# An RDF Approach for Discovering the Relevant Semantic Associations in a Social Network

Thushar A.K. #1, P. Santhi Thilagam #2

# Department of Computer Engineering, National Institute of Technology Karnataka,Surathkal
Mangalore - 575025, India
1 thushar_ak@yahoo.com
2 santhi@nitk.ac.in

***Abstract*—A social network is a network of interactions between entities of social interest like people, organisations, hobbies and transactions. Finding relevant associations between entities in a social network is of great value in many areas like friendship networks, biology and countering terrorism. Semantic web technology enables us to capture and process relationships among social entities as metadata. Analysing semantic social networks requires newer methods. In a social network, entities are connected by short chains of relationships. Query to find associations between two entities returns a large number of results. One of the major issues is to rank the associations as per user preference. The work presents an approach to rank two categories of semantic associations viz. common associations and informative associations. Associations are modelled as property sequences in an RDF graph and they are ranked based on preferred search mode. Heuristics such as i) information content due to occurrence of a property with respect to all the properties in a description base ii) unpredictability of an association due to participation of its properties in multiple domains iii) the extent of match between user specified keywords and properties and iv) the popularity of nodes involved in a sequence are used to rank associations. The results obtained suggest that these heuristics indeed help in obtaining relevant associations. To scale the results to large RDF graphs, a relevant subgraph is extracted from the input graph on which ranking is applied. The approach is tested successfully on real RDF datasets and multigraphs.**

## I. Introduction

A social network is a set of people or group of people with some pattern of interaction between them [1]. Social networks have interesting properties. They influence our lives enormously without us being aware of the implications they raise: Suppose you are on a marketing campaign for a product intended for enterprises. You need to target on a few companies where you can sell your product. What you need is a few influential contacts in the company to whom you can introduce your product, and hope to gain an order for your product. If you could find the relevant ways by which you are connected to these influential people,you have your task cut out. You can follow one of these convenient ways to establish contact with them. Consider another scenario where there is suspicion in the behaviour of two passengers travelling by a flight as to whether they have a terrorist link. Definitely, here we will be interested in finding rare associations like their participation in some unlawful transactions.This motivates the need for finding relevant semantic associations as per the requirements of the domain.

Traditional social network methods focused on statistical properties of network like randomness, power law distributions, clustering and centrality. These studies have generally considered only one type of social entity and one type of relationship: for example the email communications between a set of people. For the purposes of this study, we have considered networks with multiple types of social entities and relationships, since real world networks will generally be of this form. Semantic web technologies help to effectively represent relationships as properties of social entities. One of the actively pursued standards is the Resource Description Framework(RDF)[2]. RDF is a W3C standard used for describing resources using a simple model based on named relationships between resources. RDF has an abstract syntax [3] that reflects a simple graph-based data model, and formal semantics [4] with a rigorously defined notion of entailment providing a basis for well founded deductions in RDF data.An RDF statement, which is a triple of the form `(Subject,Property,Object)` asserts that a resource, the `Subject` has a property whose value is the `Object` (which can be either another resource or a literal). This model can be represented as a labeled directed graph where nodes represent the resources and arcs representing properties whose source is the subject and target is the object, and are labeled with the name of the property. The meaning of the nodes and arcs is derived from the connection of these nodes and arcs to a vocabulary. The vocabulary describes types of entities and types of properties for the domain. The vocabulary description is done using the companion specification to RDF called the RDF Schema Specification [5].

There are query languages like RQL using which one can find direct relationship among entities. But in a social network, there can be more complex relationships among entities. The small world effect of social networks states that most pairs of vertices are connected through a short path through the network. This leads to a combinatorial explosion in the number of associations between two entities. Appropriate heuristics are required to rank the paths and provide user with a set of most significant paths.In this work we address the problem of ranking semantic associations in social networks represented as RDF Graphs.

## II. Related Work

The notion of semantic associations is presented as complex relationships between resource entities by Anyanwu *et al.*. [6]. These relationships capture both connectivity of entities as well as similarity of entities based on a specific notion of similarity called $\rho$-isomorphism.

