A Flexible Approach for Ranking Complex Relationships on the Semantic Web

By: Chris Halaschek

Advisors: Dr. I. Budak Arpinar
Dr. Amit P. Sheth

Committee: Dr. E. Rodney Canfield
Dr. John A. Miller

Outline

- Background
- Motivation
- Ranking Approach
- System Implementation
- Ranking Evaluation
- Conclusions and Future Work

The Semantic Web [2]

- An extension of the Web
  - Ontologies used to annotate the current information on the Web
  - RDF and OWL are the current W3C standard for metadata representation on the Semantic Web
- Allow machines to interpret the content on the Web in a more automated and efficient manner

Semantic Web Technology Evaluation Ontology (SWETO)

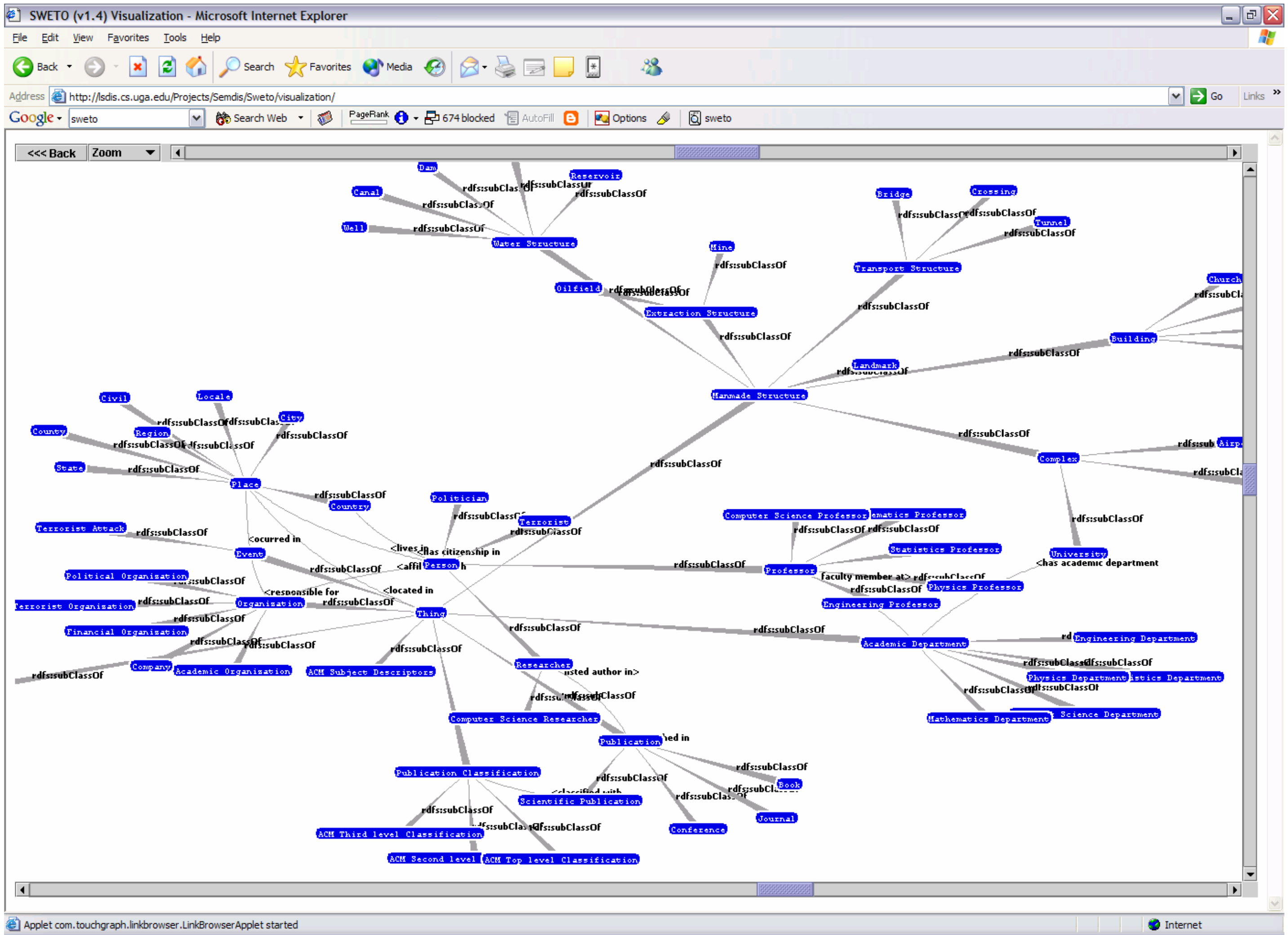
- Large scale test-bed ontology containing instances extracted from heterogeneous Web sources
- Developed using Semagix Freedom[1]
  - Created ontology within Freedom
  - Use extractors to extract knowledge and annotate with respect to the ontology

[1]Semagix Inc. Homepage: http://www.semagix.com

SWETO - Statistics

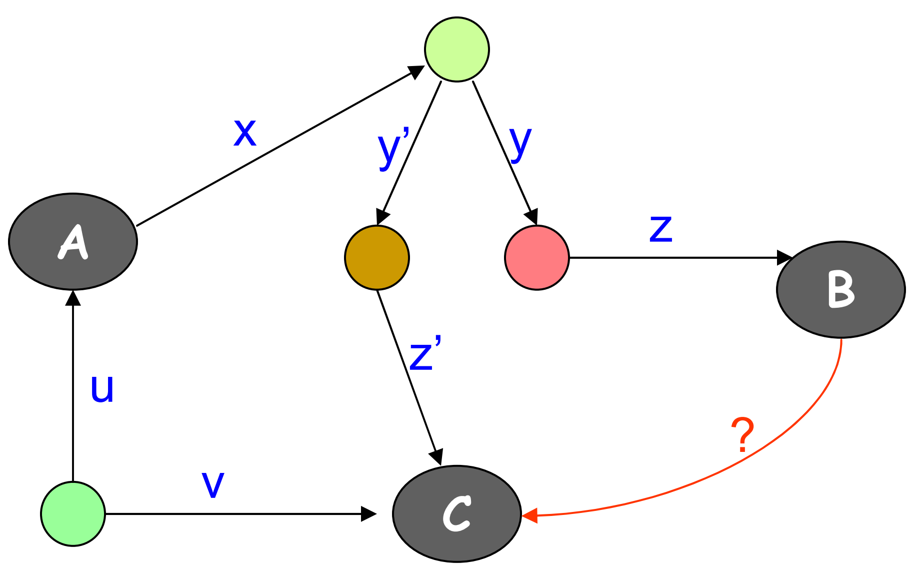
- Covers various domains
  - CS publications, geographic locations, terrorism, etc.
- Version 1.4 includes over 800,000 entities and over 1,500,000 explicit relationships among them

