wedge typeOf(c_j) = typeOf(c_i)\} \tag{11}$$

With a restriction that in case c_i and c_j are of type rdf: Property, then the subject and object of c_i and c_j must have the same rdf: Type. Thus rar_i represents the frequency of occurrence of component c_i with respect to the whole knowledge-base. Now the rarity weight, R_S, of association S is defined as;

$$R_S = \frac{1}{length(S)} \times \sum_{i=1}^{length(S)} rar_i \tag{12a}$$

$$R_S = 1 - \frac{1}{length(S)} \times \sum_{i=1}^{length(S)} rar_i \tag{12b}$$

If user wish to find rare associations (12a) is used; in contrast, if he wish to find common associations (12b) is used.

4.1.6. Trust Weight Ts

The entities and relationships in a semantic association come from different sources. Some sources may be more trusted and some sources may be less trusted. For example in India "The Hindu" is considered to be more trusted source. Thus trust value is assigned to components in an association based on the source from which it is coming. The trust weight of association, T_S, of S, is computed as;

$$T_S = \min(t_{c_i}) \tag{13}$$

Where t_{c_i} is the assigned trust value of the component c_i.
The overall ranking formula for ranking semantic association, S, is given as;

$$R_S = k_1 \times C_S + k_2 \times S_S + k_3 \times L_S + k_4 \times P_S + k_5 \times R_S + k_6 \times T_S \tag{14}$$

In this, k1+k2+k3+k4+k5+k6≤1and are required to fine-tune the ranking of semantic associations i.e. if user wishes to give more importance to context in ranking semantic associations then k_1 should be more.

4.2. Reusing Previously Computed Values in Ranking Semantic Associations

As stated in Section 4.1, the final rank of a semantic association is computed as a function of six values viz. context, subsumption, association length, popularity, rarity, and trust. Therefore, all these six values are computed before computing the final rank. It is obvious that many components of the semantic associations may occur repeatedly in many of the semantic associations, therefore it is not needed to compute the values viz. context, subsumption, popularity, rarity, and trust of each component every time instead the previously computed values of the components may be reused in the subsequent computations thus reduce the overall computing time. Since path lengths of different associations may be different so this value can't be reused. This section describes how to reuse the previously computed values.

When computing the overall rank of an association, the values for all the six metrics described in Section 4.1 are computed and the overall rank is calculated. The values corresponding to context,

subsumption, popularity, rarity, and trust are stored in a Hash data structure. As shown in Figure 3, a Hash data structure is a collection of <compID, values> pairs, where compID denotes the ID of the component and values represents context, subsumption, popularity, rarity, and trust values of the corresponding component. It is used to store the pre-computed values of context, subsumption, popularity, rarity, and trust of each component. In computing the overall rank of the associations, following approach is used; split the association into components and store the components in a list by assigning compID to each component. For each component, check whether the component exist in a Hash data structure. If it exists, then retrieve context, subsumption, popularity, rarity, and trust values of the component and use these values in computing the overall rank. If the component does not exist in the Hash data structure, then compute context, subsumption, popularity, rarity, and trust values as described in Section 4.1, store these values in a Hash data structure and use these values in computing the overall rank. This process is continued for all the associations. Algorithm 1 is used to compute the overall rank of associations using this approach.
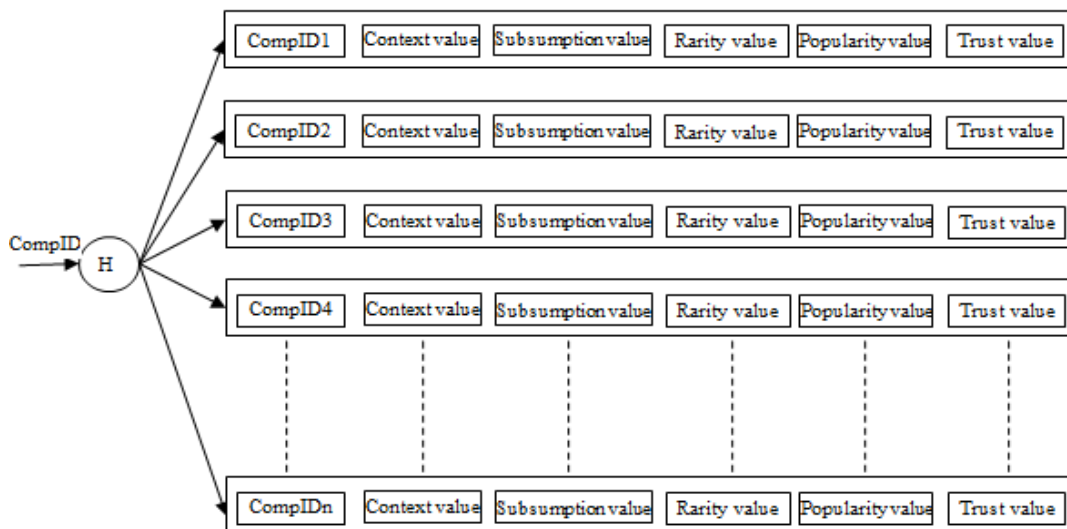