A modulated approach for ranking semantic associations is proposed by Anyanwu *et al.*. [7]. They consider various parameters like information content of a semantic association with respect to occurrence of a property, participation of properties to classes belonging to different schema and matching of keywords to properties to develop a model for ranking property paths.

Given a knowledge base and any two entities X and Y there could be a myriad of relatively short chains of relationships linking the two. Hence the need for some way of semantically constraining the discovery of possible ways in which X and Y could be related. Faloutsos *et al.* [8] address this issue by developing an algorithm to extract relatively small connection subgraphs. Formally,they considered the following problem: Given an edge-weighted undirected graph *G*, vertices *s* and *t* from *G*, and an integer budget *b*, find a connected subgraph *H* containing *s* and *t* and atmost *b* other vertices that maximises a goodness function $g(H)$. The graph is interpreted as an electrical network in which each edge *e* represents a resistor with conductance $C(e)$ and a connection subgraph that can deliver as many subunits of electric current is chosen. To avoid the expensive computations involving large graphs, an optional precursor step called *candidate generation* is proposed. This step extracts a subgraph that contains the most important paths.

Ramakrishnan *et al.* [9] adapted the approach to the more general problem of RDF graph. They proposed heuristics for edge weighting that depend indirectly on the semantics of entity and property types in the ontology and on characteristics of instance data. More specifically they defined *Class* and *Property Specificity*,*Instance Participation Selectivity* and a *Span Heuristic*. The aim is to use semantics suggested by the schema to systematically convert an arbitrary un-weighted RDF graph into an edge-weighted graph appropriate as input to the algorithm described previously.

Sougata *et al.* [10] presents a system that facilitates information retrieval from biomedical patents. The system determines the importance of resources to rank the results of a search and prevent information overload while determining the semantic associations. They propose that a resource that has relationships with many other resources in the semantic web can be considered to be important since it is an important aspect of overall semantics; the meaning of many other resources of the semantic web have to be defined with respect to that resource.In the context of a property graph vertices that have high indegree or outdegree should be considered important.

A semantic web application that detects Conflict of Interest(COI) relationships among potential reviewers and authors of scientific papers is described by Aleman Meza *et al.*[11]. This application discovers various semantic associations between the reviewers and authors in a populated ontology to determine a degree of Conflict of Interest. This ontology was created by integrating entities and relationships from two social networks namely *"knows"* from a Friend-of-a-Friend(FOAF) social network and *"co-author"* from the underlying co-authorship network of the Digital Bibliography and Library Project(DBLP). They assess the weight of significant relationships that exist between reviewers and authors using specific heuristics. For example, the strength of a `co-authors` relationship from $A$ to $B$ is obtained as the ratio of number of publications co-authored by the two to the total number of publications of $A$.This strength is used to determine the level of COI that exists between reviewers and authors.

Brahms [12] is an RDF storage system designed to support fast semantic association discovery in large RDF bases. The features include search for associations of variable length and unspecified directionality, work on large RDF Graphs in main memory with leaving a sufficient amount of memory for the operation of search algorithm and limit traversal paths to instance resources only. But Brahms has not addressed the problem of ranking semantic associations.

## III. Problem Description

The research problem is to design a framework to *find the k most significant simple paths between any two nodes in a social network* where $k$ is a user specified number. By significant associations we mean associations which add value to the user in a significant way. It varies according to the domain of interest.For the purposes of this work, we consider two categories of significant associations,viz,*common associations* and *informative associations*. Common associations are characterised by frequently occurring properties, highly connected nodes and paths which follow the schema of a single domain. Informative associations are those which contain the least frequently occurring property, less connected nodes and those which span multiple schemas.

Input is a RDF graph representing the social network. The graph can have multiple arcs connecting two nodes as it will be the case in a real world social network. The nodes between which the paths are to be found is input by the user. Output is $k$ most common paths and $k$ most informative paths between the nodes if any path exist between them.Alternatively these paths will be called conventional and discovery mode paths.