SWETO Schema - Visualization



Semantic Associations [1]

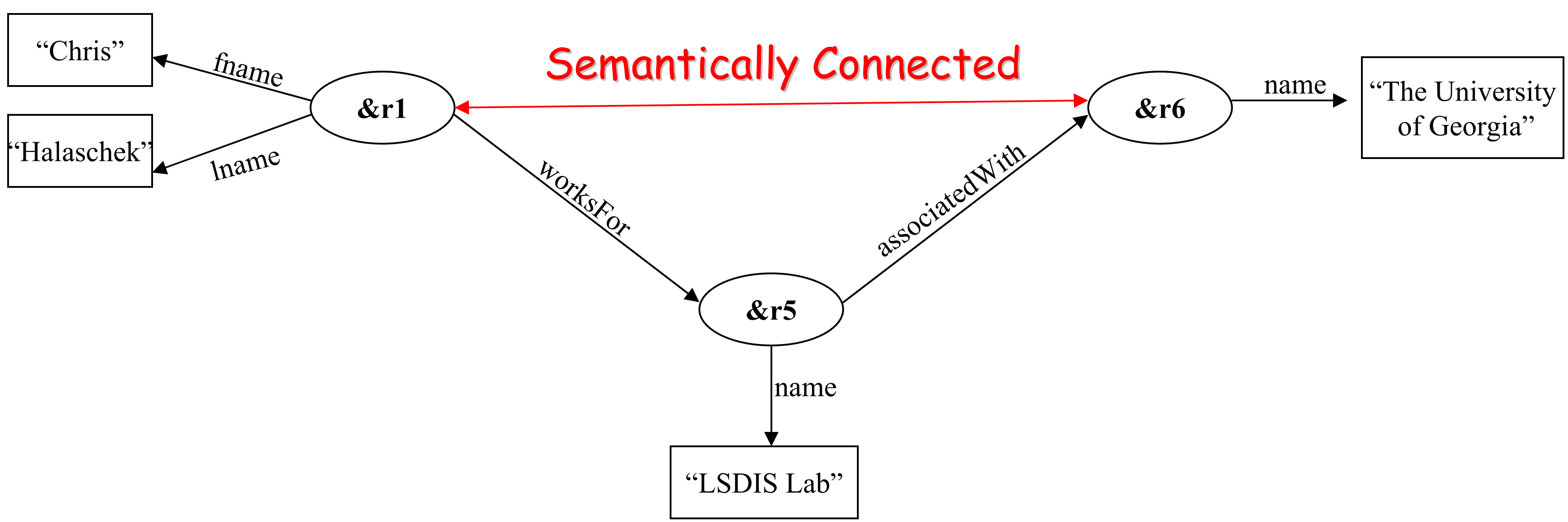
- Mechanisms for querying about and retrieving complex relationships between entities



1. A is related to B by x.y.z
2. A is related to C by
   i. x.y'.z'
   ii. u.v (*undirected path*)

3. A is "related *similarly*" to B as it is to C
   (y' ⊆ y and z' ⊆ z → x.y.z ≅ x.y'.z')
   So are B and C related?

Semantic Connectivity Example



Motivation

- Query between "*Hubwoo* [Company]" and "*SONERI* [Bank]" results in 1,160 associations
- Cannot expect users to sift through resulting associations
- Results must be presented to users in a relevant fashion…need ranking

Observations

- Ranking associations is inherently different from ranking documents
  - Sequence of complex relationships between entities in the metadata from multiple heterogeneous documents
  - No one way to measure relevance of associations
- Need a flexible, query dependant approach to relevantly rank the resulting associations

Ranking – Overview

- Define association rank as a function of several ranking criteria
- Two Categories:
  - *Semantic* – based on semantics provided by ontology
    - Context
    - Subsumption
    - Trust
  - *Statistical* – based on statistical information from ontology, instances and associations
    - Rarity
    - Popularity
    - Association Length

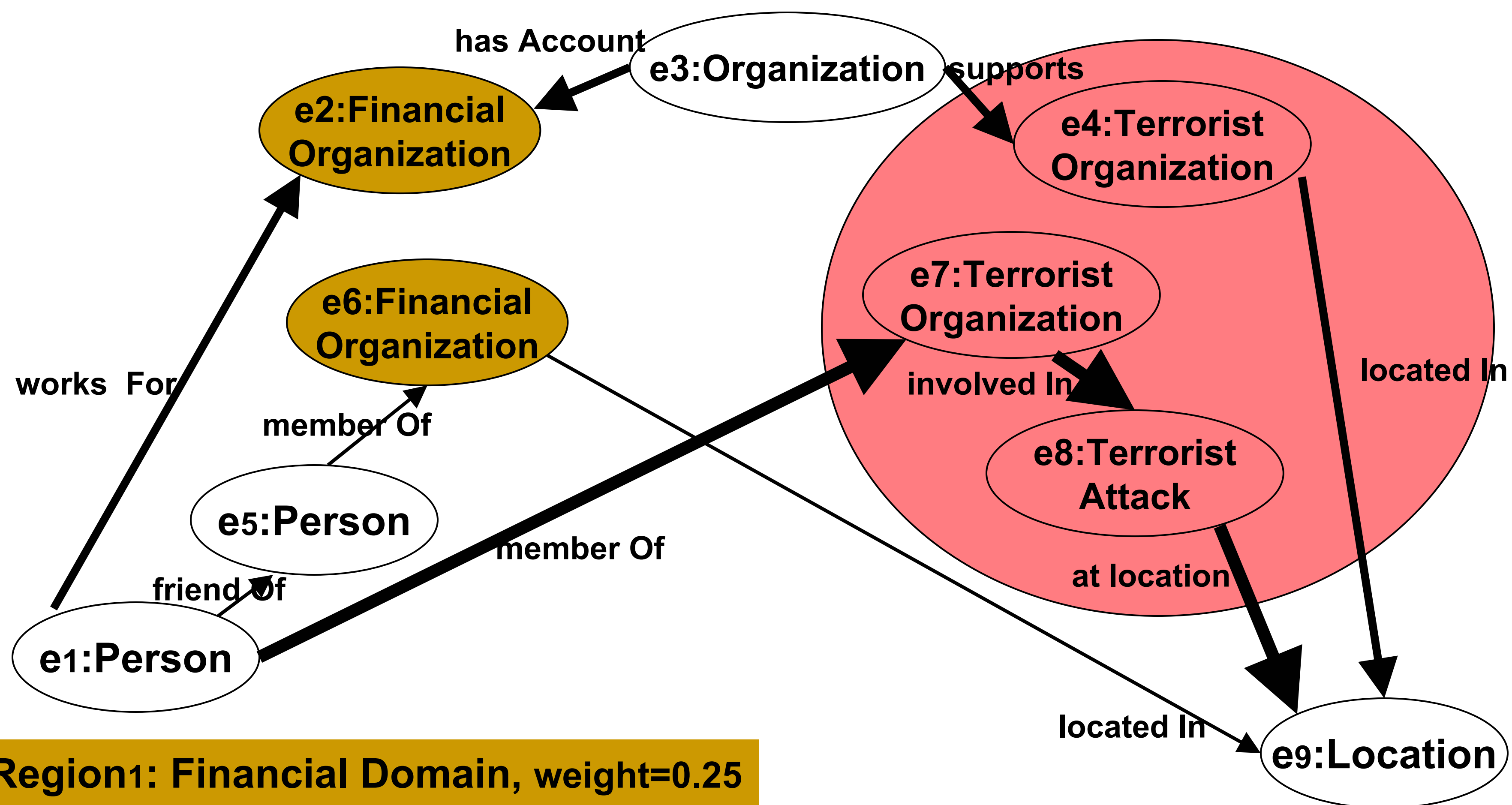
Context: What, Why, How?

