Enhancing the Power of the Internet: Studies in Fuzziness and Soft Computing, Masoud Nikravesh, Ben Azvin, Ronal Yager and Lotfi A. Zadeh, Eds., Springer-Verlag.

Relationships at the Heart of Semantic Web: Modeling, Discovering, and Exploiting Complex Semantic Relationships

AMIT SHETH[1,3], I. BUDAK ARPINAR[1], AND VIPUL KASHYAP[2]

[1] *LSDIS Lab, Computer Science Department, University of Georgia*

[2] *National Library of Medicine,* [3] *Semagix, Inc.*

amit@cs.uga.edu, budak@cs.uga.edu, kashyap@nlm.nih.gov

Abstract. The primary goal of today's search and browsing techniques is to find relevant documents. As the current web evolves into the next generation termed the Semantic Web, the emphasis will shift from finding documents to finding facts, actionable information, and insights. Improving ability to extract facts, mainly in the form of entities, embedded within documents leads to the fundamental challenge of discovering relevant and interesting relationships amongst the entities that these documents describe. Relationships are fundamental to semantics—to associate meanings to words, terms and entities. They are a key to new insights. Knowledge discovery is also about discovery of heretofore new relationships. The Semantic Web seeks to associate annotations (i.e., metadata), primarily consisting of based on concepts (often representing entities) from one or more ontologies/vocabularies with all Web-accessible resources such that programs can associate "meaning with data". Not only it supports the goal of automatic interpretation and processing (access, invoke, utilize, and analyze), it also enables improvements in scalability compared to approaches that are not semantics-based. Identification, discovery, validation and utilization of relationships (such as during query evaluation), will be a critical computation on the Semantic Web.

Based on our research over the last decade, this paper takes an empirical look at various types of simple and complex relationships, what is captured and how they are represented, and how they are identified, discovered or validated, and exploited. These relationships may be based only on what is contained in or directly derived from data (direct content based relationships), or may be based on information extraction, external and prior knowledge and user defined computations (content descriptive relationships). We also present some recent techniques for discovering indirect (i.e., transitive) and virtual (i.e., user-defined) yet meaningful (i.e., contextually relevant) relationships based on a set of patterns and paths between entities of interest. In particular, we will discuss modeling, representation and computation or validation of three types of complex semantic relationships: (a) using predefined multi-ontology relationships for query processing and

corresponding the issue of "loss of information" investigated in the OBSERVER project, (b) ρ (Rho) operator for semantic associations which seeks to discover contextually relevant and relevancy ranked indirect relationships or paths between entities using semantic metadata and relevant knowledge, and (c) IScapes which allows interactive, human-directed knowledge validation of hypothesis involving user-defined relationships and operations in a multi-ontology, and multi-agent InfoQuilt system.

Representing, identifying, discovering, validating and exploiting complex relationships are important issues related to realizing the full power of the Semantic Web, and can help close the gap between highly separated information retrieval and decision-making steps.

Keywords: Complex relationship, Semantic Annotation, Multi-ontology Query Processing, Information Landscape, Semantic Association, Semantic Web, Semantic Relationship Validation, Semantic Relationship Discovery

1. Introduction

Most Internet users today find information in one of two ways – either by browsing the information space or through the use of a search engine. Browsing is completely under the control of the user but requires choosing a good directory that has organized the document space, combined with user's constant attention and decision-making. Systems based on search engines perform essentially the task of delivering a document based on keywords or key phrases. Some search engines, such as Google, use heuristics and statistics to improve ranking for a generic user, but that only seeks to improve document retrieval for most users. None of these approaches attempts to get at the user's underlying intentions or information goals. And none give new insights related to user's information needs. This is readily evident from their results – most of the retrieved documents are either irrelevant unless the search objective is relatively straightforward (e.g., home page of a person or specific document posted at a well respected source), or contain the information buried in a morass of other data. A user must decide which of the retrieved documents are relevant or within his information need context, and then use his mental model of the information sought to "process" the documents to obtain the relevant information. This is a very serious and as yet unsolved problem, as evidenced by the fact that practically all of today's technical efforts in search engine, content management, and other technologies are geared towards dealing with data overload, which leads to

information starvation (the inability to find useful and actionable information from massive amounts of data).

Significant past research has been conducted in managing heterogeneous data, and providing interoperability and integration of information systems so that data can be shared, collectively accessed, and processed [Sheth98]. This has been a long process, with earlier research dating to the late 1970, going through the architectures for federated databases [Sheth90], mediators [Weiderhold92] and information brokering systems [Kashyap00]. With the ability to access and share all forms of data, now we have the familiar challenge of data overload.

We believe the a more fundamental challenge is to make decisions or take actions based on data than finding relevant documents – an objective that a new generation of content management systems subscribe to, and the one most of today's search and browsing techniques fail to address. One step towards gaining this capability is to discover relevant and interesting relationships amongst the entities that these documents describe. These relationships are the basis of analysis, and underpin the semantics of the data. We face several challenges in meeting this task. One reason is that the data retrieval (i.e., "search") phase is not geared towards dealing with relationships. For instance, if a search for "data" results in a large numbers of irrelevant documents, any technique for finding relationships will generate a correspondingly much larger (perhaps by an order of magnitude) number of irrelevant, and useless relationships. As the adage says, every one is related by only six degrees of separation!

For computing (identifying, discovering or validating) relationships, what we need is very different from data mining, at least as it has been traditionally understood in terms of grouping or market basket type analysis through the discovery of association rules. Data mining techniques are typically based on statistics and look for patterns that are already present in the data. Moreover, the patterns are sought at a syntactic level, and do not take into consideration the meaning of the data. They are typically not easily extendable to look for the types of relationships that are meaningful to humans or to the software agent performed target information processing tasks, and they are not based on the semantics of the underlying data. The clustering and machine learning techniques in themselves will similarly not be sufficient. However, computing complex relationships

require new forms of processing data and relevant knowledge, and associated techniques of creating and maintaining a variety of relationships. Instead of relying on data alone, they utilize a broad variety of domain knowledge, and context, which enables scalability by ignoring irrelevant information, and knowledge.

Developing a system focused around finding semantic relationships rather than documents is challenging for several reasons. Each document may describe (and hence be annotated with) many entities. The number of relationships or paths connecting entities directly or through a Knowledge Base (KB), however, is vastly larger. Whether seen as a graph theoretic or deductive logic problem, many approaches for computation are not tractable, let along scalable. Furthermore, imposing constraints that only relevant or interesting relationships are discovered may add to the complexity.

This chapter significantly borrows from our benefits from past efforts including:

- Research in semantic interoperability and integration of heterogeneous data [Kashyap96], partially performed in InfoHarness [Shah99], and its follow on VisualHarness [Shah97], and VideoAnywhere projects [Bertram98],
- Semagix's Semantic Content Organization and Semantic Engine (SCORE) technology [Sheth02a, Hammond02] partially based on technology licensed from UGA, and based on above projects,
- UGA's research on human-directed knowledge discovery in InfoQuilt project [Sheth02b], multi-ontology query processing in OBSERVER project [Mena00], and the on-going project on Semantic Association discovery [Anyanwu02].

In this paper, we do not attempt to present a comprehensive taxonomy of relationships, nor do we survey all relevant literature. Rather our treatment is empirical and involves a review of semantic relationship computation and use in various research systems we have worked on during the last decade. Section 2 provides and overview and a partial classification of challenges in dealing with relationships. In Section 3, we start with *identification of simple semantic relationships* based on a large knowledge base in a state of the art commercial system SCORE based on technology transfer from our academic research. Section 4 discusses as examples of *semantic relationship discovery*. It introduces the concept of complex relations called Semantic Associations, and some

preliminary thoughts on computing a ranked list of these associations using a context. In Section 5, we discuss IScapes, user-defined complex relationships, and their *validation* in the InfoQuilt system as a way to support user directed knowledge discovery. Section 6 provides an example of query *evaluation involving semantic relationships*. We discuss use of inter-ontology relationships in OBSERVER's multi-ontology query processing, and the corresponding effort in computing information loss. We conclude with Section 7.

2. Classification of Complex Relationships

The questions of if and how two or more entities relate to each other are both technical and philosophical questions. Yet, these are the essential questions to exploit to discover new, interesting, and useful relations across entities in diverse domains including national security, life sciences, and economics. On what dimensions should a study of different kinds of relationships be organized? One dimension of relationship is whether it is based on explicit, precise or exact knowledge, or that it is based on imprecise or approximate knowledge (such as one based statistical and probabilistic measures). As an enhancement of this perspective, we propose three dimensions along which it might be useful to organize such a study: (a) the information content captured by a relationship; (b) various ways of representing a relationship; and (c) methodologies for computing (i.e., identifying, discovering, and validating), and exploiting the various relationships.

2.1 A Taxonomy of Relationships Based on the Information Content

Metadata has been used to describe data, document or content [Boll98]. Patterned after the classification used for metadata [Kashyap95], we classify the relationships as follows:

- **Content Independent Relationships:** These types of relationships are typically independent of the content and are an artifact of the organization of content on a computer system due to reasons of organization, performance, scalability, etc., e.g., two documents may be related to each other by virtue of them being stored on the same server or file system, or the relationship between a document and it's date of modification, etc.
- **Content Dependent Relationships:** These capture the relationships between two entities based on the either the information content they refer to in the real world or

based on some representation of it thereof. Various types of content dependent relationships are as follows:

- **Direct Content Dependent Relationships:** These types of relationships typically depend on some representation of the information content to which the entities refer to and are directly computed from them. It may be noted that some of these relationships might be fuzzy in nature. For example, the relations between two entities being mentioned in the same paragraph and spatial locations of two objects in an image suggest crisp relations, whereas the similarity between two documents in a vector space is a fuzzy measure.
- **Content Descriptive Relationships:** These types of relationships are based on the information content, which the entities refer to in the real world. These are typically not computable directly from the representation of the information content and help of additional resources such as taxonomies, and ontologies along with heuristic algorithms may be used to compute these relationships. For example, the fact that an entity X is the CEO of a company Y is computed based on the existence of an ontology that models businesses (which specifies the relationship "CEO") and heuristic document processing algorithms (which discover the relationship) applied to relevant documents. These relationships are typically viewed as crisp as some thresholding techniques are applied to the heuristic algorithms, whereas they are in reality fuzzy and reflect a probability of the person X being the CEO of a company Y. These relationships might associate entities within a domain (**intra-domain relationships**) or across multiple domains (**inter-domain relationships**). An informal (and incomplete) (sub-) classification of this type of relationship is as follows:
  - **Direct Semantic Relationships:** These are direct intra-domain relationships between two documents or entities, e.g., an HREF link annotated with semantic information (Figure 1.a), Intel *is-a-competitor-of* Motorola (Figure 1.b). Examples of these are discussed in the SCORE system in Section 3.
  - **Complex Transitive Relationships:** Remzi and Dick are associated with each other because they are linked to the same terrorist organization through

their financial transaction (Figure 1.c). This . These type of intra-domain relationships are captured using the **ρ** operator discussed in Section 4.