Formally, given an RDF Graph G=(V,E) with a mapping $\lambda : E \rightarrow P$ where $P$ is the set of all properties in the RDF Schema and two vertices $v_1$ and $v_n$, find $k$ edge sequences of the form $e_a, e_b, .....e_m$ connecting $v_1$ and $v_n$ that best captures the relationship between the two entities. We assume that the properties in the RDF Graph are bidirectional (i.e. every relationship has a corresponding inverse relationship). This assumption is necessary because two resources may not be coonected by a directed path but by a path which contains inverse relations. Ignoring this path could exclude vital information about connections between the entities.

## IV. PRELIMINARIES

### A. RDF Syntax and Semantics

RDF has been used to make statements about web resources. RDF uses Uniform Resource Identifier(URI)s to identify things it describe. The primary difference between a Uniform Resource Locator(URL) used in the Web and a URI is that a URI can be used to refer to anything in the Universe, not just network accessible things like Webpages. More precisely RDF uses *URI References(URIref)* which is a URI together with an optional fragment identifier at the end. RDF defines a *Resource* as anything that is identifiable by a URI Reference, so using URIrefs allows RDF to describe practically anything and to state relationship between such things as well. To represent RDF statements in a machine-processable way, RDF uses the Extensible Markup Language (XML). XML was designed to allow anyone to design their own document format and then write a document in that format. RDF defines a specific XML markup language, referred to as RDF/XML, for use in representing RDF information, and for exchanging it between machines. The term vocabulary is used when referring to a set of URIrefs defined for some specific purpose, such as the set of URIrefs defined by RDF for its own use, or the set of URIrefs defined by a university to identify its students. Use of URIrefs make the identification of subjects, objects and the relationships precise. It also supports the development and use of shared vocabularies on the web, since people can discover and begin using vocabularies already used by others to describe things, reflecting a shared understanding of those concepts.

The RDF semantics is based on model theory which assigns meaning to RDF statements based on an intermediate abstract structure. The mapping of components in an RDF statement to entities in the structure is formally called *Interpretation*. URI references are mapped into the set of resources they represent, called *Domain*.URI References identifying a property $p$ is mapped to a set of ordered pairs $(x, y)$ where $x$ and $y$ belong to the Domain. The basic intuition of model-theoretic semantics is that asserting a sentence makes a claim about the world: it is another way of saying that the world is, in fact, so arranged as to be an interpretation which makes the sentence true. Reasoning in RDF is monotonous in the sense that addition or deletion of any triple does not affect the validity of existing triples. A consequence of this is an instance can belong to multiple classes as a result of its participation in different properties.

### B. Semantic Associations

To capture the notion of semantic associations between entities, we define a property sequence as follows: A property sequence PS is a sequence of properties $[P_1, P_2....P_n]$ where $P_i$ is a property defined in an RDF schema $RS_j$ of a schema set $RSS$. As an example,consider an ontology capturing the relationships that exists in a university illustrated in Fig. 1.

The ontology has two perspectives, one which is purely academic and the other with regard to the cultural societies