- Context captures the users' interest to provide them with the relevant knowledge within numerous relationships between the entities
- Context => Relevance; Reduction in computation space
- By defining regions (or sub-graphs) of the ontology

Context Specification

- Topographic approach
  - *Regions* capture user's interest
    - *Region* is a subset of classes (entities) and properties of an ontology
  - User can define multiple *regions* of interest
    - Each *region* has a relevance weight

Context: Example



Region1: Financial Domain, weight=0.25

Region2: Terrorist Domain, weight=0.75

Context Issues

- Issues
  - Associations can pass through numerous *regions* of interest
  - Large and/or small portions of associations can pass through these *regions*
- Associations outside context *regions* rank lower

Context Weight Formula

- Refer to the entities and relationships in an association generically as the *components* in the associations
- We define the following sets, note $c \in R_i$ is used for determining whether the type of c (rdf:type) belongs to context *region* R_i:

$$X_i = \{c \mid c \in R_i \wedge c \in A\}$$

$$Z = \{c \mid (\forall i \mid 1 \le i \le n) c \notin R_i \wedge c \in A\}$$

where n is the number of *regions* A passes through

- X_i is the set of components of A in the i^{th} *region*
- Z is the set of components of A not in any contextual region

Context Weight Formula

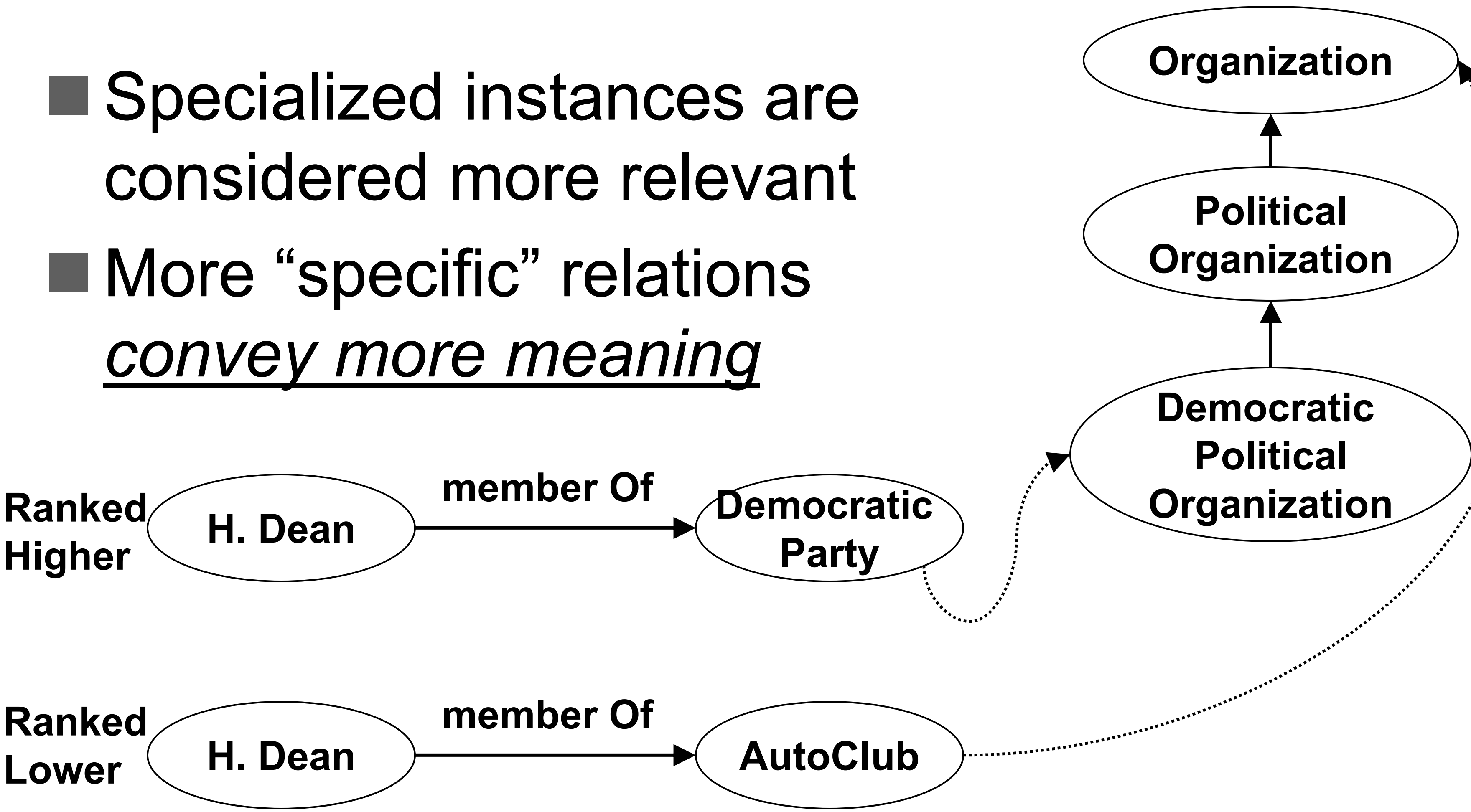
- Define the *Context* weight of a given association *A*, C_A, such that

$$C_A = \frac{1}{length(A)}\left(\left(\sum_{i=1}^{n}(w_{R_i} \times |X_i|)\right) \times \left(1 - \frac{|Z|}{length(A)}\right)\right)$$