- **Inter-domain Multi-ontology Relationships:** Some relationships span across multiple domains and are typically represented as inter-ontology relationships across multiple ontologies. This type of relationships is discussed in the context of the OBSERVER System in Section 6.
- **Semantic Proximity Relationships:** Two entities may have a semantic proximity or similarity that cannot be completely represented using crisp relationships. They may either be represented using a semantic proximity function associated with a relationship or depend on fuzzy predicates such as "close-enough" (Figure 1.e illustrates a similarity relations between two events)." Furthermore, they may be user defined (Figure 1.d). These types of relationships are discussed in the context of IScapes in Section 5.

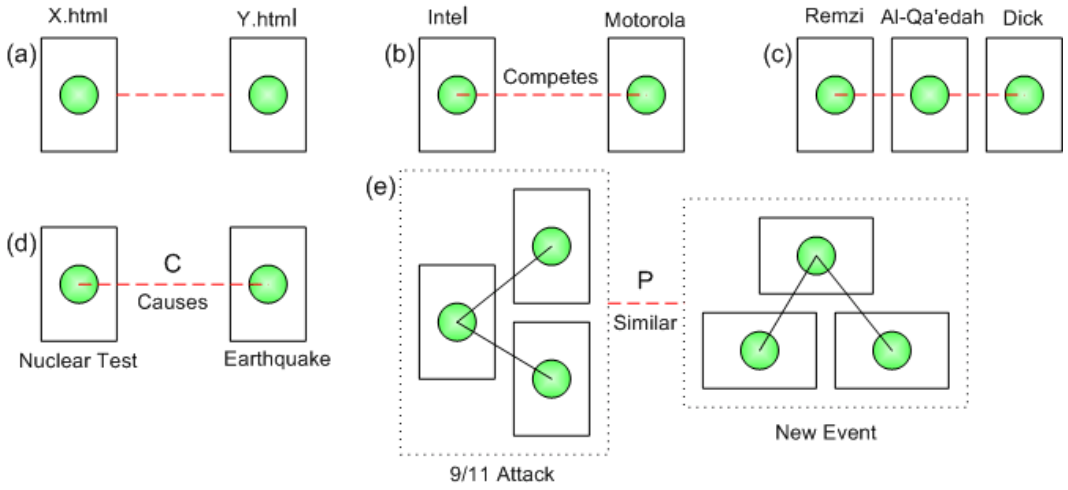


Figure 1: Different types of structural composition of relationships

2.2 Representation of Relationships

A fundamental representation of a relationship between two concepts is a mathematical structure denoting it as a set mapping between the instances belonging to the two concepts. These mappings might be characterized along the following dimensions:

- **Arity:** Typically binary relationships are of most interest, but relationships can be of arbitrary arity, i.e., we could have 3 or more concepts participating in a relationship.

- **Cardinality:** These constraints are characterized in one of the following ways: 1-1, many-1, 1-many, or many-many. A more generalized way of representing these cardinality constraints is using a pair of numbers that specify the minimum and maximum number of times an instance of a concept can participate in a relationship. This is a very useful technique for n-ary relationships and also captures partial participation of concepts in relationships. 1-1 and many-1 relationships are functions which can be exploited in various ways.

- **Direct v/s Transitive Relationships:** Some entities might be directly related to each other via their participation in a common relationship, or might be related transitively to each other via a chain of relationships.

- **Crisp vs. Fuzzy:** Most of the current modeling approaches view relationships as crisp, i.e., for an n-ary relationship, instances of n concepts are either part of a relationship or not (e.g., is-a, part-of relations). In the case of fuzzy knowledge [Zadeh65], the extension of a relationship may be viewed as a joint probability distribution on the concepts participating in a relationship. For example semantic similarity (i.e., proximity) between two entities is an example for fuzzy relations.

- **Properties vs. Relations:** Properties are special relationships where the ranges of a relationship are values of a data type (e.g., dates, age) as opposed to instances of a concept.

- **Structural Composition:** Relationships can either be composed (if they are functional in nature) or combined using join operations to create new relationships and associations based on existing relationships.

Most frequently occurring relationship is that of hypertext link (HREF). One attempt to make it more meaningful was the proposal for MetadataMetadata Reference Link (MREF) [Shah98] that associated metadata represented in RDF to HREF. This metadata

provided further semantics to otherwise a hypertext link without any information that a machine can use to understand what it is about (Figure 1.a).

Most modeling approaches whether they are graphical in nature, e.g., EER, UML diagrams or use object models and XML markup models, e.g., OMG object model, OKBC, DAML+OIL represent the fundamental structures described above using various modeling (graphical or markup) primitives which can be combined together using various (graphical, hierarchical or symbolic) constructors.

2.3 Computation and Exploitation of Relationships

Four main computations that can be performed to manage and exploit relationships are as follows.

- **Identify**: This is the process by which a relationships whose semantics is known and understood (e.g., via its representation in a domain specific ontology), and computation is directed towards identifying the presence of the relationship within a document or any other piece of data. We present an example of this in the discussion of the SCORE System (Section 3).
- **Discover**: This is the process by which we search for patterns among content or resources, within a semantic model or an ontology to discover new relationships. Other approaches of discovering new relationships might involve text mining operations. We present Rho operators that can search for patterns in an ontology and propose new relationships (Section 4).
- **Validate**: This is the process by which IScapes representing knowledge discovery hypothesis, possibly involving complex relationships and fuzzy operators (e.g., near to, same as), are validated by information gathering and analysis over a collection of heterogeneous data sources (Section 5).
- **Evaluate**: In the process of computing a given relationship, it may be noted that it may only be possible to estimate it, giving rise to uncertainty and confidence intervals. We discuss multi-ontology query processing in the OBSERVER System (Section 6), which computes the equivalence relationship between an information request and the answer (possibly spanning multiple ontologies), with the associated precision and recall measures.

3. Ontology Driven Relationship Identification: Example of the SCORE Technology

In this section, we discuss an example of identifying an instance of relationship based on a document analysis. The existence of a relationship is already know to the system, for example as part of an ontology, so the relationship is identified based on occurrence of entities in the relevant context in the document.

Identification of such a relationship is exemplified by a commercial semantic technology based on prior academic research. SCORE is a commercial Semantic Content Organization and Retrieval Engine [Sheth02a, Hammond02]. Semantic underpinning in SCORE is provided by an ontology with a definitional component (called World Model) and assertional component (called Knowledge Base – KB). In SCORE, through the use of automatic classification and contextually relevant ontology (i.e., relevant part of ontology including the assertions), domain specific metadata can be extracted from a document, enhancing the meaning of the original and allowing it to be linked with contextually heterogeneous content from multiple sources. In this way, relations between the entities, which are not explicitly evident in a single document, can be revealed. We call these types of one-to-one relations between the entities simple indirect relations.

The identification of indirect semantic relations between the entities and its use in document enhancement is illustrated in Figure 2. First, the classification technology determines the category for a document. This determines the domain of discourse, or relevant ontology, e.g., business ontology (or a relevant part of an ontology, e.g., equity market part of entertainment ontology). Then semantic metadata particular to the domain is targeted and extracted. This includes specific named entity types of interest in the category (such as “CEO” in “Business,” “Downgrade” in “Equity Markets”, or “SideEffects” in “Pharmacology”) as well as category specific, regular expression-based knowledge extraction. This domain-specific metadata can be regarded as semantic metadata, or metadata within context. The automatic extraction of semantic metadata from documents which have not been previously associated with a domain is a unique feature of SCORE. In essence, this transports the document from the realm of text and

mere syntax to a world of knowledge and semantics in a form that can be used for semantic computation.

An example is illustrated through a Web document in Figure 3. In the Figure, BEA Systems, Microsoft and PeopleSoft all engage in the "competes with" relationship with Oracle. When entities found within a document have relationships based on a known ontology, we refer to the relationships as "direct relationships." Some of the direct relationships found in this example include: HPQ identifies Hewlett-Packard Co.; HD identifies The Home Depot; Inc.; MSFT identifies Microsoft Corp.; ORCL identifies Oracle Corp.; Salomon Smith Barney's headquarters is in New York City; and MSFT, ORCL, PSFT, BEAS are traded on Nasdaq.

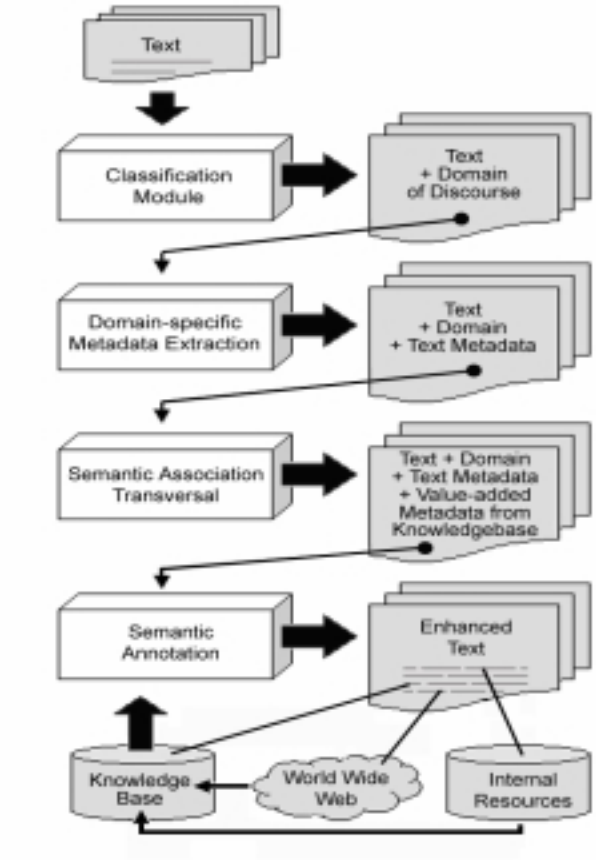


Figure 2: Semantic Document Enhancement in SCORE System

Not all of the associated entities for an entity found in the text will appear in the document. Often, the entities mentioned will have one or more relationships with another common entity. In this case, some examples include: HPQ and HD are traded on the NYSE; BEAS, MSFT, ORCL and PSFT are components of the Nasdaq 100 Index;

Hewlett-Packard and PeopleSoft invested in Marimba, Inc., which competes with Microsoft; BEA, Hewlett-Packard, Microsoft and PeopleSoft compete with IBM, Sun Microsystems and Apple Computer.