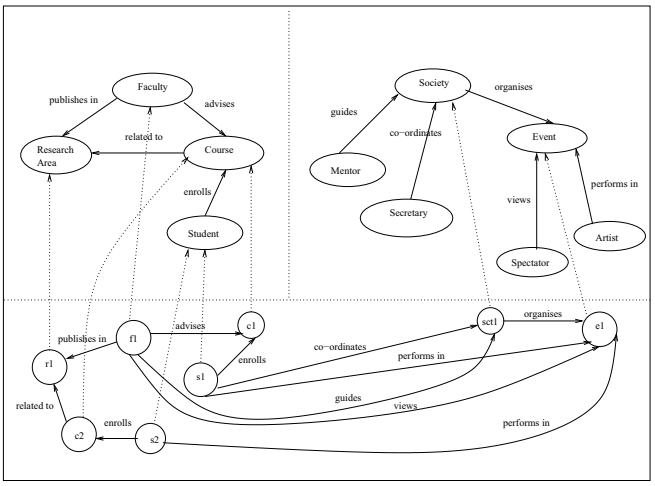



Fig. 1. An example RDF Knowledge Base

which exist in a university. These two perspectives are captured by two schema represented on the upper part of the diagram and separated by dotted lines. The lower part of the diagram indicates the instances of the ontology and the connections between them. The interpretation of PS is given by: $[[PS]] \subseteq \times_{i=1}^{i=n}[[P_i]]$ where for $ps \in [[PS]]$ called an *instance* of $PS, ps[i] \in [[P_i]]$ for $1 \leq i \leq n$ and $ps[i][1] = ps[i+1][0]$. $ps[i][1]$ refers to the second element of the $i$ th ordered pair and $ps[i+1][0]$ refers to the first element of $(i+1)$ st ordered. *pathAssociated(x,y)* is true if there exists a property sequence $ps \in [[PS]]$ and either x and y are the origin and terminus of ps respectively, or vice verse, i.e. y is origin and x is terminus. For the example ontology in Fig.1, some of the semantic associations that exist between $f1$,an instance of faculty and $s1$,an instance of student are enumerated in Fig. 2.

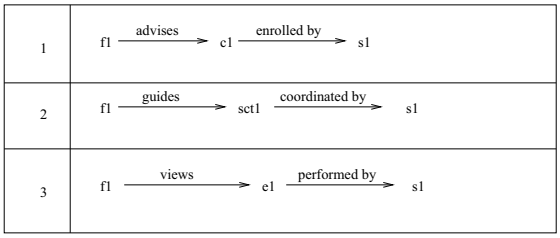



Fig. 2. Some Semantic Associations in the RDF Knowledge Base in Fig. 1.

## V. HEURISTICS FOR RANKING SEMANTIC ASSOCIATIONS

### A. Information Gain[7]

In information theory, the amount of information contained in an event is measured by the negative logarithm of occurrence of an event. Thus if $\chi$ is a discrete random variable or an event that has possible outcome values $x_1, x_2.....x_n$ occurring with probabilities $pr_1, pr_2.....pr_n$ i.e.,, $Pr(\chi = x_i) = pr_i$ with $pr_i \geq 0$ and $\sum_{\forall i} pr_i = 1$, the amount of uncertainty removed by knowing that $\chi$ has the outcome $x_i$ is given by $I(\chi = x_i) = -log\ pr_i$. Based on this we can build a model for measuring the information content of a semantic

association by considering the occurrence of an edge as an event and occurrence of a property as an outcome. Probability of occurrence of property $p$,

$$Pr(\chi = p) = |[[p]]\hat{}\,|/|[[P]]\hat{}\,|$$

where $[[p]]\hat{}$ denotes the set of proper instances of $p$ and $[[P]]\hat{}$ denote the set of proper instances of all the properties in the ontology. This probability is called *specificity* of property $p$. Information content of a property in the description base due to its specificity is

$$I_s(p) = I(\chi = p) = -log \Pr(\chi = p)$$

Let $PS = p_1, p_2...., p_n$ be a property sequence and $ps \in [[PS]]$ be a path. It is clear that $ps$ occurs as frequently as the least frequently occurring property $p_i$ in PS.The information content of ps due to its specificity is

$$I_s(ps) = max_{\forall i}\{I_s(p_i)\}$$

### B. Refraction[7]

Refraction refers to deviation from a path's representation at the schema layer.Paths with many refractions are unlikely to be easily anticipated by users, making them less predictable.Thus refractions add to the information content of an association. For example consider the following property sequences connecting students s1 and s2 in the RDF graph in Fig. 1.