Figure 3. Hash Data Structure

5. RESULTS
5.1. Experimental Results

The ranking approach presented in this work has been implemented and tested on SWETO [11] ontology. SWETO ontology is a very popular populated ontology which contains the data pertaining to cities, countries, organizations, events, banks, companies, persons, researchers, and scientific publications. User is provided with a web-based user interface through which he can enter the two entities between which he wants to find associations as shown in Figure 4(a). To define the context, a touch graph like interface is provided. Using this, the user can define contextual regions as shown in Figure 4(b). As shown in Figure 4(c) the user interface also allows the user to customize his ranking criteria by specifying other criteria such as 'Favor rare or common associations', 'Favor popular or unpopular associations', and 'Favor short or long associations'. In addition user can specify values for k1, … , k6 to fine tune the ranking. Prior to ranking, the associations are stored in Oracle database. These unranked associations are then ranked based on the criteria defined in Section 4.
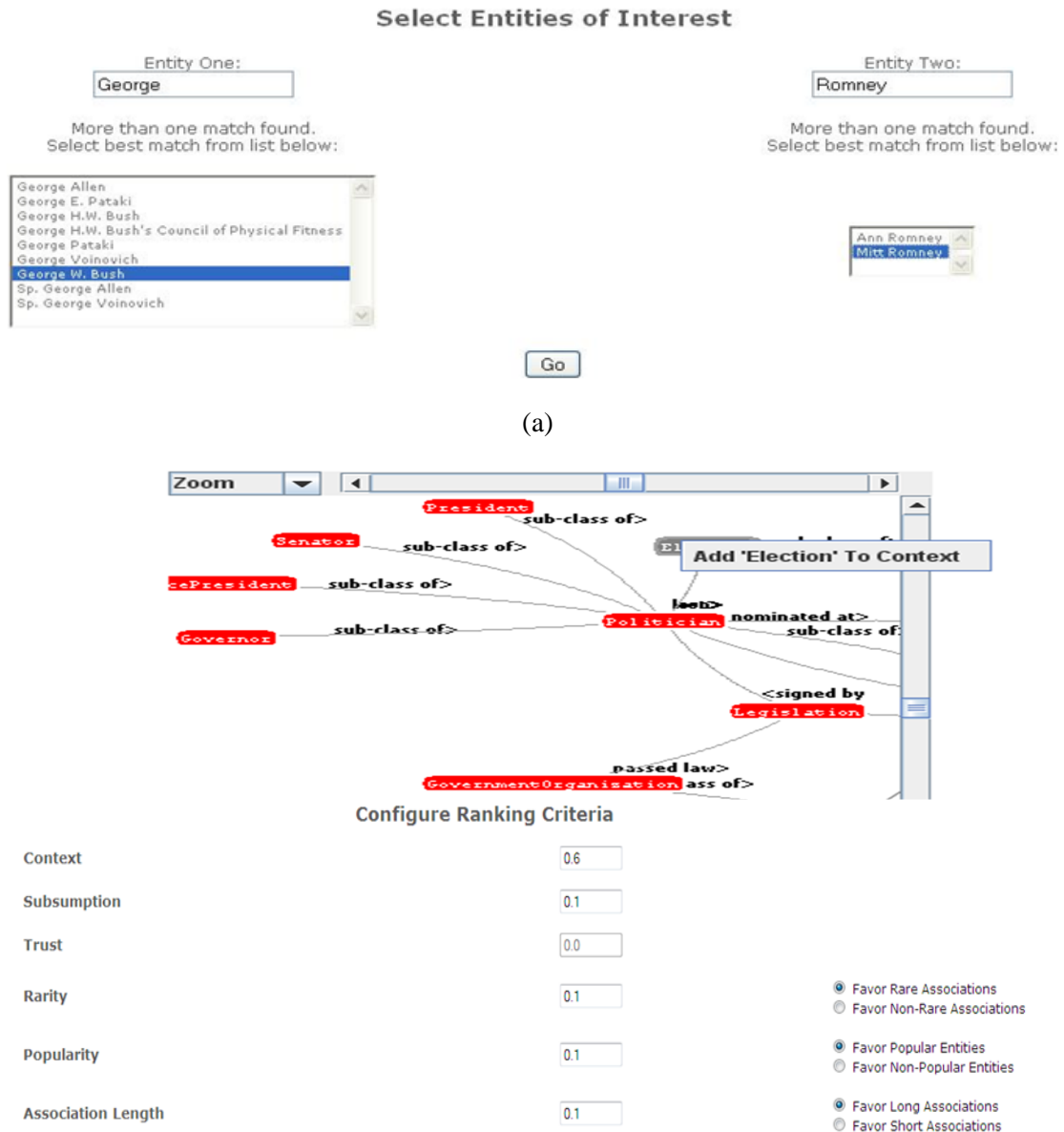
(a)

