

Semantics for the Semantic Web: The Implicit, the Formal and the Powerful

Amit Sheth, Cartic Ramakrishnan and Christopher Thomas
Large Scale Distributed Information Systems lab
University of Georgia, Athens, GA USA

Abstract:

Enabling applications that exploit heterogeneous data in the Semantic Web will require us to harness a broad variety of semantics. Considering the role of semantics in a number of research areas in computer science, we organize semantics in three forms: implicit, formal and powerful, and explore their roles in enabling some of the key capabilities related to the Semantic Web. The central message of this paper is that building the Semantic Web purely on description logics will artificially limit its potential, and that we will need to both exploit well known techniques that support implicit semantics, and develop more powerful semantic techniques.

Keywords: Semantic Web, Semantic Technology, Formal Semantics, Informal Semantics, Implicit Semantics, Analytical Processing, Document Management and Retrieval, Knowledge Discovery, Soft Computing, Metadata, Semantic Search, Relationship Discovery, Semantic Analytics, Semantic Matching, Semantic Integration

1. Introduction

Semantics has been a part of several scientific disciplines, both in the realm of Computer Science and outside of it. Research areas such as Information retrieval (IR), Information Extraction (IE), Computational Linguistics (CL), Knowledge Representation (KR) Artificial Intelligence (AI) and Data(base) Management (DB) have all addressed issues pertaining to semantics in their own ways. Most of these areas have very different views of what “meaning” is, and these views are all built on some meta-theoretical and epistemological assumptions. These different views imply very different views of cognition, of concepts and of meaning [Hjorland 1998]. In this paper, we organize these different views to three forms of semantics: Implicit, Formal and Powerful (aka Soft). We use these forms to explore the role of semantics that go beyond the narrower interpretation of the Semantic Web (that involve adherence to contemporary SW standards) and encompass those required for a broad variety of semantic applications. We advocate that for the Semantic Web to be realized, we must harness the power of a broad variety of semantics encompassing all three forms:

IR, IE and CL techniques primarily draw upon analysis of unstructured texts in addition to document repositories that have a loosely defined and less formal structure. In these sorts of data sources the semantics are *Implicit*.

In the fields of KR, AI and DB, however, the data representation takes a more formal and/or rigid form. Well defined syntactic structures are used to represent information or knowledge where these structures have definite semantic interpretations associated with them. There are also definite rules of syntax that govern the ways in which syntactic

structures can be combined to represent the meaning of complex syntactic structures. In other words, techniques used in these fields rely on *Formal Semantics*.

Usually, efforts related to Formal Semantics have involved limiting expressiveness to allow for acceptable computational characteristics. Since most KR mechanisms and the Relational Data Model are based on set theory, the ability to represent and utilize knowledge that is imprecise, uncertain, partially true and approximate is lacking, at least in the base/standard models. However, there have been several efforts to extend the base models (e.g.,[Barbara et al 1992]). Representing and utilizing these types of more powerful knowledge is, in our opinion, critical to the success of the Semantic Web. Soft computing has explored these types of powerful semantics. We deem these *Powerful (soft) semantics* as distinguished, albeit not distinct from or orthogonal to *Formal* and *Implicit semantics*.

More recently, semantics has been driving the next generation of the Web as the Semantic Web where the focus is on the role of semantics for automated approaches to exploiting Web resources. This involves two well recognized critical enabling capabilities- ontology generation [Maedche and Staab 2001][Omelayenko 2001] and automated resource annotation [Hammond et al 2002][Dill et al 2003][Handschuh et al 2002][Patil et al 2004], which should be complemented by appropriate computational approach such as reasoning or query processing. We use a couple of such enabling capabilities to explore the role and importance of all three forms of semantics.

A majority of the attention in the Semantic Web has been centered on a logic based approach, more specifically, that of description logic. However, looking at past applications of semantics, it is very likely that more will be expected from the Semantic Web than what the careful compromise of expressiveness and computability represented by description logic and the W3C adopted ontology representation language OWL (even its three flavors) can support. Supporting expressiveness that meet requirements of practical applications and the techniques that support their development is crucial. It is not desirable to limit the Semantic Web to one type of representation where expressiveness has been compromised at the expense of computational property such as decidability.