1) `s1.enrolls.c1.advised by.f1.guides. sct1.organises.e1.performed by.s2`
2) `s1.enrolls.c1.advised by.f1.publishes in.r1.related to.c2.enrolled by.s2`

The former path is preferred in a discovery mode search because it deviates from one schema to another at the property `guides` while the latter path follows only one schema. The resource `f1` gets classified into both `faculty` and `mentor` as a result of its participation in the properties `advised by` and `guides`. Formally for a path sequence $PS = p_1, p_2, ....., p_n$

$$refraction(p_i, p_{i+1}) = \begin{cases} 1 & \text{if } e_i \text{ is not adjacent to } e_{i+1} \text{ in the RDF schema Graph} \\ 0 & \text{otherwise} \end{cases}$$

*Refraction Count* RC refers to the number of refractions on a path, given by

$$RC(ps) = \begin{cases} \sum_{i=1}^{n-1} refraction(p_i, p_{i+1}) & \text{for } n \geq 2 \\ 0 & \text{otherwise} \end{cases}$$

### C. Node Popularity

The existing approaches for ranking semantic associations have considered either the semantics of the links or the attributes of nodes but not both of them simultaneously. Taking a cue from ranking of web pages, it is obvious that nodes with more number of connections are more popular than those with fewer connections. So paths which have more number of highly connected nodes should be preferred in a conventional search and vice verse in discovery mode search. As an example, consider two paths connecting two students in a university ontology. One is through a faculty who has so many connections and the other is through another student who is sparsely connected. The former path is preferred in conventional search while the latter is preferred in discovery mode search. To make this notion precise, we can assign a connectivity score to a property sequence as:

$$CS(ps) = \frac{n_d}{N}$$

where $n_d$ is the number of nodes in the path having degree greater than the average degree of the graph and $N$ is the total number of nodes in the path.

### D. Keywords[7]

The user can be allowed to specify a set of interested properties in the form of keywords. Then the importance of a property with respect to a keyword can be determined by measuring at what level in the property hierarchy, a keyword matches a property. Given a property sequence $PS = p_1, p_2, ....., p_n$ and a set of keywords $K = k_1, k_2, ....k_m$, the degree of match between $k_i$ and $p_j$ is given by
$KMatch(k_i, p_j) = 0 \leq (2^d)^{-1} \leq 1$ where $d$ is the distance between properties in a property hierarchy.

$$KMatch(ps) = \sum_{i=1}^{n}\{Kmatch(k_i, p_j)\}$$

### E. Overall Rank

All the above factors contribute to the semantic rank of a property association. The exact nature in which they are combined is dependent on the search mode which can be a common association search or informative search. For a common association search, rank is inversely proportional to the information content from the search. Rank of an association due to the information content can be expressed as:

$$Rank_I(ps) = (1 - \mu)(I(ps))^{-1} + \mu I(ps)$$

where $\mu = 0$ for a common association search and $\mu = 1$ for an informative search. This leads to higher rank values being assigned to the most unpredictable paths in informative mode, and lower rank values being assigned at the common mode. The refraction count affects only the informative mode search and hence the rank of a property sequence due to refraction can be expressed as:

$$Rank_R(ps) = \mu RC(ps)$$

Overall rank in a conventional search is given by:

$$Rank_C(ps) = Rank_I(ps) + CS(ps) + KMatch(ps)$$

Overall rank in a discovery mode search is given by:

$$Rank_D(ps) = Rank_I(ps) + (1 - CS(ps)) + Rank_R(ps) + KMatch(ps)$$

## VI. Implementation

The implementation is done using Standard Template Library and g++ 4.1.1. Raptor 1.4.17 Library is used for parsing the RDF file and accessing the triples. The RDF schema triples containing $subPropertyOf$ relation is parsed and properties are assigned a hierarchical labelling as shown in Fig. 3. The label of a child in the hierarchy is obtained by appending a number to the label of the parent. This helps in computing the keyword match information with regard to the context.