The use of semantic associations allows entities not explicitly mentioned in the text to be inferred or linked to a document. This one-step-removed linking is referred to as "indirect relationships." The relationships that are retained are application specific and are completely customizable. Additionally, it is possible to traverse relationship chains to more than one level. It is possible to limit the identification of relationships between entities within a document, within a corpus across documents or allow indirect relationships by freely relating an entity in a document with any known entity in the SCORE KB.

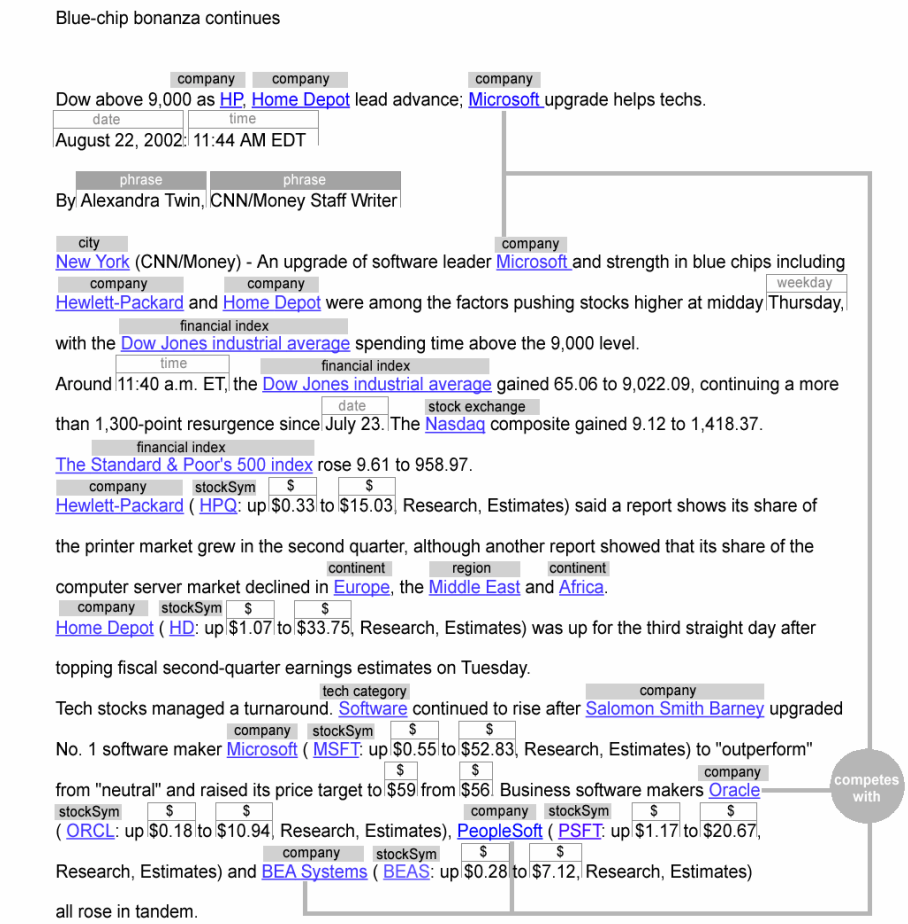


Figure 3: When SCORE recognizes an entity, knowledge about its entity relationships to other entities becomes available through relevant (parts of) ontology based on context provided by automatic classification

Indirect relationships provide a mechanism for producing value-added semantic metadata. Each entity in the KB provides an opportunity for rich semantic associations. As an example, consider the following:

Oracle Corp.	
Sector:	Computer Software and Services
Industry:	Database and File Management Software
Symbol:	ORCL
CEO:	Ellison, Lawrence J.
CFO:	Henley, Jeffrey O.
Headquartered in:	RedWood City, California, USA
Manufactured by:	8i Standard Edition, Application Server, etc.
Subsidiary of:	Liberate Technologies and OracleMobile
Competes with:	Agile, Ariba, BEA Systems, Informix, IBM, Microsoft, PeopleSoft and Sybase

This represents only a small sample of the sort of knowledge in the SCORE KB. Here, the ability to extract from disparate resources can be seen clearly. The "Redwood City" listed for the "Headquartered in" relationship above, has the relationship "located within" to "California," which has the same relationship to the "United States of America." Each of the entities related to "Oracle" are also related to other entities radiating outward. Each of the binary relationships has a defined *directionality (some may be bi-directional)*. In this example, *Manufactured by* and *Subsidiary of* are marked as *right-to-left* and should be interpreted as "8i Standard Edition, Application Server, etc. are *manufactured by* Oracle" and "Liberate Technologies and OracleMobile are *subsidiaries of* Oracle." SCORE can use these relationships to put entities within context.

When a document mentions "Redwood City," SCORE can add "California," "USA," and "North America." Thus, when a user looks for stories that occur in the United States or California, a document containing "Redwood City" can be returned, even though the more generalized location is not explicitly mentioned. This is one of the capabilities a keyword-based search cannot provide, where the information implicit in the text is revealed and can then be linked with other sources of content.

4. ρ Operator for Semantic Associations: Example of Semantic Relationship Discovery and Ranking

In this example, we will discuss an ongoing research on discovery of complex semantic relationships in the Semantic Web. Many applications in analytical domains such as national security and business intelligence require a more complex notion of relationships than the simple direct relationships between the entities, of the types discussed in Section 3. For example, in the light of the recent breach of flight security, it has become pertinent to enable airport security agents are able to ask questions like, what *important relationships* exist between Passenger X and Passenger Y? A new relationship may emerge because of complex transitive relations connecting these two persons. Furthermore, the notion of importance depends primarily on the context, which in this case is the assessment the risk of flight based on passenger associations. In this scenario, it is not possible to encode all the relevant relationships as rules, because they are not usually known; yet they can be discovered through an analytical process. In general, the relevant relationships emerge as a set of connections or various interesting patterns of connections between the entities. As an example, consider some passengers who are the nationals of the same country, and purchased their tickets using the same credit card, even though they do not have a known family relationship, and furthermore one of them is on the FBI watch-list. Because different domains may have different notions of relationships, in other words, what kind of connections constitute a relationship, it may be useful to use domain-specific ontology to guide the search for semantic relations.

Semantic relations in the most basic sense involve evaluating a set of contextually relevant paths of relations from one entity to another. By evaluating such paths we may identify relations based on connectivity or similarity of paths. This allows us to analyze sequences of binary relationships instead of just single binary relationships, and manipulate these sequences to find similar entities as well as entities that may be connected, albeit not directly. This technique is different from data mining that uses statistical techniques to find co-occurrence relationships between predicates based on patterns in data.

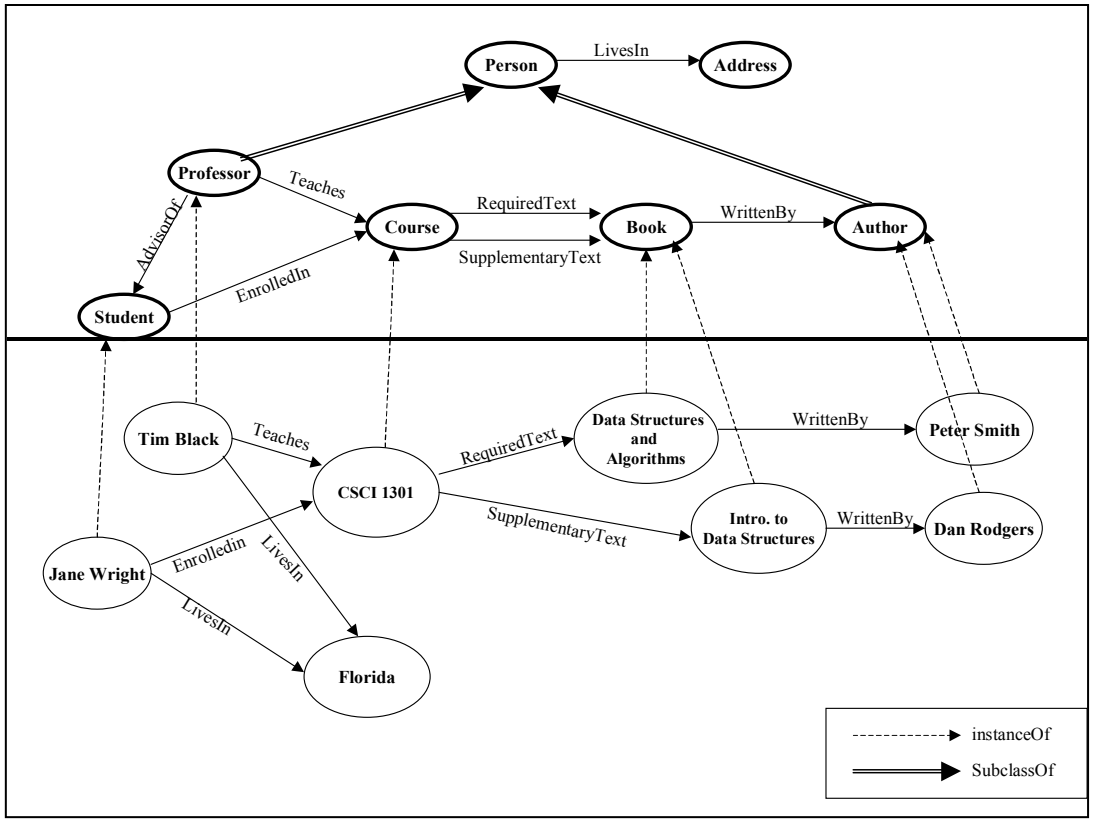


Figure 4: An Example Ontology and Knowledge Base

We will illustrate the notion of complex semantic relations, termed semantic associations through a pedagogical example [Anyanwu02]. Figure 4 shows a simple ontology containing information about Professors, Students, Courses, Books, and Book Authors. The top part of the figure shows the descriptional part of ontology which contains the entity types (i.e., classes) depicted as nodes, and the domain specific relationships between entity types are illustrated by single-lined arcs. Entity types may also be related by special relationships such as a *subclassOf* relationship denoted by a double-lined arc. The bottom part of the Figure shows assertional component of the ontology, i.e., instances of the classes, and dotted lines illustrate *instanceOf* relations. In this simplified example, semantic relations include the following: Tim Black can be said to be associated with Peter Smith because he *Teaches* a course CSCI1301 that has as its text a book *WrittenBy* by Peter Smith. Also, Peter Smith and Dan Rodgers are associated in that they