  - *n* is the number of *regions A* passes through
  - *length(A)* is the number of components in the association
  - X_i is the set of components of *A* in the i^{th} *region*
  - *Z* is the set of components of *A* not in any contextual *region*

Subsumption

- Specialized instances are considered more relevant
- More “specific” relations *convey more meaning*



Subsumption Weight Formula

- Define the *component subsumption weight* (*csw*) of the i^{th} component, c_i, in an association *A* such that

$$csw_i = \frac{H_{c_i}}{H_{height}}$$

  - H_{c_i} is the position of component c_i in hierarchy *H*
  - H_{height} is the total height of the class/property hierarchy of the current branch
- Define the overall *Subsumption* weight of an association *A as*

$$S_A = \frac{1}{length(A)} \times \sum_{i=1}^{length(A)} csw_i$$

  - *length(A)* is the number of components in *A*

Trust

- Entities and relationships originate from differently trusted sources
  - Assign trust values depending on the source
  - e.g., Reuters could be more trusted than some of the other news sources
- Adopt the following intuition
  - The strength of an association is only as strong as its weakest link
    - *Trust* weight of an association is the value of its least trustworthy component

Trust Weight Formula

- Let t_{c_i} represent the *component trust weight* of the component, c_i, in an association, A
- Define the *Trust* weight of an overall association A as

$$T_A = \min(t_{c_i})$$

Rarity

- Many relationships and entities of the same type (rdf:type) will exist
- Two viewpoints
  - Rarely occurring associations can be considered more interesting
    - Imply uniqueness
    - Adopted from [3] where rarity is used in data mining relational databases
      - Consider rare infrequently occurring relationship more interesting

Rarity

- Alternate viewpoint
  - Interested in associations that are frequently occurring (common)
    - e.g., money laundering…often individuals engage in normal looking, common case transactions as to avoid detection
- User should determine which Rarity preference to use

Rarity Weight Formula

- Define the *component rarity* of the i^{th} component, c_i, in A as rar_i such that

$$rar_i = \frac{|M| - |N|}{|M|}$$, where

$M = \{res \mid res \in K\}$ (all instances and relationships in K), and

$$N = \{res_j \mid res_j \in K \wedge type(res_j) = type(c_i)\}$$

  - With the restriction that in the case res_j and c_i are both of type rdf:Property, the subject and object of c_i and res_j must be of the same rdf:type

- rar_i captures the frequency of occurrence of the rdf:type of component c_i, with respect to the entire knowledge-base

Rarity Weight Formula

- Define the overall *Rarity* weight, *R*, of an association, *A*, as a function of all the components in *A*, such that

  (a) $R_A = \frac{1}{length(A)} \times \sum_{i=1}^{length(A)} rar_i$

  (b) $R_A = 1 - \frac{1}{length(A)} \times \sum_{i=1}^{length(A)} rar_i$

  - where *length(A)* is the number of components in *A*
  - rar_i is *component rarity* of the i^{th} component in *A*
- To favor rare associations, **(a)** is used
- To favor more common associations **(b)** is used

Popularity

- Some entities have more incoming and outgoing relationships than others
  - View this as the *Popularity* of an entity
- Entities with high popularity can be thought of as *hotspots*
- Two viewpoints
  - Favor associations with popular entities
  - Favor unpopular associations

Popularity

- Favor popular associations
  - Ex. interested in the way two authors were related through co-authorship relations
    - Associations which pass through highly cited (popular) authors may be more relevant
- Alternate viewpoint…rank popular associations lower
  - Entities of type '*Country*' have an extremely high number of incoming and outgoing relationships
    - Convey little information when querying for the way to persons are associated through geographic locations

Popularity Weight Formula

- Define the *entity popularity*, p_i, of the i^{th} entity, e_i, in association *A* as

$$p_i = \frac{|pop_{e_i}|}{\max_{1 \le j \le n}(|pop_{e_j}|)} \quad \text{where} \quad typeOf(e_i) = typeOf(e_j)$$

  - 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
  - $\max_{1 \le j \le n}(|pop_{e_j}|)$ represents the size of the largest such set among all entities in the knowledge-base of the same class as e_i
- p_i captures the *Popularity* of e_i, with respect to the most popular entity of its same rdf:type in the knowledge-base

Popularity Weight Formula

- Define the overall *Popularity* weight, P, of an association A, such than

$$\textbf{(a)}\ P_A = \frac{1}{n} \times \sum_{i=1}^{n} p_i$$

$$\textbf{(b)}\ P_A = 1 - \frac{1}{n} \times \sum_{i=1}^{n} p_i$$

  - where n is the number of entities (nodes) in A
  - p_i is the *entity popularity* of the i^{th} entity in A
- To favor popular associations, **(a)** is used
- To favor less popular associations **(b)** is used

Association Length

- Two viewpoints
  - Interest in more direct associations (i.e., shorter associations)
    - May infer a stronger relationship between two entities
  - Interest in hidden, indirect, or discrete associations (i.e., longer associations)
    - Terrorist cells are often hidden
    - Money laundering involves deliberate innocuous looking transactions

Association Length Weight

- Define the *Association Length* weight, *L*, of an association *A* as

(a) $L_A = \dfrac{1}{length(A)}$

(b) $L_A = 1 - \dfrac{1}{length(A)}$

  - where *length(A)* is the number of components in the A
- To favor shorter associations, **(a)** is used, again
- To favor longer associations **(b)** is used