This paper is not the first to make this above observation. We specifically identify a few. Michael Uschold [Uschold 2003] has discussed a semantic continuum involving informal to formal and implicit to explicit, and Tom Gruber has talked about informal, semi-formal and formal ontologies [Gruber 2003]. The way we use the term implicit semantics, however, is different compared to [Uschold 2003] insofar as we see implicit semantics in all kinds of data sets, not only in language. We assume that machines can analyze implicit semantics with several, mostly statistical, techniques. William Woods has extensively written regarding the limitations of the FOL (and hence DLs) in the context of natural language understanding, although limitations emanating from rigidity and limitation of expressive power as well as limited value reasoning supported in DLs can also be identified: "Over time, many people have responded to the need for increased rigor in knowledge representation by turning to first-order logic as a semantic criterion. This is distressing, since it is already clear that first-order logic is insufficient to deal with many semantic problems inherent in understanding natural language as well as the semantic requirements of a reasoning system for an intelligent agent using knowledge to interact with the world." [Woods 2004]. We also recall Lotfi Zadeh's long standing

work, such as [Zadeh 2002], in which he extensively discussed the need for what constitutes key part of the “powerful semantics” here. In essence, we hope to provide an integrated and complementary view on the range of options. One may ask what the uses of each of these types of semantics are in the context of the Semantic Web. Here is a quick take.

- Implicit Semantics is either largely present in most resources on the web or can easily (quickly) be extracted. Hence mining and learning algorithms applied to these resources can be utilized to extract structured knowledge or enrich existing structured formal representations. Since formal semantics intrinsically does not exist, implicit semantics is useful in processing data sets or corpus to obtain or bootstrap semantics that can be then represented in formal languages, potentially with human involvement.
- Formal Semantics in the form of Ontologies is relatively scarce but representation mechanisms with such semantics have definite semantic interpretations that make them more machine-processable. Representation mechanisms with formal semantics therefore afford applications the luxury of automated reasoning making the applications more intelligent.
- Powerful (soft) Semantics in the form of fuzzy or probabilistic KR mechanisms attempt to overcome the shortcomings of the rigid set-based interpretations associated with currently prevalent representation mechanisms by allowing for representation of degree of membership and degree of certainty. Some of the domain knowledge which human experts possess is intrinsically complex, and may require these more expressive representations and associated computational techniques.

These uses are further exemplified in sections 3 and 4 using Semantic Web applications as driving examples. In section 2 we define and describe *Implicit*, *Formal* and *Powerful (soft) semantics*.

2. Types of Semantics

In this section we give an overview of the three types of semantics mentioned. It is rather informal in nature as we only give a broad overview without getting in depth about the various formalisms or methods used. We assume that the reader is somewhat familiar with statistical methods on one hand and Description Logics/OWL on the other. We present a view of these methods in order to lead towards the necessity of powerful viz. soft semantics.

2.1 Implicit Semantics

This type of Semantics refers to the kind that is implicit from the patterns in data and that is not represented explicitly in any strict machine processable syntax. Examples of this sort of Semantics are the kind implied in the following scenarios:

- Co-occurrence of documents or terms in the same cluster after a clustering process based on some similarity measure is completed.
- A document linked to another document via a hyperlink, potentially associating semantic metadata describing the concepts that relate the two documents.

- The sort of Semantics implied by two documents belonging to categories that are siblings of each other in a concept hierarchy.
- Automatic classification of a document to broadly indicate what a document is about with respect to a chosen taxonomy. Further use the implied semantics to disambiguate (does the word “palm” in a document refer to a palm tree, the palm of your hand or a palm top computer?)
- Bioinformatics applications that exploit patterns like sequence alignment, secondary and tertiary protein structure analysis, etc.

One may argue that although there is no strict syntactic and explicit representation, the knowledge about patterns in data may yet be machine processable. For instance, it is possible to get a numeric similarity judgment between documents in a corpus. Although this is possible, this is the only sort of processing possible. It is not possible to look at documents and automatically infer the presence of a named relationship between concepts in the documents.