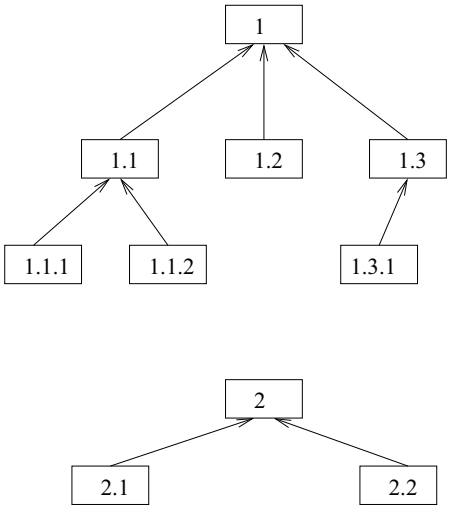


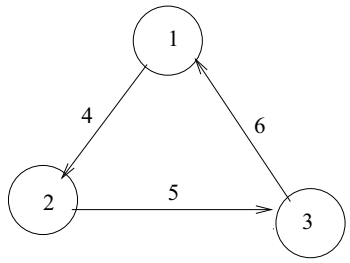

Fig. 3. Hierarchical Labelling of Properties

Each resource and each relation between resources is given a unique numeric identifier. The RDF instance Graph is represented as an edge list using the STL map data structure: each vertex number is mapped to the list of edges incident on the vertex. Edges outgoing with respect to the vertex are given a positive sign while those incoming are given a negative sign. This is just to keep track of the direction of an edge with respect to a vertex when finding the paths. Further, every edge is mapped to the originating vertex id and terminating vertex id. This helps in finding the vertices adjacent to a vertex easily even in multigraphs. This representation is illustrated with an example in Fig. 4.

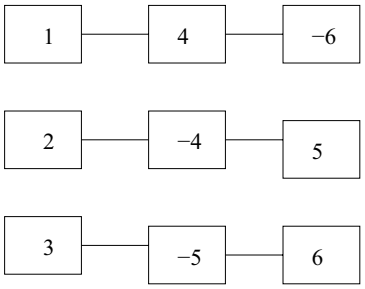
The only triples being considered are of the form $< a >< b >< c >$ where $a, c$ are resources and $b$ is a user defined property in the ontology. Paths are found using depth first search based algorithm. Paths are ranked as they are traversed as shown in Algorithm 1 and inserted into a vector with associated weight. The vector of paths is sorted to find the $k$ most significant paths. The time complexity of the brute force method of finding all paths is exponential. Hence, it is prohibitive for large graphs. The candidate generation approach proposed by Faloutsos *et al.* [8] is used to generate a smaller graph which contains the important connections.

## VII. Results and Discussion

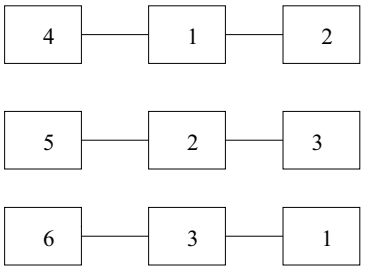
The results of common association query between resources $b$ and $c$ are listed in Fig. 6. As expected, paths containing the




(a) A coded RDF Graph




(b) Representing the graph as an adjacency list of edges




(c) Mapping each edge to source and destination vertices

Fig. 4. Graph representation used in the implementation

properties *teaches* and *friend of* are ranked higher compared to paths which contain only rare properties. Also the path through node *a* is preferred in a common association query since it has many connections. On the other hand while finding informative associations, the path which contain the least frequent property is ranked higher. This is evident from the results of informative association query between *b* and *c* shown in Fig. 7. Multiply classified resources on the path add to the information content of the association. Hence presence of resource *a* adds high rank to the path listed first in Fig.7 as *a* is classified both as *customer* and *student* according to the RDF file in Fig. 5 . User specified keywords contribute to ranking in such a way that paths containing properties which match the keywords at any level in the property hierarchy are preferred over other paths. This is illustrated in Fig. 8 when *extracurricular_activity_of* is specified as a keyword during the query to find common associations between *b* and *c*. The paths which contain the same keyword, subproperties or superproperties there of are ranked higher compared to other paths. The ranking is assigned based on the distance between the two labels in the property hierarchy tree.