Figure 4. User interface for ranking semantic associations; (a) User Interface for entering two entities; (b) User interface for selecting contextual regions; (c) User interface for fine-tuning the ranking criteria

5.2. Preliminary Results

To demonstrate the efficiency of the proposed method, the execution time of the proposed method is compared with the execution times of Aleman Meza et al., and Viswanathan et al. methods for a sample query involving two entities **'George W. Bush'** and **'Mitt Romney'** . The experiments were conducted on a Laptop PC running Windows 7 with Intel ® Core ™ i5 CPU and 4 GB RAM. Figure 5 shows the comparison of execution time of the proposed method with Aleman Meza et al., and Viswanathan et al. methods. The x-axis shows top k-associations and the y-axis shows the execution time (in Seconds) taken for each top k-associations. As shown in figure 5, the execution times of all the three methods are increasing as the number of associations is

increasing. However, it shows that the execution time of the proposed method is very much less than the other two existing methods because it reuses the pre-computed values of the components in computing the overall rank of the associations. Figure 6 shows the total execution time (in Seconds) of the proposed, Aleman Meza et al., and Viswanathan et al. methods. It shows that, the total execution time of the proposed method is only 76.85 seconds which is very much less than the other two existing methods.
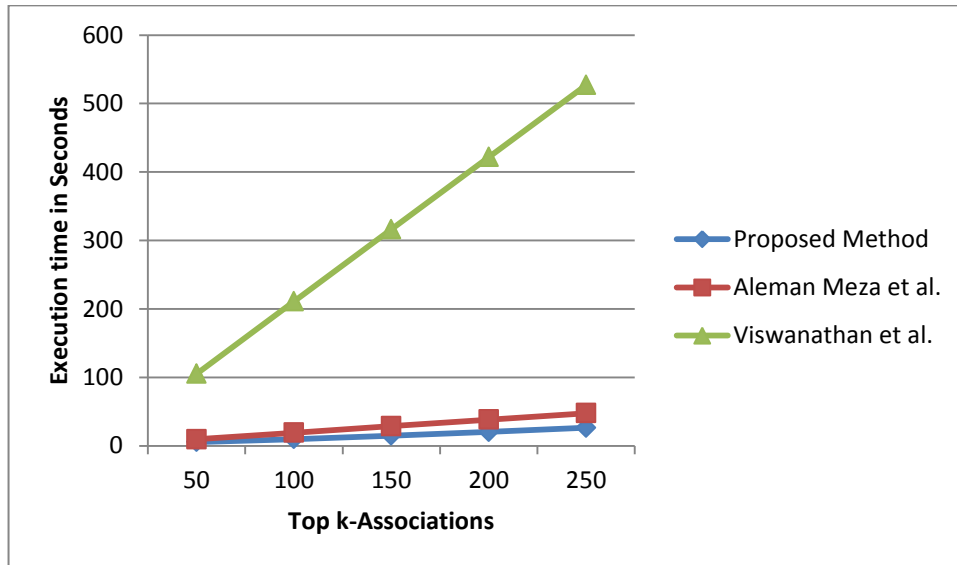