both are authors of the books that are used as textbooks in a particular course. These two relations are slightly different because the first involves a directed path between entities, while the second involves an undirected path. Discovery of more complex relations between two entities may require checking the semantic similarity between the sub-graphs of a knowledge base involving these entities; furthermore, the similarity checking may require custom defined computations (e.g., two Professors can be related because they use similar investigative methods in two different scientific experiments). Another dimension is aggregation of entities and associations to find more meaningful group associations than individual links connecting the entities of interest (i.e., discovery of association structures vs. individual associations). Some example association types we have been addressing are illustrated in Figure 5.

The associations 1, and 2-4 are examples of direct and transitive links between two entities, respectively. For example, 3 may represent a semantic relation between two Professors whose books are used for the same course. Entities that have a common successor and predecessor can be represented by 3 and 4 respectively. The arbitrary combinations of these link types may result in more complex relations as illustrated in 5. An example might be two Professors whose projects are funded by two different agencies having a common manager. In general, two entities having an un-directed path between them can be associated in varying degrees according to the path length (and possibly path strength).

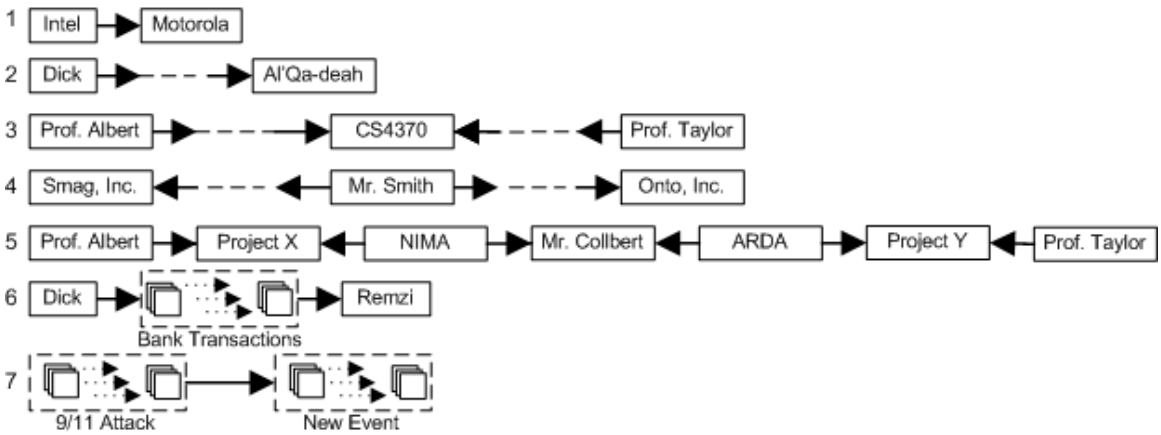


Figure 5: Some Complex Semantic Association Types

Association 6 represents an aggregation of several associations, which is more meaningful and interesting than the individual member associations. For example if a person makes some periodic deposits to another person's account in an overseas bank the aggregation of the links for individual transactions may provide a clue for a money laundering operation. Similarly aggregation of certain entities into groups (i.e., *spheres of semantics*) and investigating group associations may yield more interesting results. In 7, a semantic similarity relation between two events exists, because both of them contain a "similar" set of associations. In another example, two terrorist organizations can be related if the set of associations representing their operation styles resemble each other.

Assigning more weights to certain entities and relations and favoring discovery process for visiting these entities and associations can improve the efficiency of the semantic association discovery. For example, if the entity of interest is a certain person, it can be given more weight and relationship discovery may focus on the paths passing through this person. Another technique involves specification of relevant context by identifying certain regions in the ontologies and knowledge base to limit the discovery in traversing transitive links.

If there are too many associations between the entities of interest, then analyzing them and deciding which ones are actually useful might be a burden on a user. Therefore ranking these new relations in accordance of the user's interest is an essential task. In general, a relation can be ranked higher if it is a relatively original (e. g., previously unknown), more trustworthy, and useful in a certain context.

4.1. A Comparative Analysis of Semantic Relation Discovery and Indexing

As the emergence of the Semantic Web gathers momentum, it is imperative to propagate the novel ideas of representing, correlating, and presenting the wealth of available semantic information. A traditional search engine with the associated inverted keyword index (or similar) has served the Web community quite well to a certain point. However, to make searching more precise, a typical search engine must evolve to incorporate a new query language, capable of expressing semantic relationships and conditions imposed on them.

Our KB contains *entities* as well as *relationships* connecting the entities. An entity has a name and a classification (type). A relationship has a name and a vector of entity classifications, specifying the types of entities allowed to participate in the relationship. Both entity classifications and relationships will be organized into their respective hierarchies. The *entity classification hierarchy* represents the similarities among the entity classifications. For example, a general entity class "terrorist" may have subtypes of "planner", "assassin", or "liaison". The *relationship hierarchy* is intended to represent the similarities among the existing relationships (following the "is-a" semantics). For example, "*supports*" is a relationship linking people and terrorist organizations (in the context of terrorism). It is the parent of several other relationships, including "*funds*", "*trains*", "*shelters*", etc.

A semantic query language can be used to express various semantic queries outlined below (the first two represents existing technology, third represents emerging technology, and the remaining represent novel research):

1. *Keyword queries*, as offered by traditional, search engines today. The query is a Boolean combination of search keywords and the result is the set of documents satisfying the query.
2. *Entity queries.* The query is a Boolean combination of entity names and the result is the set of documents satisfying the query. Note, that a given entity may be identified by different names (or different forms of the same name), as for example "*Usama bin Laden,*" "*Osama bin Laden*," and "*bin Laden, Osama,*" all identify the same entity.
3. *Relationship queries*. This type of queries involves using a specific relationship (for example, *sponsoredBy*) from the KB to find related entity(ies). A secondary result may include a set of documents matching the identified entities, and if possible, supporting the used relationship, as stored in the KB.
4. *Path queries.* Queries of this type involve using a sequence (path) of specific relationships in order to find connected entities. In addition, in order to take into account the relationship hierarchy, a query involving the relationship *supports* (as one of the relationships in the path) will result in entities linked by this and any of the sub-relationships (such as "*funds*", "*trains*", "*shelters*", etc.). The secondary result

may include a set of documents matching the identified entities, and if possible, supporting the relationships used in the path and stored in the KB.

5. *Path discovery queries.* This is the most powerful and arguably the most interesting form of semantic queries. This type of query involves a number of entities (possibly just a pair of entities) and attempts to return a set of paths (including relationships and intermediate entities) that connect the entities in the query. Each computed path represents a semantic association of the named entities.

Semantic query processing involves the construction of a specialized Semantic Index (SI). We view the structure of the SI as a three-level index, involving the "traditional" keywords (at level 1), entities and/or concepts (at level 2), as well relationships (at level 3) existing among the entities. The SI is shown in the Figure 6.

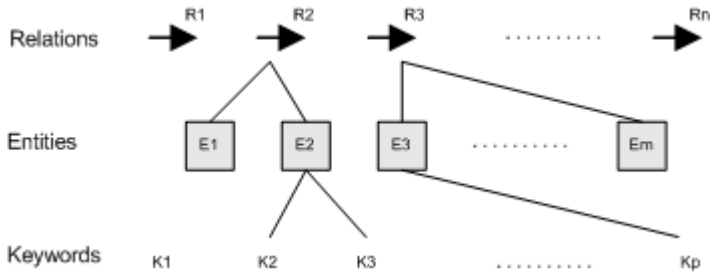


Figure 6: Semantic Index

The SI constitutes a foundation for the design of a suitable semantic query engine. We must note that the most general of the semantic queries (of type 6 above) in an unconstrained form may be computationally prohibitive. However, when the length of the path is limited to a relatively small fixed number, the computation of the result set is possible.

4.2. ρ Operator

In this section, we highlight an approach for computing complex semantic relations using an operator we call ρ (Rho) [Anyanwu02]. The ρ operator is intended to facilitate complex path navigation in KBs. It permits the navigation of metadata (e.g., resource descriptions in RDF) as well as schema/taxonomies (e.g., ontologies in RDFS, DAML+OIL, or OWL [Heflin02]).

More specifically, the operator ρ provides the mechanism for reasoning about semantic associations that exist in KBs. The binary form of this operator, $\rho_T(a, b)$ [C, K], will return a set of semantic relations between entities a and b. Since semantic relations include not just single relationships but also associations that are realized as a sequence of relationships in a KB or based on certain patterns in such sequences, a mechanism that attempts to find possible paths, and in some cases makes comparisons about similarity of paths/sub-graphs is need. Of course this may be computationally very expensive. The parameters C and K allow us to focus and speed up the computation. C is the context (e.g., a relevant ontology) given by the user, which helps to narrow the search for associations to a specific region in the KB. K is a set of constraints that includes user given restrictions, heuristics and some domain knowledge that is used to limit the search and prioritize the results.