Overall Ranking Criterion

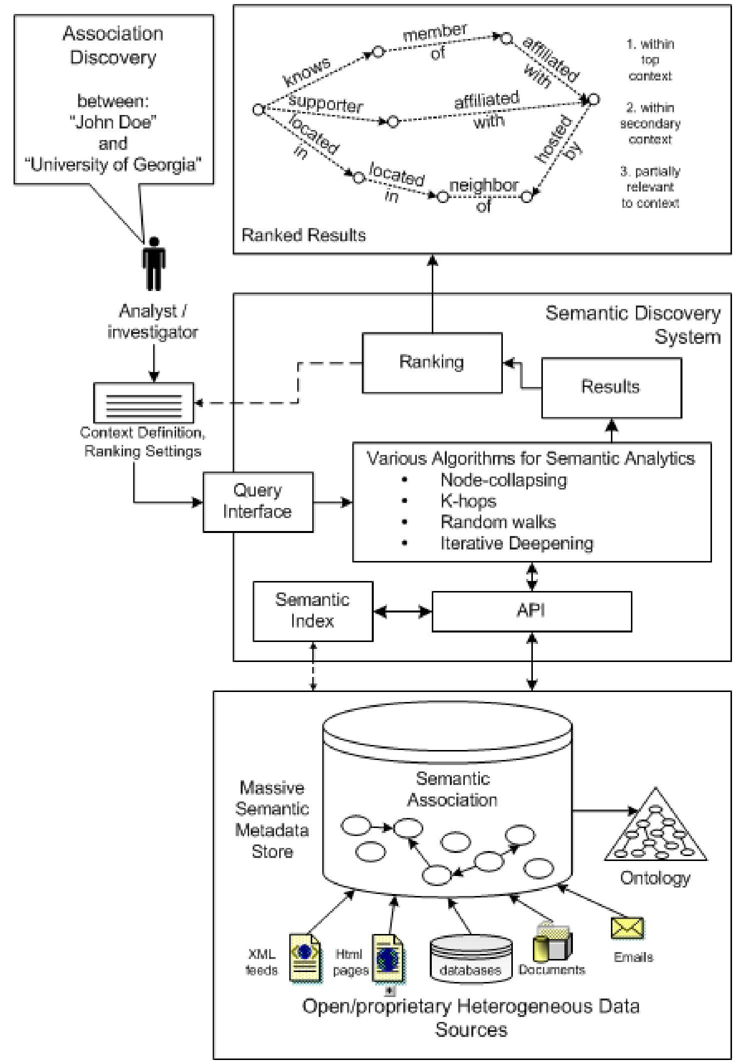
- Overall *Association Rank* of a Semantic Association is a linear function

$$\textit{Ranking Score} = \begin{aligned} & k_1 \times \textit{Context} + \\ & k_2 \times \textit{Subsumption} + \\ & k_3 \times \textit{Trust} + \\ & k_4 \times \textit{Rarity} + \\ & k_5 \times \textit{Popularity} + \\ & k_6 \times \textit{Association Length} \end{aligned}$$

  - where k_i adds up to 1.0
- Allows a flexible ranking criteria

System Implementation

- Ranking approach has been implemented within the LSDIS Lab's SemDIS[2] and SAI[3] projects



[2] NSF-ITR-IDM Award #0325464, titled 'SemDIS: Discovering Complex Relationships in the Semantic Web.'
[3] NSF-ITR-IDM Award #0219649, titled 'Semantic Association Identification and Knowledge Discovery for National Security Applications.'

System Implementation

- Native main memory data structures for interaction with RDF graph
- Naïve depth-first search algorithm for discovering Semantic Associations
- SWETO (subset) has been used for data set
  - Approximately 50,000 entities and 125,000 relationships
- SemDIS prototype[4], including ranking, is accessible through Web interface

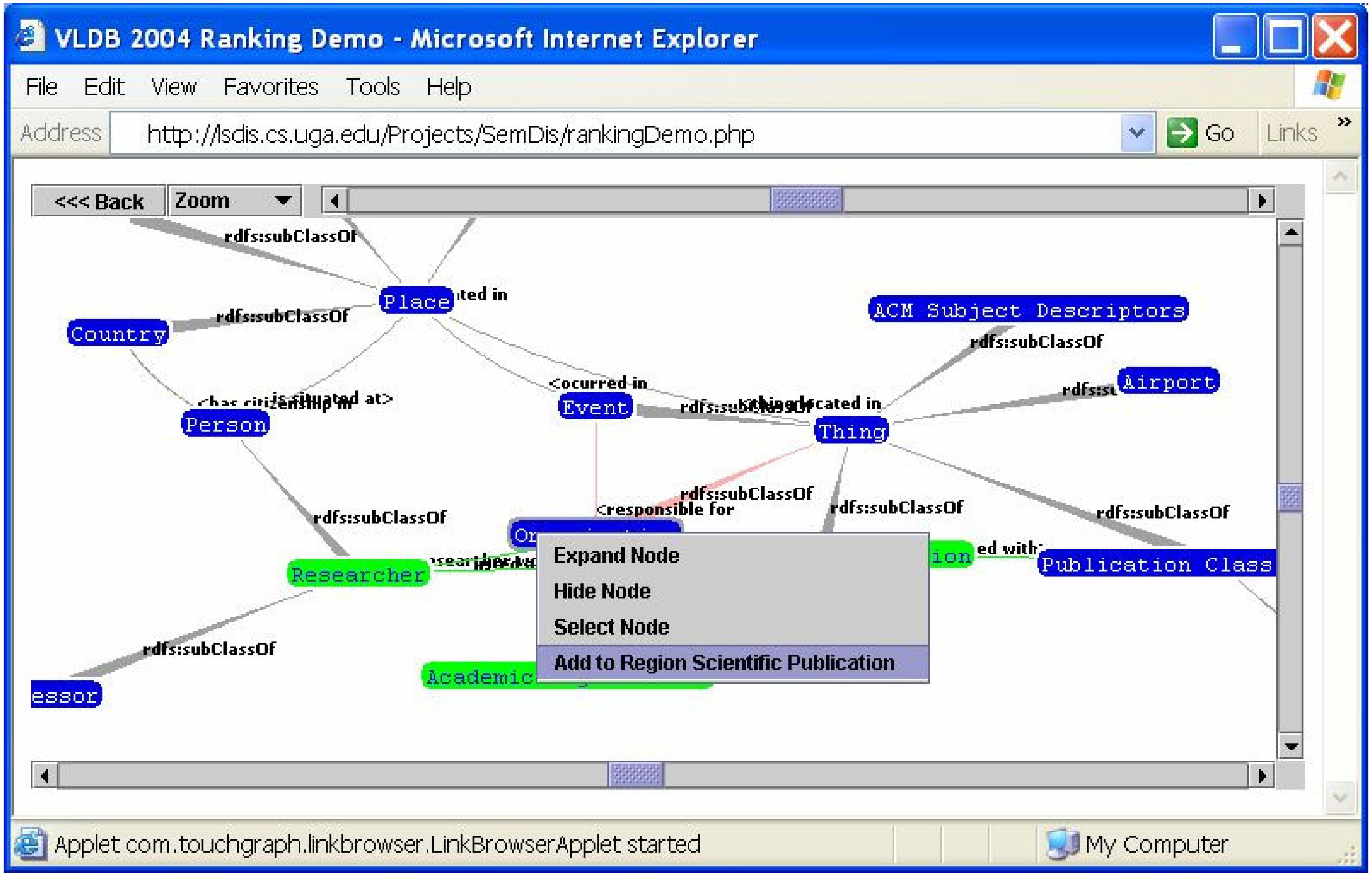
[4] SemDIS Prototype: http://vader.cs.uga.edu:8080/semdis/