**Algorithm 1** Rank a path from s to t in G

**for all** $edge$ on the path such that $terminal\_vertex(edge) \neq t$ **do**
  **if** $(frequency(edge) < min\_frequency)$ **then**
    $min\_frequency = frequency(edge)$
  **end if**
  **if** $degree(terminal\_vertex(edge)) > average\_degree(G)$ **then**
    $count\_popular\_nodes = count\_popular\_nodes + 1$
  **end if**
  **if** $edge.label$ matches a user specified keyword **then**
    $keyword\_match = keyword\_match + 1$
  **end if**
  **if** $Range\_Class(edge) \neq Domain\_Class(next\_edge)$ **then**
    $Refraction\_Count = Refraction\_Count + 1$
  **end if**
  $path\_weight = wt\_frequency + wt\_popularity + wt\_keyword + wt\_refraction$
**end for**

```
<univ:teaches> rdfs:subPropertyOf
<univ:guides>.
<univ:has_hobby> rdfs:subPropertyOf
<univ:extracurricular_activity_of>.
<univ:teaches> rdfs:domain <univ:faculty>.
<univ:teaches> rdfs:range <univ:student>.
<bank:has_account> rdfs:domain <bank:customer>.
<bank:has_account> rdfs:range <bank:branch>.
<bank:has_account_held_by> rdfs:domain
<bank:branch>.
<bank:has_account_held_by> rdfs:range
<bank:customer>.
<univ:has_hobby> rdfs:domain <univ:student>.
<univ:has_hobby> rdfs:range <univ:activity>.
<univ:friend_of> rdfs:domain <univ:student>.
<univ:friend_of> rdfs:range <univ:student>.
<univ:b>  <univ:teaches>  <univ:a>.
<univ:c>  <univ:is_taught_by>  <univ:b>.
<univ:b>  <univ:teaches>  <univ:d>.
<univ:b>  <univ:teaches>  <univ:e>.
<univ:b>  <univ:teaches>  <univ:f>.
<univ:a>  <univ:friend_of>  <univ:c>.
<univ:c>  <univ:friend_of>  <univ:a>.
<univ:d>  <univ:friend_of>  <univ:c>.
<univ:c>  <univ:friend_of>  <univ:d>.
<univ:a>  <univ:has_hobby>  <univ:h1>.
<univ:h1> <univ:is_hobby_of> <univ:c>.
<univ:b> <univ:has_account> <bank:branch1>.
<bank:branch1>
<bank:has_account_held_by> <univ:a>.
```

Fig. 5. An example RDF file in ntriples format.

The ranking algorithm was applied to real RDF datasets like Semantic Web Technology Evaluation Ontology(SWETO) [13]. SWETO was created to address the requirements of an ontology test-bed to support research in semantic analytics. The data set size and time taken for execution are summarised in Table I.

| | | |
|---|---|---|
| 1 | b —teaches→ a —friend of→ c | 1.588 |
| 2 | b —teaches→ a —has_hobby→ h1 —is hobby of→ c | 0.77 |
| 3 | b —has account→ branch1 —has account held by→ a —friend of→ c | 0.77 |
| 4 | b —has account→ branch1 —has account held by→ a —has_hobby→ h1; h1 —is hobby of→ c | 0.60 |

Fig. 6. Result of Common Association query between b and c on the RDF file of Fig. 5

| | | |
|---|---|---|
| 1 | b —has account→ branch1 —has account held by→ a —has_hobby→ h1; h1 —is hobby of→ c | 5.36 |
| 2 | b —has account→ branch1 —has account held by→ a —friend of→ c | 5.20 |
| 3 | b —teaches→ a —has_hobby→ h1 —is hobby of→ c | 4.20 |
| 4 | b —teaches→ a —friend of→ c | 2.70 |

Fig. 7. Result of Rare Association query between b and c on the RDF file of Fig. 5

TABLE I
DATA SETS AND EXECUTION TIME

| Data Set | Size | Time for execution |
|---|---|---|
| US Senate Data | 264.9 KB | $0.5s$ |
| SWETO | 13.6 MB | $11.33s$ |
| SWETO DBLP | 951.4 MB | $7m33s$ |

## VIII. CONCLUSION AND FUTURE WORK

The main contribution of the work is identifying two potential significant associations,viz. common and informative associations based on which paths can be ranked in social network applications. Heuristics for ranking paths in each of these associations is proposed. A prototype implementation is done and tested on various RDF datasets and results obtained are intuitive. Real world social networks can contain multiple direct links between entities and our approach considers such a scenario also. With the proliferation of online social networks and availability of semantic content in the form of RDF and related technologies, semantic analysis of social networks will become a promising research area. Standardised vocabularies for different social concepts will help to integrate the semantic data from various social networks. In such a scenario,finding meaningful associations between two resources becomes an important task.