$\rho_T(a , b)$ [C, K] represents the generic form of the ρ operator where the subscript T represents the type of the operator. The types are as follows:

$\rho_{PATH}(a, b)$ [C, K]	Given the entities *a,* and *b,* ρ_{PATH} looks for directed paths from *a* to *b* and returns a subset of possible paths.
$\rho_{INTERSECT}(a, b)$ [C, K]	Given entities *a,* and *b,* $\rho_{INTERSECT}$ looks to see if there are directed paths from *a* and *b* that intersect at some node, say *c*. In other words, it checks to see if there exists a node c such that: $\rho_{PATH}(a , c) \& \rho_{PATH}(b , c)$.Thus, this query returns a set of path pairs where the paths in each pair are intersecting paths.
$\rho_{CONNECT}(a, b)$ [C, K]	Given entities *a,* and *b,* $\rho_{CONNECT}$ treats the graph as an undirected graph and looks for a set of edges forming an undirected path between *a,* and *b*. This query returns a subset of possible paths.
$\rho_{ISO}(a, b)$ [C, K]	Given entities a, and b, ρ_{ISO} looks for a pair of directed sub-graphs rooted at *a,* and *b,* respectively, such that the 2 sub-graphs are $\rho_{ISOMORPHIC}$. $\rho_{ISOMORPHISM}$ represents the notion of semantic similarity between the 2 sub-graphs.

5. Human-Assisted Knowledge Discovery Involving Complex Relations

In this section, we discuss the concept of IScape in the InfoQuilt system which allows a hypothesis involving complex relationships and its validation over heterogeneous, distribution content.

A great deal of research into enabling technologies for the Semantic Web and semantic interoperability in information systems has focused on domain knowledge representation through the use of ontologies. Current state-of-the-art ontological representational schemes represent knowledge as a hierarchical taxonomy of concepts and relationships such as is-a/role-of, instance-of/member-of and part-of. Fulfilling information requests on systems based on such representation and associated "crisp logic" based reasoning or inference mechanisms [Dec] allow for supporting queries of limited complexity [DHM+01], and additional research in query languages and query processing is rapidly continuing. For example, SCORE allows combining querying of metadata and ontology. An alternative approach has been taken in the InfoQuilt system that supports human-assisted knowledge discovery [Sheth02b]. Here users are able to pose questions that involve exploring complex hypothetical relationships amongst concepts within and across domains, in order to gain a better understanding of their domains of study, and the interactions between them. Such relationships across domains, e.g., causal relationships, may not necessarily be hierarchical in nature and such questions may involve complex information requests involving user defined functions and fuzzy or approximate match of objects, therefore requiring richer environment in terms of expressiveness and computation. For example, a user may want to know "Does Nuclear Testing *cause* Earthquakes?" Answering such a question requires correlation of data from sources of the domain Natural-Disasters**.**Earthquake with data from sources of Nuclear-Weapons**.**Nuclear-Testing domain. Such a correlation is only possible if, among other things, the user's notion of "cause" is clearly understood and exploited. This involves the use of ontologies of the involved domains for shared understanding of the terms and their relationships. Furthermore, the user should be allowed to express their meaning (or definition) of the causal relationship. In this case it could be based on the proximity in

time and distance between the two events (i.e., nuclear tests and earthquakes), and this meaning should be exploited when correlating data from the different sources. Subsequent investigation of the relationship by refining and posing other questions based on the results presented, may lead the user to a better understanding of the nature of the interaction between the two events. This process is what we refer to as Human-Assisted KNowledge Discovery (HAND). Note that this approach is fundamentally different than the relationship types discussed earlier in the sense that a non-existent new relationship is named, and its precise semantic is defined through a computation. If that computation verifies the existence of this hypothetical relationship it can be placed permanently in an ontology.

InfoQuilt uses ontologies to model the domains of interest. Ontology captures useful semantics of the domain such as the terms and concepts of interest, their meanings, relationships between them and the characteristics of the domain. Ontology provides a structured, homogeneous view over all the available data sources. It is used to standardize the meaning, description and the representation of the attributes across the sources (we call it semantic normalization). All the resources are mapped to this integrated view and this helps to resolve the source differences and makes schema integration easier. An example of "disaster" ontology is shown in Figure 7.

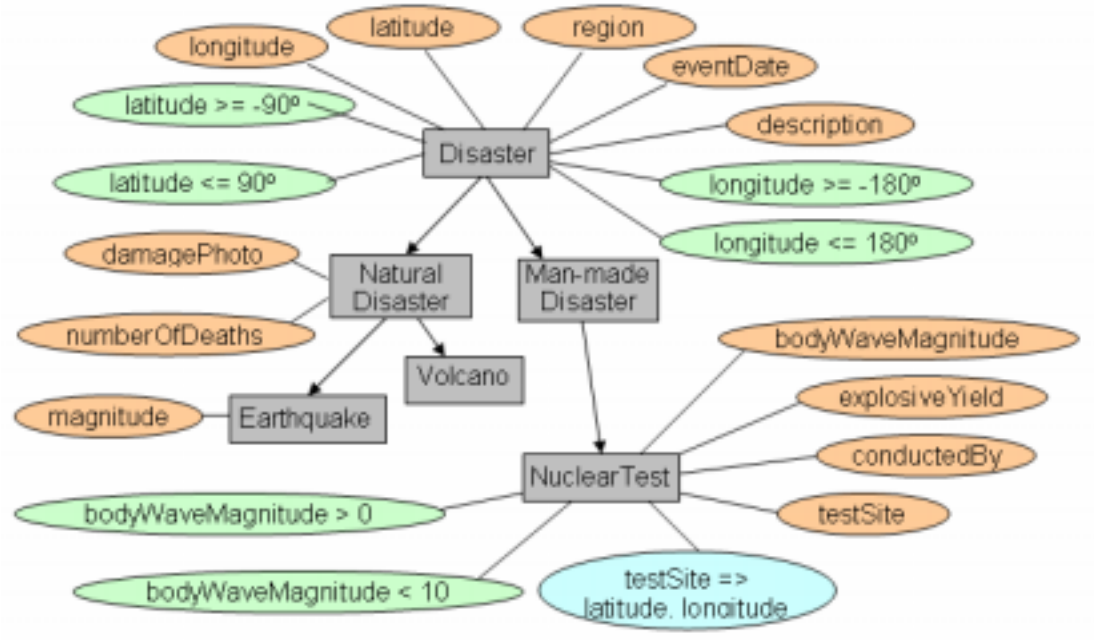


Figure 7: Disaster Ontology

5.1. User-Defined Functions

A distinguishing feature of InfoQuilt is its framework to support user-defined operations. The user can use them to specify additional constraints in their information requests. For example, consider the information request:

"*Find all earthquakes with epicenter in a 5000 mile radius area of the location at latitude 60.790 North and longitude 97.570 East*"

The system needs to know how it can calculate the distance between two points, given their latitudes and longitudes, in order to check which earthquakes' epicenters fall in the range specified. The function distance can again be used here.

These user-defined functions are also helpful for supporting a context-specific fuzzy matching of attribute values. For example, assume that we have two data sources for the domain of earthquakes. It is quite possible that two values of an attribute testSite retrieved from the two sources may be syntactically unequal but refer to the same location. For example, the value available from one source could be "Nevada Test Site, Nevada, USA" and that from another source could be "Nevada Site, NV, USA". The two are semantically equal but syntactically unequal [KS96]. Fuzzy matching functions can be useful in comparing the two values.

Another important advantage of using operations is that the system can support complex post-processing of data. An interesting form of post-processing is the use of simulation programs. For instance, researchers in the field of Geographic Information Systems (GIS) use simulation programs to forecast characteristics like urban growth in a region based on a model. InfoQuilt supports the use of such simulations like any other operation.

5.2. Information Scapes (IScapes)

InfoQuilt uses IScape, a paradigm for information request which is "*a computing paradigm that allows users to query and analyze the data available from a diverse autonomous sources, gain better understanding of the domains and their interactions as well as discover and study relationships*."

Consider the following information request.

"*Find all earthquakes with epicenter in a 5000 mile radius area of the location at latitude 60.790 North and longitude 97.570 East and find all tsunamis that they might have caused.*"

In addition to the obvious constraints, the system needs to understand what the user means by saying "*find all tsunamis that might have been caused due to the earthquakes*". The relationship that *an earthquake caused a tsunami* is a complex inter-ontological relationship.

Any system that needs to answer such information requests would need a comprehensive knowledge of the terms involved and how they are related. An IScape is specified in terms of relevant ontologies, inter-ontological relationships and operations. Additionally, this abstracts the user from having to know the actual sources that will be used by the system to answer it and how the data retrieved from these sources will be integrated, including how the results should be grouped, any aggregations that need to be computed, constraints that need to be applied to the grouped data, and the information that needs to be returned in the result to the user.

The ontologies in the IScape identify the domains that are involved in the IScape and the inter-ontological relationships specify the semantic interaction between the ontologies. The preset constraint and the runtime configurable constraint are filters used to describe the subset of data that the user is interested in, similar to the WHERE clause in an SQL query. For example, a user may be interested in earthquakes that occurred in only a certain region and had a magnitude greater than 5. The difference between the preset constraint and the runtime constraint is that the runtime constraint can be set at the time of executing the IScape. The results of the IScape can be grouped based on attributes and/or values computed by functions.

5.3. Human Assisted Knowledge Discovery (HAND) Techniques

InfoQuilt provides a framework that allows users to access data available from a multitude of diverse autonomous distributed resources and provide tools that help them to

analyze the data to gain a better understanding of the domains and the inter-domain relationships as well as help users to explore the possibilities of new relationships.

Existing relationships in the knowledgebase provide a scope for discovering new aspects of relationships through transitive learning. For example, consider the ontologies Earthquake, Tsunami and Environment. Assume that the relationships "Earthquake affects Environment", "Earthquake causes Tsunami" and "Tsunami affects Environment" are defined and known to the system. We can see that since Earthquake causes a Tsunami and Tsunami affects the environment, effectively this is another way in which an earthquake affects the environment (by causing a tsunami). If this aspect of the relationship between an earthquake and environment was not considered earlier, it can be studied further.

Another valuable source of knowledge discovery is studying existing IScapes that make use of the ontologies, their resources and relationships to retrieve information that is of interest to the users. The results obtained from IScapes can be analyzed further by post processing of the result data. For example, the Clarke UGM model forecasts the future patterns of urban growth using information about urban areas, roads, slopes, vegetation in those areas and information about areas where no urban growth can occur.