Ranking Configuration

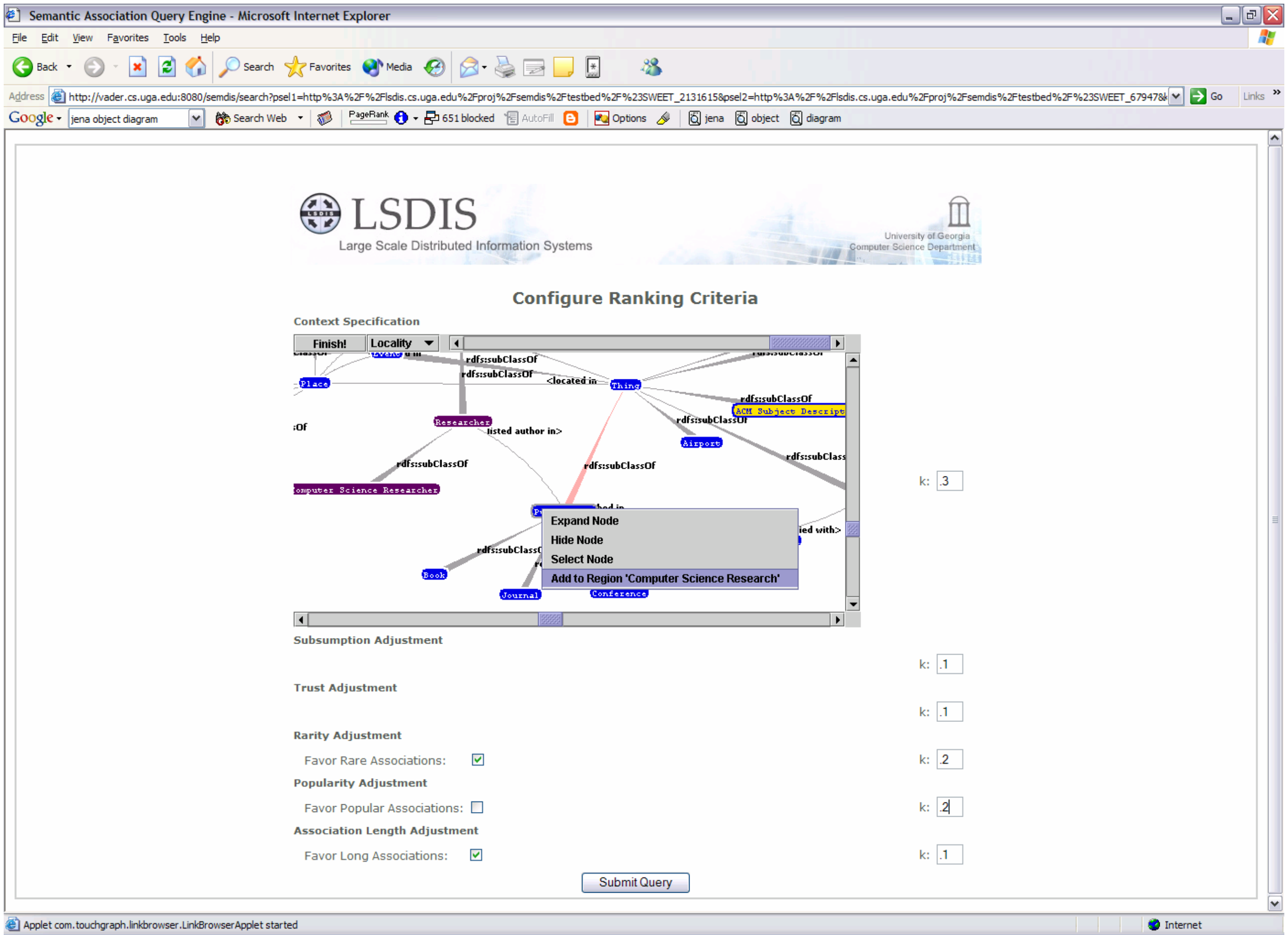
- User is provided with a Web interface that gives her/him the ability to customize the ranking criteria
- Use a modified version of TouchGraph[5] to define the query *context*
  - A Java applet for the visual interaction with a graph