Even though the exploitation of implicit semantics draws upon well known statistical techniques, the wording is not a mere euphemism, but meant to give a different perception of the problem.

Many tools and applications for implicit semantics have been developed for decades and are readily available. Basically all machine learning exploits implicit semantics, namely clustering, concept- and rule learning, Hidden Markov Models, Artificial Neural Networks and others. These techniques supporting implicit semantics are found in early steps towards the Semantic Web, such as clustering in the Vivisimo search engine, as well as in early Semantic Web products, such as metadata extraction on Web Fountain technology [Dill et al 2003], automatic classification and automatic metadata extraction in Semagix Freedom [Sheth et al 2002].

2.2 Formal Semantics

Humans communicate mostly through language. Natural language, however, is inherently ambiguous; semantically, but also syntactically. Computers lack the ability to disambiguate and understand complex natural language. For these reasons, it is infeasible to use natural language as a means for machines to communicate with other machines. As a first step, statements or facts need to be expressed in a way that computers can process them. Semantics that are represented in some well formed syntactic form (governed by syntax rules) is referred to as Formal Semantics. There are some necessary and sufficient features that make a language formal and by association their semantics formal. These features include:

The notions of Model and Model Theoretic Semantics: Expressions in a Formal Language are *interpreted in models*. The structure common to all models in which a given language is interpreted (the *model structure* for the model-theoretic interpretation of the given language) reflects certain basic presuppositions about the “structure of the world” that are implicit in the language.

The Principle of Compositionality: The meaning of an expression is a function of the meanings of its parts and of the way they are syntactically combined. In other words, the semantics of an expression is computed using the semantics of its parts, obtained using an interpretation function.

From a less technical perspective, formal Semantics means machine-processable semantics where the formal language representing the semantics has the above-mentioned features. Basically, the semantics of a statement are unambiguously expressed in the syntax of the statement in the formal language. A very limited subset of natural language is thus made available for computer processing.

Examples of such Semantics are:

- The semantics of subsumption in Description Logics, reflecting the human tendency of categorizing by means of broader or narrower descriptions.
- The semantics of Partonomy, accounting for what is part of an object, not which category the object belongs to.

Description Logics

Recently, description logics (DLs) have been the dominant formalisms for knowledge representation. Although DLs have gained substantial popularity there are some fundamental properties of DLs that can be seen as drawbacks when viewed in the context of the Semantic Web and its future. The *Formal Semantics* of DLs is based on set theory. A concept in Description Logics is interpreted as a set of things that share one required common feature. Relationships between concepts or Roles are interpreted as a subset of the cross-product of the domain of interpretation. This leaves no scope for the representation of degrees of concept membership or uncertainty associated with concept membership.

DL based representation and reasoning for both schema and instance data is being applied in Network Inference's Cerebra [Cerebra] product for such problems as data integration. This product uses a highly optimized tableaux algorithm to speed up ABox reasoning which was the bane of Description Logics. Although a favorable trade-off between computational complexity and expressive power has been achieved there is still the fundamental issue of the inability of DLs to allow for representation of fuzzy and probabilistic knowledge.

2.3 Powerful (Soft) Semantics

The statistical analysis of data allows the exploration of relationships that are not explicitly stated. Statistical techniques give us great insight into a corpus of documents or a large collection of data in general, when a program exists that can actually “pose the right questions to the data”, i.e. analyze the data according to our needs. All derived relationships are statistical in nature and we only have an idea or a likelihood of their validity.

The above mentioned formal knowledge representation techniques give us certainty, that the derived knowledge is correct, provided the explicitly stated knowledge was correct in the first place. Deduction is truth preserving. Another positive aspect of a formal representation is its universal usability. Every system that adheres to a certain representation of knowledge will understand and a well founded formal semantics guarantees that the expressed statements are interpreted the same way on every system. The restriction of expressiveness to a subset of FOL also allows the system to verify the consistency of its knowledge.

But here also lies the crux of this approach. Even though it is desirable to have a consistent knowledge base, it becomes impractical as the size of the knowledge base increases or as knowledge from many sources is added. It is rare that human experts in most scientific domains have a full and complete agreement. In these cases it becomes more desirable that the system can deal with inconsistencies.