For the users that are well-versed with the domain, the InfoQuilt framework allows exploring new relationships. The data available from various sources can be queried by constructing IScapes and the results can be analyzed by using charts, statistical analysis techniques, etc. to study and explore trends or aspects of the domain. Such analysis can be used to validate any hypothetical relationships between domains and to see if the data validates or invalidates the hypothesis. For example, several researchers in the past have expressed their concern over nuclear tests as one of the causes of earthquakes and suggested that there could be a direct connection between the two. The underground nuclear tests cause shock waves, which travel as ripples along the crust of the earth and weaken it, thereby making it more susceptible to earthquakes. Although this issue has been addressed before, it still remains a hypothesis that is not conclusively and scientifically proven. Suppose we want to explore this hypothetical relationship.

Consider the NuclearTest and Earthquake ontologies again. We assume that the system has access to sufficient resources for both the ontologies such that they together provide

sufficient information for the analysis. However, note that the user is not aware of these data sources since the system abstracts him from them. To construct IScapes, the user works only with the components in the knowledgebase. If the hypothesis is true, then we should be able to see an increase in the number of earthquakes that have occurred after the nuclear testing started.

An example IScape for testing this hypothesis is given below:

"*Find nuclear tests conducted after January 1, 1950 and find any earthquakes that occurred not later than a certain number of days after the test and such that its epicenter was located no farther than a certain distance from the test site.*"

Note the use of "*not later than a certain number of days*" and "*no farther than a certain distance*". The IScape does not specify the value for the time period and the distance. These are defined as runtime configurable parameters, which the user can use to form a constraint while executing the IScape. The user can hence supply different values for them and execute the IScape repeatedly to analyze the data for different values without constructing it repeatedly from scratch. Some of the interesting results that can be found by exploring earthquakes occurring that occurred no later than 30 days after the test and with their epicenter no farther than 5000 miles from the test site are listed below.

- China conducted a nuclear test on October 6, 1983 at Lop Nor test site. USSR conducted two tests, one on the same day and another on October 26, 1983, both at Easter Kazakh or Semipalitinsk test site. There was an earthquake of magnitude 6 on the Richter scale in Erzurum, Turkey on October 30, 1983, which killed about 1300 people. The epicenter of the earthquake was about 2000 miles away from the test site in China and about 3500 miles away from the test site in USSR. The second USSR test was just 4 days before the quake.
- USSR conducted a test on September 15, 1978 at Easter Kazakh or Semipalitinsk test site. There was an earthquake in Tabas, Iran on September 16, 1978. The epicenter was about 2300 miles away from the test site.

More recently, India conducted a nuclear test at its Pokaran test site in Rajasthan on May 11, 1998. Pakistan conducted two nuclear tests, one on May 28, 1998 at Chagai test site and another on May 30, 1998. There were two earthquakes that occurred soon after these

tests. One was in Egypt and Israel on May 28, 1998 with its epicenter about 4500 miles away from both test sites and another in Afghanistan, Tajikistan region on May 30, 1998, with a magnitude of 6.9 and its epicenter about 750 miles away from the Pokaran test site and 710 miles from Chagai test site.

6. Evaluations involving Semantic Relationships: Example of Multi-ontology Query Processing

Our last section deals with some issues in evaluating complex relationships across information domains, potentially spanning multiple ontologies. Most practical situations in the Semantic Web will involve multiple overlapping or disjoint but related ontologies. For example, an information request might be formulated using terms in one ontology but the relevant resources may be annotated using terms in other ontologies. Computations such as query processing in such cases will involve complex relationships spanning multiple ontologies. This raises several difficult problems, but perhaps the key problem is that of impact on quality of results or the change in query semantics when the relationships involves are not synonyms. In this chapter, we present the case study of multi-ontology query processing in the OBSERVER project..

A user query formulated using terms in domain ontology is translated by using terms of other (target) domain ontologies. Mechanisms dealing with incremental enrichment of the answers are used. The substitution of a term by traversing inter-ontological relationships like synonyms (or combinations of them [Mena96]) and combinations of hyponyms (specializations) and hypernyms (generalizations) provide answers not available otherwise by using only a single ontology. This, however, changes the semantics of the query. We discuss with the help of examples, mechanisms to estimate loss of information (based on intensional and extensional properties) in the face of possible semantic changes when translating a query across different ontologies. This measure of the information loss (whose upper limit is defined by the user) guides the system in navigating those ontologies that have more relevant information; it also provides the user with a level of confidence in the answer that may be retrieved. Well-established metrics like precision and recall are used and adapted to our context in order

to measure the change in semantics instead of the change in the extension, unlike techniques adopted by classical Information Retrieval methods.

6.1. Query Processing in OBSERVER

The idea underlying our query processing algorithm is the following: give the first possible answer and then enrich it in successive iterations until the user is satisfied. Moreover, certain degree of imprecision (defined by each user) in the answer could be allowed if it helps to speed up the search of the wanted information. We use ontologies, titled **WN** and **Stanford-I** (see [Mena00]) and the following example query to illustrate the main steps of our query expansion approach.

User Query: *`Get title and number of pages of books written by Carl Sagan'*

The user browses the available ontologies (ordered by knowledge areas) and chooses a user ontology that includes the terms needed to express the semantics of her/his information needs. Terms from the user ontology are chosen, to express the constraints and relationships that comprise the query. In the example, the WN ontology is selected since it contains all the terms needed to express the semantics of the query, i.e., terms that store information about titles (`NAME'), number of pages (`PAGES'), books (`BOOK') and authors (`CREATOR').

Q = [NAME PAGES] for (**AND** BOOK (**FILLS** CREATOR “Carl Sagan”))
Syntax of the expressions is taken from CLASSIC [BBMR89], the system based on Description Logics (DL) that we use to describe ontologies.

Controlled and Incremental Query Expansion to Multiple Ontologies

If the user is not satisfied with the answer, the system retrieves more data from other ontologies in the Information System to “enrich” the answer in an incremental manner. In doing so, a new component ontology, the target ontology, whose concepts participate in inter-ontological relationships with the user ontology is selected. The user query is then expressed/translated into terms of that target ontology. The user and target ontologies are integrated by using the inter-ontology relationships defined between them.

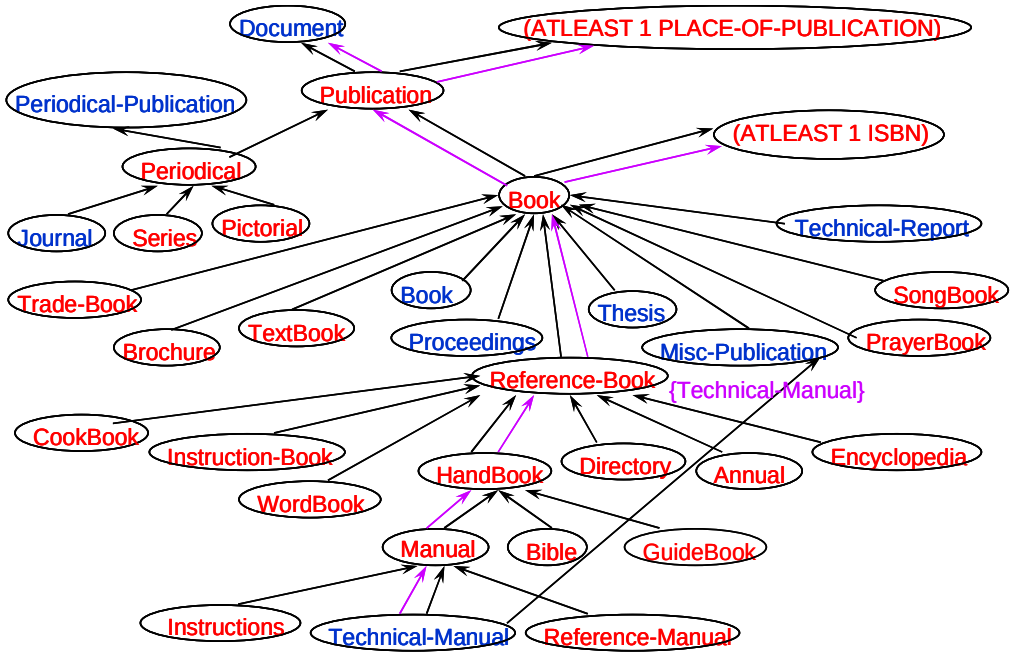


Figure 8: Use of inter-ontological relationships to integrate multiple ontologies

- All the terms in the user query may have been rewritten by their corresponding synonyms in the target ontology. Thus the system obtains a semantically equivalent query (**full translation**) and no loss of information is incurred.
- There exist terms in the user query that can not be translated into the target ontology - they do not have synonyms in the target ontology (we called them conflicting terms). This is called a **partial translation**.

Each conflicting term in the user query is replaced by the intersection of its immediate parents (hypernyms) or by the union of its immediate children (hyponyms), recursively, until a translation of the conflicting term is obtained using only the terms of the target ontology. This could lead to several candidate translations, leading to change in semantics and loss of information. The query Q discussed above has to be translated into terms of the Stanford-I ontology [Mena00]. After the process of integrating the WN and Stanford-I ontologies (Figure 8), Q is redefined as follows:

Q = [title number-of-pages] for (**AND** BOOK (**FILLS** doc-author-name “Carl Sagan”))

The only conflicting term in the query is `BOOK' (it has no translation into terms of Stanford-I). The process of computing the various plans for the term `BOOK' results in

four possible translations: `document', `periodical-publication', `journal' or `UNION(book, proceedings, thesis, misc-publication, technical-report)'. Details of this translation process can be found in [MKIS98]. This leads to 4 possible translations of the query:

Plan 1: (**AND** document (**FILLS** doc-author-name "Carl Sagan"))

Plan 2: (**AND** periodical-publication (**FILLS** doc-author-name "Carl Sagan"))

Plan 3: (**AND** journal (**FILLS** doc-author-name "Carl Sagan"))

Plan 4: (**AND** UNION(book, proceedings, thesis, misc-publication, technical-report)
(**FILLS** doc-author-name "Carl Sagan"))

6.2. Estimating the Loss of Information

We use the Information Retrieval analogs of soundness (precision) and completeness (recall), which are estimated based on the sizes of the extensions of the terms. We combine these two measures to compute a composite measure in terms of a numerical value. This can then be used to choose the answers with the least loss of information.