[5]TouchGraph Homepage: http://www.touchgraph.com/

Context Specification Interface



Ranking Configuration Interface



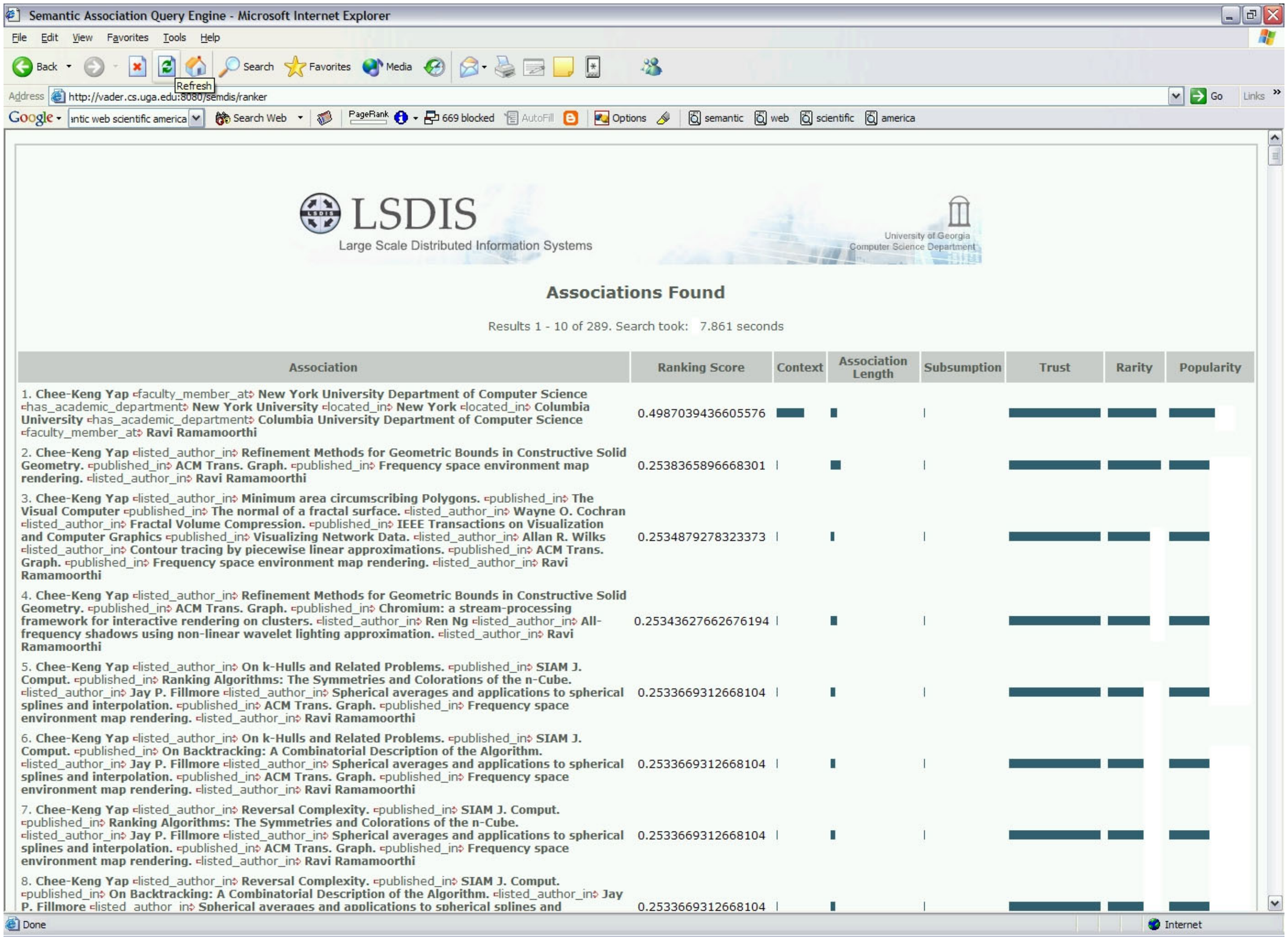
Ranking Module

- Java implementation of the ranking approach
- Unranked associations are traversed and ranked according to the ranking criteria defined by the user
- Ranking is decomposed into finding the context, subsumption, trust, rarity, and popularity rank of all *entities* in each association

Ranking Module

- Context, subsumption, trust, and rarity ranks of each *relationship* are found during the traversal as well
  - When the RDF data is parsed, rarity, popularity, trust, and subsumption statistics of both entities and relationships are maintained
  - Finding the context rank consists of checking which context regions, if any, each entity or relationship in each association belongs to