Sometimes it is useful to look at a knowledge base as a map. This map can be partitioned according to different criteria, e.g. the source of the facts or their domain. While on such a map the knowledge is usually locally consistent, it is almost impossible and practically infeasible to maintain a global consistency. Experience in developing the Cyc ontology demonstrated this challenge. Hence, a system must be able to identify sources of inconsistency and deal with contradicting statements in such a way that it can still produce derivations that are reliable.

In the traditional bivalent-logic based formalisms we, that is the users or the systems, have to make a decision. Once two contradictory statements are identified, one has to be chosen as the right one. While this is possible in domains that are axiomatized, fully explored or in which statements are true by definition, it is not possible for most scientific domains. In the life sciences for instance, hypotheses have to be evaluated, contradicting statements have promoting data, etc. Decisions have to be deferred until enough data is available that either verifies or falsifies the hypothesis. Nevertheless, it is desirable to express these hypotheses formally to have means to computationally evaluate them on the one hand and to exchange them between different systems on the other.

In order to allow the sort of reasoning that would allow this, the expressiveness of the formalism needs to be increased. It is known that increasing the expressive power of a KR language causes problems relating to computability. This has been the main reason for limiting the expressive power of KR languages. The real power behind human reasoning however is the ability to do so in the face of imprecision, uncertainty, inconsistencies, partial truth, and approximation. There have been attempts made in the past at building KR languages that allow such expressive power.

Major approaches to reasoning with imprecision are 1) probabilistic reasoning, 2) possibilistic reasoning [Dubois et al 1994] and 3) fuzzy reasoning. In [Zadeh 2002], Lotfi Zadeh proposed a formalism that combines fuzzy logic with probabilistic reasoning to exploit the merits of both approaches. Other formalisms have focused on resolving local inconsistencies in knowledge bases, for instance the works by Blair, Kifer, Lukasiewicz, Subrahmanian and others in annotated logic and paraconsistent logic (see [Kifer and Subrahmanian 1992][Blair and Subrahmanian 1989]). In [Lukasiewicz 2004], Lukasiewicz proposes a weak probabilistic logic and addresses the problem of inheritance. Cao [Cao 2000] proposed an annotated fuzzy logic approach that is able to

handle inconsistencies and imprecision; Umberto Straccia has done extensive work on Fuzzy Description logics, see for example [Straccia 2004][Straccia 1998]. With P-CLASSIC, Daphne Koller and others presented an early approach to probabilistic description logics implemented in Bayesian Networks [Koller et al 1997]. Other probabilistic description logics have been proposed by Heinsohn [Heinsohn 1994] and Jaeger [Jaeger 1994]. Early research on Bayesian-style inference on OWL was done by Zhongli Ding [Ding and Peng 2004]. In her formalism, OWL is augmented to represent prior probabilities. However, the problem of inconsistencies arising through inheritance of probability values (see [Lukasiewicz 2004]) is not taken into account.

The combination of probabilistic and fuzzy knowledge under one representation mechanism proposed in [Zadeh 2002] appears to be a very promising approach. Zadeh argues that fuzzy logics and probability theory are “complementary rather than competitive”. Under the assumption that humans tend to linguistically categorize a continuous world into discrete classes, but in fact still perceive it as continuous, fuzzy set theory classifies objects into sets with fuzzy boundaries and gives objects degrees of set membership in different sets. Hence it is a way of dealing with a multitude of sets in a computationally tractable way that also follows the human perception of the world. Fuzzy logic allows us to blur artificially imposed boundaries between different sets. The other powerful tool in soft computing is probabilistic reasoning. Definitely in the absence of complete knowledge of a domain and probably even in its presence, there is a degree of uncertainty or randomness in the ways we see real-world entities interact. OWL as a description language is meant to explicitly represent knowledge and to deductively derive implicit knowledge. In order to use a similar formalism as a basis for tools that help in the derivation of new knowledge, we need to give this formalism the ability to be used in abductive or inductive reasoning. Bayesian type reasoning is a way to do abduction in a logically feasible way by virtue of applying probabilities. In order to use these mechanisms, the chosen formalism needs to express probabilities in a meaningful way, i.e. a reasoner must be able to meaningfully interpret the probabilistic relationships between classes and between instances. The same holds for the representation of fuzziness. The formalism must give a way of defining classes by their membership functions.