The focus has been on identifying the heuristics for semantically ranking the associations. More heuristics can be added as per the requirement of the domain. The correlation between the heuristics needs to be investigated to obtain a clear idea of how each heuristic contribute to a particular type of association. The algorithm used can still be improved by further reducing the size of input graph. Experiment with

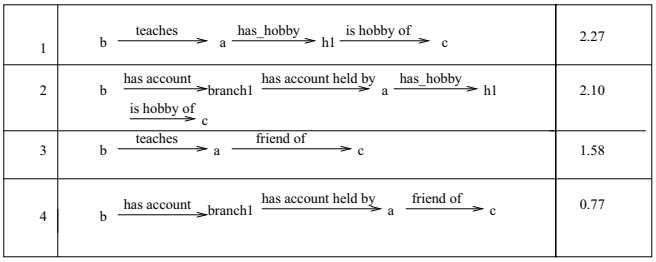



Fig. 8. Result of Rare Association query between b and c on the RDF file of Fig. 5 with Keyword "extracurricular activity of"

different social network data sets needs to be done. Presently such data sets in RDF format are very few in number.

## REFERENCES


[1] M.E.J. Newman, " The structure and function of complex networks," *SIAM Review*,vol. 45,pp. 167-256,2003.

[2] (2007) RDF Primer, W3C Recommendation 10 February 2004.[Online]. Available: http://www.w3.org/TR/2004/REC-rdf-primer-20040210/

[3] (2007) Resource Description Framework(RDF): Concepts and Abstract Syntax, W3C Recommendation 10 February 2004.[Online]. Available: http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/

[4] (2007) RDF Semantics, W3C Recommendation 10 February 2004.[Online]. Available: http://www.w3.org/TR/2004/REC-rdf-mt-20040210/

[5] (2007) RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation 10 February 2004.[Online]. Available: http://www.w3.org/TR/2004/REC-rdf-schema-20040210/

[6] K. Anyanwu and A. Sheth, " $\rho$-Queries:enabling querying for semantic associations on the Semantic Web," *in Proc WWW*,2003,pp. 690-699.

[7] K. Anyanwu, A. Maduko, and A. Sheth, "SemRank:Ranking Complex relationship search results on the Semantic Web," *in Proc WWW*,2005.

[8] C. Faloutsos, K. S. McCurley, and A. Tomkins, "Fast Discovery of Connection Subgraphs," *in Proc Tenth ACM SIGKDD Conference*,2004.

[9] C. Ramakrishnan, W. H. Milnor, M. Perry, and A. Sheth, "Discovering informative connection subgraphs in multirelational graphs," *SIGKDD Explorations*,vol. 7,no. 2,pp. 56-63,2005.

[10] S. Mukherjee and B. Bamba, "BioPatentMiner: An information retrieval system for biomedical patents," *in Proc thirtieth VLDB Conference*,2004.

[11] B. Aleman Meza, M. Nagarjan, C. Ramakrishnan, L. Ding, P. Kolari, A. Sheth, I. Arpinar, A. Joshi, and T. Finin, "Semantic Analytics on Social Networks: Experiences in addressing the problem of Conflict of Interest detection," *in Proc WWW*,2006.

[12] M. Janik and K. Kochut, "BRAHMS: A WorkBench RDF Store And High Performance Memory System for Semantic Association Discovery," *in Proc Fourth International Semantic Web Conference ISWC*,2005.

[13] (2008) Knoesis Center, Wright State University.[Online]. Available: http://knoesis.wright.edu/library/ontologies/sweto/