Loss of information based on intensional information

The loss of information can be expressed like the terminological difference between two expressions, the user query and its translation. The terminological difference between two expressions consists of those constraints of the first expression that are not subsumed by the second expression. The loss of information for Plan 1 is as follows:

Plan 1: (**AND** document (**FILLS** doc-author-name "Carl Sagan"))

Taking into account the following term definitions[1]:

BOOK = (**AND** PUBLICATION (**ATLEAST** 1 ISBN)),

PUBLICATION = (**AND** document (**ATLEAST** 1 PLACE-OF-PUBLICATION))

The terminological difference is, in this case, the constraints not considered in the plan:

(**AND** (**ATLEAST** 1 ISBN) (**ATLEAST** 1 PLACE-OF-PUBLICATION))

The intensional loss of information of the 4 plans can thus be enumerated as:

[1] The terminological difference is computed across extended definitions

- Plan = (**AND** document (**FILLS** doc-author-name "Carl Sagan"))
  Loss = "Instead of books written by Carl Sagan, all the documents written by Carl Sagan are retrieved, even if they do not have an ISBN and place of publication".
- Plan = (**AND** periodical-publication (**FILLS** doc-author-name "Carl Sagan"))
  Loss = "Instead of books written by Carl Sagan, all periodical publications written by Carl Sagan are retrieved, even if they do not have an ISBN and place of publication".
- Plan = (**AND** journal (**FILLS** doc-author-name "Carl Sagan"))
  Loss = "Instead of books written by Carl Sagan, all journals written by Carl Sagan are retrieved, even if they do not have an ISBN and place of publication".
- Plan = (**AND** UNION(book, proceedings, thesis, misc-publication, technical-report)
  (**FILLS** doc-author-name "Carl Sagan"))
  Loss = "Instead of books written by Carl Sagan, book , proceedings, theses, misc-publication and technical manuals written by Carl Sagan are retrieved".

An intensional measure of the loss of information can make it hard for the system to decide between two alternatives, in order to execute first plan with less loss. Thus, some numeric way of measuring the loss should be explored.

Loss of information based on extensional information

The loss of information is based on the number of instances of terms involved in the substitutions performed on the query and depends on the sizes of the term extensions. A composite measure combining measures like *precision* and *recall* [Sal89] used to estimate the information loss is described, which takes into account the bias of the user ("is precision more important or recall ?").

The extension of a query expression is a combination of unions and intersections of concepts in the target ontology since and is estimated with an upper (|Ext(Expr)|.high) and lower (|Ext(Expr)|.low) bound. It is computed as follows:

$|Ext(Subexpr_1) \cap Ext(Subexpr_2)|.low = 0$

$|Ext(Subexpr_1) \cap Ext(Subexpr_2)|.high = min\ [|Ext(Subexpr_1)|.high, |Ext(Subexpr_2)|.high\ ]$

$|Ext(Subexpr_1) \cup Ext(Subexpr_2)|.low = max\ [|Ext(Subexpr_1)|.high, |Ext(Subexpr_2)|.high\ ]$

$|Ext(Subexpr_1) \cup Ext(Subexpr_2)|.high = |Ext(Subexpr_1)|.high + |Ext(Subexpr_2)|.high$

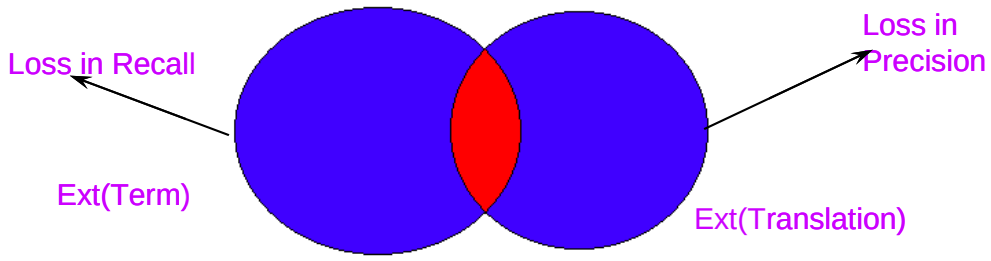
A composite measure combining precision and recall

Precision and Recall have been very widely used in Information Retrieval literature to measure loss of information incurred when the answer to a query issued to the information retrieval system contains some proportion of irrelevant data [Sal89]. The measures are adapted to our context, as follows:

$$\Pr ecision = \frac{|\, Ext(Term) \cap Ext(Translation) \,|}{|\, Ext(Translation) \,|}, \Re call = \frac{|\, Ext(Term) \cap Ext(Translation) \,|}{Ext(Term)}$$

We use a composite measure [vR] which combines the precision and recall to estimate the loss of information. We seek to measure the extent to which the two sets do not match. This is denoted by the shaded area in Figure 1. The area is, in fact, the symmetric difference:

RelevantSet Δ RetrievedSet = RelevantSet $\cup$ RetrievedSet - RelevantSet $\cap$ RetrievedSet



The loss of information may be given as:

$$Loss = \frac{|\, \Re elevantSet \,\Delta\, \Re trievedSet \,|}{|\, \Re elevantSet \,| + |\, \Re trievedSet \,|} \quad Loss = 1 - \frac{1}{\frac{1}{2}\left(\frac{1}{\Pr ecision}\right) + \frac{1}{2}\left(\frac{1}{\Re call}\right)}$$

Semantic adaptation of precision and recall

Higher priority needs to be given to semantic relationships than those suggested by the underlying extensions. The critical step is to estimate the extension of Translation based on the extensions of terms in the target ontology. Precision and recall are adapted as follows:

- **Precision and recall measures for the case where a term subsumes its translation**. Semantically, we do not provide an answer irrelevant to the term,

as Ext(Translation) ⊆ Ext(Term) (by definition of subsumption).

Thus, Ext(Term) ∩ Ext(Translation) = Ext(Translation).

Therefore:

$$\Pr ecision = 1, \quad \text{Re}call = \frac{|\ Ext(Term) \cap Ext(Translation)\ |}{|\ Ext(Term)\ |} = \frac{|\ Ext(Translation)\ |}{|\ Ext(Term)\ |}$$

$$\text{Re}call.low = \frac{|\ Ext(Translation)\ |.low}{|\ Ext(Term)\ |}, \quad \text{Re}call.high = \frac{|\ Ext(Translation)\ |.high}{|\ Ext(Term)\ |}$$

- **Precision and recall measures for the case where a term is subsumed by its translation**

  Semantically, all elements of the term extension are returned,

  as Ext(Term) ⊆ Ext(Translation) (by definition of subsumption).

  Thus, Ext(Term) ∩ Ext(Translation) = Ext(Term).

  Therefore:

$$\text{Re}call = 1, \quad \Pr ecision = \frac{|\ Ext(Term) \cap Ext(Translation)\ |}{|\ Ext(Translation)\ |} = \frac{|\ Ext(Term)\ |}{|\ Ext(Translation)\ |}$$

$$\Pr ecision.low = \frac{|\ Ext(Term)\ |}{|\ Ext(Translation)\ |.high}, \quad \Pr ecision.high = \frac{|\ Ext(Term)\ |}{|\ Ext(Translation)\ |.low}$$

- **Term and Expression are not related by any subsumption relationship**.

  The general case is applied directly since intersection cannot be simplified. In this case the interval describing the possible loss will be wider as Term and Translation are not related semantically[2].

$$\Pr ecision.low = 0, \ \Pr ecision.high = \max\left[\frac{\min\left[|\ Ext(Term)\ |, |\ Ext(Translation)\ |.high\right]}{|\ Ext(Translation)\ |.high}, \frac{\min\left[|\ Ext(Term)\ |, |\ Ext(Translation)\ |.low\right]}{|\ Ext(Translation).low\ |}\right]$$

$$\text{Re}call.low = 0, \ \text{Re}call.high = \frac{\min\left[|\ Ext(Term)\ |, |\ Ext(Translation)\ |.low\right]}{|\ Ext(Term)\ |}$$

The various measures defined above are applied to the 4 translations and the loss of information intervals are computed. The values are illustrated in Table 1. For a detailed account of the compuations involved, the reader may look at [Mena00].

[2] As we change the numerator and the denominator, we do not know which one is greater.

TRANSLATION	**LOSS OF INFORMATION**
(**AND** document (**FILLS** doc-author-name "Carl Sagan"))	91.571% ≤ Loss ≤ 91.755%
(**AND** periodical-publication (**FILLS** doc-author-name "Carl Sagan"))	94.03% ≤ Loss ≤ 100%
(**AND** journal (**FILLS** doc-author-name "Carl Sagan"))	98.56% ≤ Loss ≤ 100%
(**AND** (**FILLS** doc-author-name "Carl Sagan") UNION(book, proceedings, thesis, misc-publication, technical report))	0 ≤ Loss ≤ 7.22%

Table 1: Various Translations and the respective loss of Information

7. Conclusions

Ontologies provide the semantic underpinning, while relationships are the backbone for semantics in the Semantic Web or any approach to achieving semantic interoperability. For more semantic solutions, attention needs to shift from documents (e.g., searching for relevant documents) to integrated approach of exploiting data (content, documents) with knowledge (including domain ontologies). Relationships, their modeling, specification or representation, identification, validation or their use in query or information request evaluation are then the fundamental aspects of study. In this chapter, we have provided an initial taxonomy for studying various aspects of semantic relationships. To exemplify the some points in the broad scope of studying semantic relationships, we discussed four examples of our own research efforts during the past decade. Neither the taxonomy nor our empirical exemplification through four examples is a complete study. We hope it would be extended with study of extensive research reported in the literature by other researchers.

Acknowledgements

Ideas presented in this chapter have benefited from team members at the LSDIS Lab (projects: InfoHarmess, VisualHarness, VideoAnywhere, InfoQuilt, and Semantic Association Identification), and Semagix. Special acknowledgements to Eduardo Mena (for his contributions to the OBSERVER project), Kemafor Anyanwu (for her work on Semantic Associations), Brian Hammond, Clemens Bertram and David Avant (for their work on relevant parts of SCORE discussed here), and Krys Kochut (for discussions on semantic index and his work on SCORE).