A major drawback of logics dealing with uncertainties is the required assignment of prior probabilities and/or fuzzy membership functions. Obviously, there are two ways of doing that, manual assignment by domain experts and automatic assignment using techniques such as machine learning. Manual assignments require the domain expert to assign these values to every class and every relationship. This assignment will be arbitrary, even if the expert has profound knowledge of the domain. Automatic assignments of prior values require a large and representative dataset of annotated instances, and finding or agreeing on what is a representative set is difficult or at times impossible. Annotating instances instead of categorizing them in a top-down approach is

tedious and time consuming. Often, however, the probability values for relationships can be obtained from the dataset using statistical methods, thus we categorize these relationships as implicit semantics.

Another major problem here is that machine learning usually deals with flat categories rather than with hierarchical categorizations. Algorithms that take these hierarchies into account need to be developed. Such an algorithm needs to change the prior values of the superclasses according to the changes in the subclasses, when necessary. Most likely the best way will be a combination of both, when the domain expert assigns prior values that have to be validated and refined using a testing set from the available data.

In the end, powerful semantics will combine the benefits of both worlds. Hierarchical composition of knowledge and statistical analysis; reasoning on available information, but with the advantage over statistical methods that it can be formalized in a common language and that general purpose reasoners can utilize it and with the advantage over traditional formal DL representation that it allows abduction as well as induction in addition to deduction.

It might be argued that more powerful formalisms are already under development, such as SWRL [Straccia 1998], which works on top of OWL. These languages extend OWL by a function free subset of First Order Logics allowing the definition of new rules in the form of Horn clauses. The paradigm is still that of bivalent FOL and the lack of function symbols makes it impossible to define functions that can compute probability values. Furthermore SWRL is undecidable. We believe that abilities to express probabilities and fuzzy membership functions, and the ability to cope with inconsistencies are important. It is desirable, (and some would say necessary) that the inference mechanism is sound and complete with respect to the semantics of the formalism and the language is decidable. Straccia proves this in [Straccia 1998] for a restricted fuzzy DL, Giugno and Lukasiewicz prove soundness and completeness for the probabilistic description logic formalism P-SHOQ(D) in [Giugno and Lukasiewicz 2002]

So far, this powerful semantic and soft computing research has not been utilized in the context of developing the Semantic Web. In our opinion, for this vision to become a reality, it will be necessary to go beyond RDFS and OWL and work towards standardized formalisms that support powerful semantics.

3. Correlating Semantic Capabilities with types of Semantics

Building practical Semantic Web applications (e.g., see [TopQuadrant 2004][Sheth and Ramakrishnan 2003][Kashyap and Shklar 2002] for some examples) require certain core capabilities. A quick look at these core capabilities reveals a sequence of steps towards building such an application. We group this sequence in to two categories as shown in Table 1 and identify the type of semantics utilized by each.

	<i>Capabilities</i>	<i>Implicit Semantics</i>	<i>Formal Semantics</i>	<i>Possible use of Powerful (soft) Semantics</i>
Bootstrapping Phase (building phase)	Building ontologies either automatically or semi-automatically	Analyzing word co-occurrence patterns in text to learn taxonomies/ontologies [Kashyap et al 2003]		Using fuzzy or probabilistic clustering to learn taxonomic structures or ontologies
	Annotation of unstructured content wrt. these ontologies (resulting in semantic metadata)	Analyzing word occurrence patterns or hyperlink structures to associate concept names from and ontology with both resources and links between them [Naing et al 2002]		Using fuzzy or probabilistic clustering to learn taxonomic structures or ontologies OR Using fuzzy ontologies
	Entity Disambiguation	Using clustering techniques or Support Vector Machines (SVM) for Entity Disambiguation [Han et al 2004]	Using an ontology for Entity Disambiguation	Using fuzzy KR mechanisms to represent ontologies that may be used for Disambiguation
	Semantic Integration of different schemas and ontologies	Analyzing the extension of the ontologies to integrate them [Wang et al 2004]	Schema based integration techniques [Castano et al 2001]	
	Semantic Metadata Enrichment (further enriching the existing metadata)	Analyzing annotated resources in conjunction with an ontology to enhance semantic metadata [Hammond et al 2002]		This enrichment could possibly mean annotating with fuzzy ontologies
Utilization Phase	Complex Query processing		Hypothesis validation queries [Sheth et al 2003] or path queries [Anyanwu and Sheth 2002]	
	Question Answering (QA) Systems ¹	Word frequency and other CL techniques to analyze both the question and answer sources [Ramakrishnan et al 2004]	Using <i>Formal</i> ontologies for QA [Atzeni et al 2004]	Providing confidence levels in answers based on fuzzy concepts or probabilistic
	Concept-based search ¹	Analyzing occurrence	Using	