Ranked Results Interface



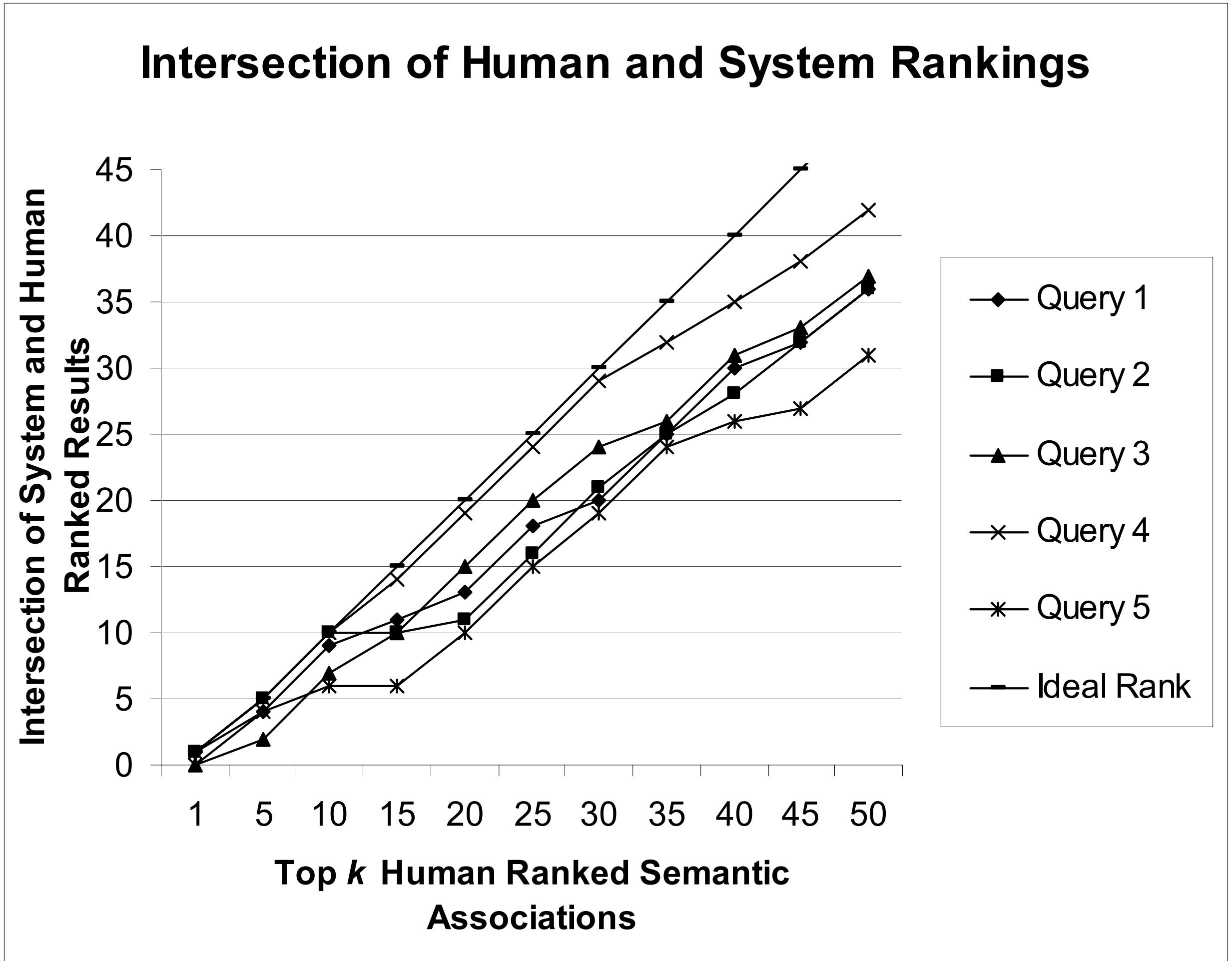
Ranking Evaluation

- Evaluation metrics such as precision and recall do not accurately measure the ranking approach
- Used a panel of five human subjects for evaluation
  - Due to the various ways to interpret associations

Ranking Evaluation

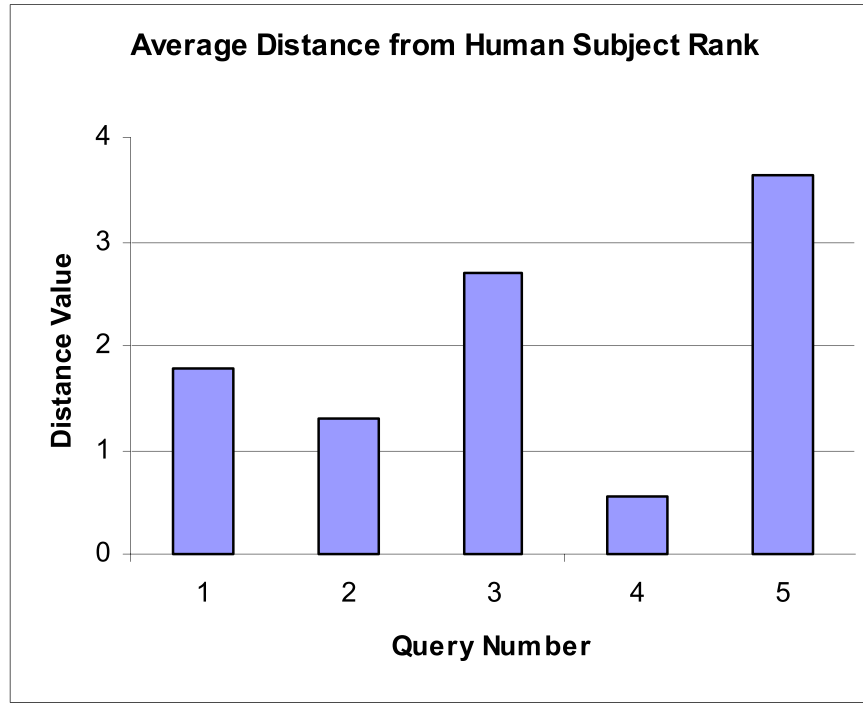
- Evaluation process
  - Subjects given randomly sorted results from different queries
    - each consisting of approximately 50 results
  - Provided subjects with the ranking criteria for each query
    - i.e., context, whether to favor short/long, rare/common associations, etc.
  - Provided type(s) of the components in the associations
    - To measure context relevance
  - Subjects ranked the associations based on this modeled interest and emphasized criterion

Ranking Evaluation (1)



Ranking Evaluation (2)

- Average distance of system rank from that given by subjects
  - Based on relative order



Conclusions

- Defined a flexible, query dependant approach to relevantly rank Semantic Association query results
- Presented a prototype implementation of the ranking approach
- Empirically evaluated the ranking scheme
  - Found that our proposed approach is able to capture the user's interest and rank results in a relevant fashion

Future Work

- *'Ranking-on-the-Fly'*
  - Ranks can be assigned to associations as the algorithm is traversing them
    - Possible performance improvements
- Use of the ranking scheme for the *Semantic Association* discovery algorithms (scalability in very large data sets)
  - Utilize context to guide the depth-first search
  - Associations that fall below a predetermined minimal rank could be discarded
- Additional work on context specification
- Develop ranking metrics for Semantic Similarity Associations

Publications

[1] **Chris Halaschek**, Boanerges Aleman-Meza, I. Budak Arpinar, Cartic Ramakrishnan, and Amit Sheth, A Flexible Approach for Analyzing and Ranking Complex Relationships on the Semantic Web, Third International Semantic Web Conference, Hiroshima, Japan, November 7-11, 2004 (submitted)

[2] **Chris Halaschek**, Boanerges Aleman-Meza, I. Budak Arpinar, and Amit Sheth, Discovering and Ranking Semantic Associations over a Large RDF Metabase, 30th Int. Conf. on Very Large Data Bases, August 30 September 03, 2004, Toronto, Canada. Demonstration Paper

[3] Boanerges Aleman-Meza, **Chris Halaschek**, Amit Sheth, I. Budak Arpinar, and Gowtham Sannapareddy, SWETO: Large-Scale Semantic Web Test-bed, International Workshop on Ontology in Action, Banff, Canada, June 20-24, 2004

[4] Boanerges Aleman-Meza, **Chris Halaschek**, I. Budak Arpinar, and Amit Sheth, Context-Aware Semantic Association Ranking, First International Workshop on Semantic Web and Databases, Berlin, Germany, September 7-8, 2003; pp. 33-50

References

[1] ANYANWU, K., AND SHETH, A. 2003. r-Queries: Enabling Querying for Semantic Associations on the Semantic Web. In Proceedings of the 12th International World Wide Web Conference (WWW-2003) (Budapest, Hungary, May 20-24 2003).

[2] BERNERS-LEE, T., HENDLER, J., AND LASSILA, O. 2001. The Semantic Web. Scientific American, (May 2001)

[3] LIN, S., AND CHALUPSKY, H. 2003. Unsupervised Link Discovery in Multi-relational Data via Rarity Analysis. The Third IEEE International Conference on Data Mining.

Questions & Comments

Thank You