References

[Anyanwu02] K. Anyanwu and A. Sheth, "The ρ Operator: Computing and Ranking Semantic Associations in the Semantic Web", SIGMOD Record, December 2002.

[Arumugam02] M. Arumugam, A. Sheth, and I. B. Arpinar, "Towards Peer-to-Peer Semantic Web: A Distributed Environment for Sharing Semantic Knowledge on the Web", Intl. Workshop on Real World RDF and Semantic Web Applications 2002, Hawaii, May 2002.

[Bailin01] S. C. Bailin, and W. Truszkowski, "Ontology Negotiation Between Agents Supporting Intelligent Information Management", Workshop On Ontologies In Agent Systems, 2001.

[Berners-Lee01] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web, A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities", Scientific American, May 2001.

[Boll98] S. Boll, W. Klas and A. Sheth, "Overview on Using Metadata to Manage Multimedia Data", in Multimedia Data Management: Using Metadata to Integrate and Apply Digital Media, A. Sheth and W. Klas, Eds., McGraw-Hill Publishers, March 1998.

[Brezillon02] P. Brezillon, and J.-C. Pomerol, "Reasoning with Contextual Graphs", European Journal of Operational Research, 136(2): 290-298, 2002.

[Brezillon01] P. Brezillon, and J.-C. Pomerol, "Is Context a Kind of Collective Tacit Knowledge?", European CSCW 2001 Workshop on Managing Tacit Knowledge. Bonn, Germany. M. Jacovi and A. Ribak (Eds.), pp. 23-29, 2001.

[Brezillon99a] P. Brezillon, "Context in Problem Solving: A Survey", The Knowledge Engineering Review, 14(1): 1-34, 1999.

[Brezillon99b] P. Brezillon, "Context in Artificial Intelligence: I. A Survey of the Literature", Computer & Artificial Intelligence, 18(4): 321-340, 1999.

[Brezillon99c] P. Brezillon, "Context in Artificial Intelligence: II. Key Elements of Contexts", Computer & Artificial Intelligence,18(5): 425-446, 1999.

[Buneman00] P. Buneman, S. Khanna, and W.-C. Tan, "Data Provenance: Some Basic Issues", Foundations of Software Technology and Theoretical Computer Science (2000).

[Buneman02a] P. Buneman, S. Khanna, K. Tajima, and W.-C. Tan, "Archiving Scientific Data", Proceedings of ACM SIGMOD International Conference on Management of Data (2002).

[Chen99] Y. Chen, Y. Peng, T. Finin, Y. Labrou, and S. Cost, "Negotiating Agents for Supply Chain Management", AAAI Workshop on Artificial Intelligence for Electronic Commerce, AAAI, Orlando, June 1999.

[Constantopoulos93] P. Constantopoulos, and M. Doerr, "The Semantic Index System - A brief presentation", Institute of Computer Science Technical Report. FORTH-Hellas, GR71110 Heraklion, Crete, 1993.

[Cost02] R. S. Cost, T. Finin, A. Joshi, Y. Peng, et. Al., "ITTALKS: A Case Study in DAML and the Semantic Web", IEEE Intelligent Systems Special Issue, 2002.

[Finnin88a] T. Finin, "Default Reasoning and Stereotypes in User Modeling", International Journal of Expert Systems, Volume 1, Number 2, Pp. 131-158, 1988.

[Finin92] T. Finin, R. Fritzson, and D. McKay, "A Knowledge Query and Manipulation Language for Intelligent Agent Interoperability", Fourth National Symposium on Concurrent Engineering, CE & CALS Conference, Washington, DC June 1-4, 1992.

[Heflin02] J. Heflin, R. Volz. J. Dale, Eds., Requirements for a Web Ontology Language, March 07, 2002. http://www.w3.org/TR/webont-req/

[Hendler01] J. Hendler, "Agents and the Semantic Web", IEEE Intelligent Systems, 16(2), March/April, 2001.

[Heuer99] R. J. Heuer, Jr., "Psychology of Intelligence Analysis", Center for the Study of Intelligence, Central Intelligence Agency, 1999.

[Joshi00] A. Joshi, and R. Krishnapuram, "On Mining Web Acceess Logs", Proc. SIGMOD 2000 Workshop on Research Issues in Data Mining and Knowledge Discovery, pp 63-69, Dallas, 2000.

[Joshi02] K. Joshi, A. Joshi, Y. Yesha, "On Using a Warehouse to Analyze Web Logs", accepted for publication in Distributed and Parallel Databases, 2002.

[Kagal01a] L. Kagal, T. Finin, and A. Joshi, "Trust-Based Security For Pervasive Computing Environments", IEEE Communications, December 2001.

[Kagal01b] L. Kagal, T. Finin, and Y. Peng, "A Delegation Based Model for Distributed Trust Management", In Proceedings of IJCAI-01 Workshop on Autonomy, Delegation, and Control, August 2001.

[Kagal01c] L. Kagal, S. Cost, T. Finin, and Y. Peng, "A Framework for Distributed Trust Management", In Proceedings of Second Workshop on Norms and Institutions in MAS, Autonomous Agents, May 2001.

[Krishnapuram R. Krishnapuram, A. Joshi, O. Nasraoui, and L. Yi, "Low Complexity Fuzzy

01] Relational Clustering Algorithms for Web Mining", IEEE Trans. Fuzzy Systems, 9:4, pp 595-607, 2001.

[Kashyap96] V. Kashyap, and A. Sheth, "Semantic Heterogeneity in Global Information Systems: The Role of Metadata, Context, and Ontologies, in Cooperative Information Systems: Current Trends and Directions", M Papazoglou and G. Sclageter (eds), 1996.

[Kashyap95] V. Kashyap, and A. Sheth, "Metadata for building the Multimedia Patch Quilt," "Multimedia Database Systems: Issues and Research Directions, S. Jajodia and V. S. Subrahmanium, Eds., Springer-Verlag, p. 297-323, 1995.

[Kashyap00] V. Kashyap and A. Sheth, "Information Brokering Across Heterogeneous Digital Data", Kluwer Academic Publishers, August 2000, 248 pages.

[Kass90] R. Kass, and T. Finin, "General User Modeling: A Facility to Support Intelligent Interaction", in J. Sullivan and S. Tyler (eds.) Architectures for Intelligent Interfaces: Elements and Prototypes, ACM Frontier Series, Addison-Wesley, 1990.

[Kirzen99] L. Kirzen, "Intelligence Essentials for Everyone, Occasional Paper Number Six", Joint Military Intelligence College, Washington, D.C., June 1999.

[Liere97] R. Liere, and P. Tadepelli, "Active Learning with Committees for Text Categorization", Proc. 14th Conf. Am. Assoc. Artificial Intelligence, AAAI Press, Menlo Park, Calif., 1997, pp. 591–596.

[Mena00] E. Mena, A. Illarramendi, V. Kashyap and A. Sheth, "OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies", Distributed and Parallel Databases (DAPD), Vol. 8, No. 2, April 2000, pp. 223-271.

[Nonaka95] I. Nonaka, and H. Takeuchi, "The Knowledge-Creating Company", Oxford University Press, New York, NY, 1995.

[Sebastiani02] F. Sebastiani, "Machine Learning in Automated Text Categorization," *ACM Computing Surveys*, vol. 34, no. 1, 2002, pp. 1–47.

[Shah97] K. Shah, A. Sheth, and S. Mudumbai, "Black Box approach to Visual Image Manipulation used by Visual Information Retrieval Engines", Proceedings of 2nd IEEE Metadata Conference, September 1997.

[Shah98] K. Shah and A. Sheth, Logical Information Modeling of Web-accessible Heterogeneous Digital Assets**,** Proc. of the Forum on Research and Technology Advances in Digital Libraries," (ADL'98), Santa Barbara, CA. April 1998, pp.

266-275.

[Shah99] K. Shah and A. Sheth, "InfoHarness: An Information Integration Platform for Managing Distributed, Heterogeneous Information," IEEE Internet Computing, November-December 1999, p. 18-28.

[Shah02] U. Shah, T. Finin, A. Joshi, R. S. Cost, and J. Mayfield, “Information Retrieval on the Semantic Web”, submitted to the 10th International Conference on Information and Knowledge Management, November 2002.

[Sheth96] A. Sheth and V. Kashyap, „Media-independent correlation of Information: What? How?“ Proceedings of the First IEEE Metadata Conference, April 1996. http://www.computer.org/conferences/meta96/sheth/

[Sheth90] A. Sheth and J. Larson, “Federated Databases: Architectures and Issues," ACM Computing Surveys, 22 (3), September 1990, pp. 183-236.

[Sheth98] A. Sheth, “Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics in Interoperating Geographic Information Systems”, M. F. Goodchild, M. J. Egenhofer, R. Fegeas, and C. A. Kottman (eds.), Kluwer, 1998.

[Sheth02b] A. Sheth, S. Thacker and S. Patel, “Complex Relationship and Knowledge Discovery Support in the InfoQuilt System”, VLDB Journal, 2002 .

[Sheth02a] A. Sheth, C. Bertram, D. Avant, B. Hammond, K. Kochut, and Y. Warke, “Semantic Content Management for Enterprises and the Web”, IEEE Internet Computing, July/August 2002.

[Srinivasan02] N. Srinivasan, and T. Finin, “Enabling Peer to Peer SDP in Agents”, Proceedings of the 1st International Workshop on "Challenges in Open Agent Systems, July 2002, University of Bologna, held in conjunction with the 2002 Conference on Autonomous Agents and Multiagent Systems.

[Tolia02a] S. Tolia, D. Khushraj, and T. Finin, “ITTalks: Event Notification Service: An illustrative case for services in the Agent cities Network”, Proceedings of the 1st International Workshop on Challenges in Open Agent Systems, July 2002.

[Wiederhold92] G. Wiederhold, “Mediators in the Architecture of Future Information Systems”, IEEE Computer 25(3): 38-49, 1992.

[Zadeh65] L.A. Zadeh. *Fuzzy sets*. In Information and Control, pages 338-353, 1965.