Figure 5. Comparison of execution time among the Proposed, Aleman Meza et al. and Viswanathan et al. Methods
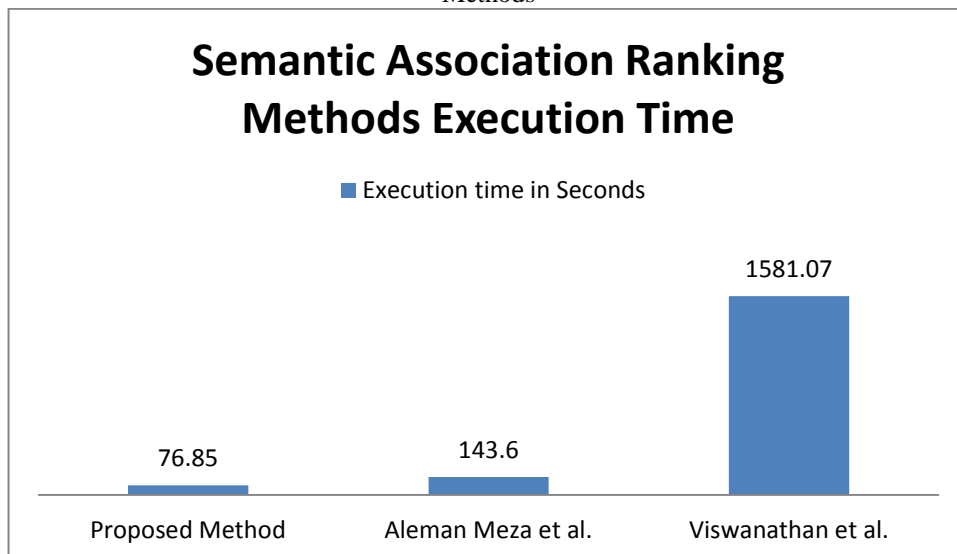

Figure 6. Comparison of total execution time among the Proposed, Aleman Meza et al. and Viswanathan et al. Methods

6. CONCLUSIONS

While the primary focus of the search techniques of the first generation Web is accessing relevant documents from the Web based on user requirements whereas the primary focus of the search techniques of the next generation Web called the Semantic Web is discovering relevant semantic associations between two entities present in large semantic metadata repositories. A semantic metadata repository contains large number of entities and relationships therefore the number of

associations that exist between two entities is increasingly overwhelming. Hence ranking of these associations is necessary to find more relevant associations as per the user interest. This paper proposed a method which reuses the previously computed values in computing the overall rank of the associations thus reducing the execution time. The experiments were conducted on SWETO ontology and the results show that the running time of the proposed method is much lesser than the other existing methods. It states that the proposed method ranks the semantic associations more efficiently. In future, this work can be extended by computing the ranks of the associations while the Bi-directional Breadth-First Search traverse the associations.

REFERENCES

[1]   Aleman-Meza Bonerges, Halaschek-Wiener Christian, Arpinar IB, Ramakrishnan Cartic, Sheth Amit (2005). Ranking Complex Relationships on the Semantic Web. IEEE Internet Computing 9(3); 37-44. Doi:10.1109/MIC.200.63.

[2]   Aleman-Meza, B., Halaschek, C., Arpinar, I. B., and Sheth, A. "Context-Aware Semantic Association Ranking." First International Workshop on Semantic Web and Databases, Berlin, Germany, 33-50.

[3]   Viswanathan V, Ilango K. Ranking semantic relationships between two entities using personalization in contest specification. Information Sciences, Elsevier, 207 (2012) 35-49.

[4]   Anyanwu Kemafor, Angela Maduko, Sheth Amit. SemRank: ranking complex relationship search results on the Semantic Web, in: Proc. of the 14th International World Wide Web Conference, ACM Press, 2005, pp. 117–127.

[5]   Myungjin Lee, Wooju Kim. Semantic association search and rank method based on spreading activation for the Semantic Web, in: IEEE International Conference on Industrial Engineering and Engineering Management, 2009, pp. 1523–1527.

[6]   Shahdad Shariatmadari, Ali Mamat, Ibrahim Hamidah, Mustapha Norwati (2008). SwSim:Discovering semantic similarity association in semantic web. Proceedings of International Symposium on ITSim 2008, 1-4.

[7]   Anyanwu Kemafor, Sheth Amit. ρ-operator: Discovering and Ranking Semantic Associations on the Semantic Web, ACM SIGMOD Record, v. 31 n.4, December 2002.

[8]   Berners-Lee, T., Hendler, J., and Lassila, O. (2001). "The Semantic Web - A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities." *Scientific American*, 284(5), 34.

[9]   Lassila Ora and Swick R. Resource Description Framework (RDF) Model and Syntax Specification, W3C Recommendation. 1999.

[10]   Brickley D and Guha RV. Resource Description Framework (RDF) Schema Specification 1.0, W3C Candidate Recommendation. 2000.