		of words that are associated with a concept, in resources	hypernymy, partonomy and hyponymy to improve search [Townley 2000]	
	Connection and pattern explorer ¹	Analyzing semi-structured data stores to extract patterns (technique in [Kuramochi and Karypis 2004] applied to RDF graphs)	Using ontologies to extract patterns that are meaningful [Aleman-Meza et al 2003]	
	Context-aware retriever ¹	Word frequency and other CL techniques to analyze resources that match the search phrase	Using formal ontologies to enhance retrieval	Using Fuzzy KR mechanisms to represent context
	Dynamic user interfaces ¹		Using ontologies to dynamically reconfigure user interfaces.[Quan and Karger 2004]	
	Interest-based content delivery ¹	Analyzing content to identify concept of content so as to match with interest profile.	User profile will have ontology associated with it which contains concepts on interest	
	Navigational and Research [Guha et al 2003] Search	Navigational searches will need to analyze unstructured content	Discovery style queries [Anyanwu and Sheth 2002] on semi-structured data which is a combination of Implicit and Formal semantics	Fuzzy matches for research search results.

Table 1 Some key Semantic capabilities and the type of Semantics exploited

4. Applications and types of Semantics they exploit.

In this section we describe some research fields and some specific applications in each field. This list is by no means a comprehensive list but rather samples of some research areas that attempt solve problems that are crucial to realizing the Semantic Web vision. We cover Information Integration, Information Extraction/Retrieval, Data Mining and Analytical Applications. We also discuss Entity Identification/Disambiguation in some detail. We associate with each of the techniques in these research areas one or more of the types of semantics we identified earlier.

4.1 Information Integration

There is, now more than ever, a growing need for several Information Systems to interoperate in a seamless manner. This sort of interoperation requires that the syntactic, structural and semantic heterogeneities [Hammer and McLeod 1993][Kashyap and Sheth 1996] between such information systems be resolved. Resolving such heterogeneities has been the focus of a lot of work in Schema Integration in the past. With the recent interest in the Semantic Web there has been a renewed interest in resolving such heterogeneities. A survey of schema matching techniques [Rahm and Bernstein 2001] identifies a wide variety of techniques that are deployed to solve this problem.

4.1.1 Schema Integration

A look at the leaf nodes and the level immediately above it, in this classification tree of schema matching techniques in [Rahm and Bernstein 2001], reveals the combination of the technique used and the type of information about the schema used for matching schemas. Depending on whether the schema or the instances are used to determine the match, the type of information harnessed varies. Our aim is to associate one or more types of semantics (from our classification) with each of the bulleted entries at the leaf nodes of the tree shown. Table 1 below does just that.

	Type of Information used	What does it mean?	Types of Semantics exploited
Linguistic Techniques	Name Similarity	Using canonical name representations, synonymy, hypernymy, string edit distance, pronunciation and N-gram like techniques to match schemas attribute and relation names.	<i>Implicit Semantics</i> are exploited by string edit distance, pronunciation and N-gram like techniques. <i>Formal Semantics</i> are exploited by synonymy etc.
	Description Similarity	Processing Natural language descriptions associated with attributes and relations.	<i>Implicit Semantics</i> are exploited by the NLP techniques deployed.
	Word Frequencies of Key terms	Using relative frequencies of keywords and word combinations at the instance level.	<i>Implicit Semantics</i>
Constraint Based Techniques	Type Similarity	Using information about data types of attributes as an indicator of a match	<i>Formal Semantics</i>

		between schemas.	
	Key Properties	Using foreign keys, part-of relationships and other constraints	<i>Formal Semantics</i>
	Graph Matching	Treating the structure of schemas as graphs algorithms to determine match degree between graphs are used to match schemas.	Combination of <i>Implicit</i> and <i>Formal Semantics</i>
	Value patterns and Ranges	Using ranges of attributes and patterns in the value of attributes as an indicator of similarity between the corresponding schemas.	<i>Implicit Semantics</i>