[11]   Aleman-Meza, B., Halaschek, C., Sheth, A., Arpinar, I. B., and Sannapareddy, G. "SWETO: Large-Scale Semantic Web Test-bed." *16th International Conference on Software Engineering and Knowledge Engineering (SEKE2004): Workshop on Ontology in Action*, Banff, Canada, 490-493.

[12]   Guha, R. V., and McCool, R. (2003). "TAP: A Semantic Web Test-bed." *Journal of Web Semantics*, 1(1), 81-87.

[13]   Opencyc. http://sw.opencyc.org

[14]   Sheth, A. P., Arpinar, I. B., and Kashyap, V. (2003). "Relationships at the Heart of Semantic Web: Modelling, 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. A. Zadeh, eds., Springer-Verlag.

[15]   Stumme, G., Hotho, A., Berendt, B.: Semantic Web Mining: State of the art and future directions. Web Semantics: Science, Services and Agents on the World Wide Web 4(2) (2006) 124 – 143.

[16]   Berendt, B., Hotho, A., Mladenic, D., van Someren, M., Spiliopoulou, M., Stumme, G.: A Roadmap for Web Mining: From Web to Semantic Web. Web Mining: From Web to Semantic Web Volume 3209/2004 (2004) 1–22.

[17] Halaschek, C., Aleman-Meza, B., Arpinar, I. B., and Sheth, A. P. "Discovering and Ranking Semantic Associations over a Large RDF Metabase." *30th International Conference on Very Large Data Bases*, Toronto, Canada.

[18] Sheth, A. P., Aleman-Meza, B., Arpinar, I. B., Halaschek, C., Ramakrishnan, C., Bertram, C., Warke, Y., Avant, D., Arpinar, F. S., Anyanwu, K., and Kochut, K. (2005a). "Semantic Association Identification and Knowledge Discovery for National Security Applications." *Journal of Database Management*, 16(1), 33-53.

[19] Narayana S, Govardhan A, Varma GPS. Discovering and Ranking Semantic Associations on the Semantic web, International Journal of Computer Science and Management Research, Vol. 1 Issue 5 December 2012, pp. 1092-1102.

[20] Narayana S, Govardhan A, Varma GPS, "An RDF Approach to Discover Relevant Semantic Associations", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 7, July 2013.

Authors

S. Narayana did his B.Tech in Computer Science and Engineering from Sri Krishna Devaraya University, Anatapur and M.Tech Degree from School of Information Technology, JNTU Hyderabad. His is currently pursuing Ph.D. from Jawaharlal Nehru Technological University, Hyderabad. He has a total of 12 years of teaching experience and guided number of M.Tech and B.Tech projects. He has presented 5 papers in conferences and published 10 papers at International journals. He is a life member of ISTE and member of CSI. His research interests include Semantic Web, Data Mining, and Web Mining.

Dr. G. P. Saradhi Varma received his B.E. (CSE) Degree from Manipal Institute of Technology Mangalore University, M.Tech from NIT (REC Warangal), Warangal and PhD (Specialized in Computer Science) Degree from Andhra University, Visakhapatnam. He is presently Professor and Head of IT, SRKR Engineering College, Bhimavaram. He is an Educational member and consultant to various companies and institutions in Andhra Pradesh. He has guided numerous M.Tech projects and B.Tech projects. He has a total of 30 research publications at International/National Journals and Conferences. He has delivered number of Keynote addresses and invited lectures and authored books on AFL, UML, and Web technology. He is also a member in various professional bodies. His areas of interest include Object Oriented Technologies, Information Retrieval, Algorithms, Computer Networks, and Image Processing.

Dr. A. Govardhan did his BE in Computer Science and Engineering from Osmania University College of Engineering, Hyderabad, M.Tech from Jawaharlal Nehru University, Delhi and PhD from Jawaharlal Nehru Technological University, Hyderabad. He guided more than 10 PhD theses, and more than 120 M.Tech projects and number of MCA and B.Tech projects. He has more than 150 research publications at International/National Journals and Conferences. He is a Member on various Advisory Boards. He has been a program committee member for various International and National conferences. He is also a reviewer of research papers of various conferences. He is a Member on the Editorial Boards for Eight International Journals. He has delivered number of Keynote addresses and invited lectures. He has chaired various sessions at International and National Conferences. He is also a member in various Professional and Service-Oriented bodies. His areas of interest include Databases, Data Warehousing & Mining, Information Retrieval, Computer Networks, Image Processing and Object Oriented Technologies.