Table 2 Techniques used for Schema Integration and the type of Semantics they exploit

4.1.2 Entity Identification/Disambiguation (EI/D)

A much harder yet fundamental (and related) problem is that of *Entity Identification/Disambiguation*. This is the problem of identifying, that two entities are in fact either the same but treated as being different or that they are in fact two different entities that are being treated as one entity. Techniques used for *Identification/Disambiguation* vary widely depending on the nature of the data being used in the process. If the application uses unstructured text as a data source then the techniques used for EI/D will rely on *Implicit Semantics*. On the other hand if EI/D is being attempted on semi-structured data the application can, for instance, disambiguate entities based on the properties they have. This implies harnessing the power of Formal or *semi-formal Semantics*. As listed in **Table 1**, the constraint-based techniques are ideally suited for used in EI/D when semi-structured data is being used. Dealing with unstructured data will require the use of the techniques listed under linguistic techniques.

4.2 Information Retrieval and Information Extraction

Let us consider information retrieval applications and the types of data they exploit. Given a request for information by the user, Information Retrieval applications have the task of processing unstructured (text corpus) or loosely connected documents (hyperlinked Web pages) to answer the “query”. There are various flavors of such applications.

4.2.1 Search Engines exploit both the content of Web documents and the structure *implicit* from the hyperlinks connecting one document to the other. In [Kleinberg 1998] the author defines the notions of hubs and authorities in a hyperlinked environment. These notions are crucial to the structural analysis and the eventual indexing of the web. A modification of this approach aimed at achieving scalability is used by Google [Brin and Page 1998]. Google has fairly good precision and recall statistics. However the demands that the Semantic Web places on search engine technology will mean that future

search engines will have to deal with information requests that are far more demanding. In [Guha et al 2003] Guha et al, identify two kinds of searches:

- **Navigational Searches:** In this class of searches, the user provides the search engine a phrase or combination of words which s/he expects to find in the documents. There is no straightforward, reasonable interpretation of these words as denoting a concept. In such cases, the user is using the search engine as a navigation tool to navigate to a particular intended document. Using the domain knowledge as specified in relevant domain ontology can enable an improved semantic search [Townley 2000].
- **Research Searches:** In many other cases, the user provides the search engine with a phrase which is intended to denote an object about which the user is trying to gather/research information. There is no particular document which the user knows about that s/he is trying to get to. Rather, the user is trying to locate a number of documents which together will give him/her the information s/he is trying to find.

We believe that research searches will require a combination of *implicit semantics*, *Formal semantics* and what we refer to as *powerful semantics*.

4.2.2 Question Answering Systems can be viewed as more advanced and more “intelligent” search engine. Current question-answering [Brin and Page 1998][Etzioni et al 2004][Ramakrishnan et al 2004] systems use Natural Language Processing (NLP) and pattern matching techniques to analyze both the question asked of the system and the potential sources of the answers. By comparing the results of these analyses such systems attempt to match portions of the sources of the answer (for instance: Web pages) with the question thereby answering them. Such systems therefore still use data like unstructured text and attempt to extract information from it. In other words the semantics are *implicit* in the text and are extracted from this text. To facilitate question answering, [Zadeh 2003] proposes the use of an epistemic lexicon of world knowledge, which would be represented by a weighted graph of objects with uncertain attributes, in our terminology this is the equivalent of an ontology using *powerful semantics*.

4.3 Data Mining

The goal of Data Mining applications is to find non-trivial patterns in unstructured and structured data.

4.3.1 Clustering: Clustering is defined as the process of grouping *similar* entities or objects together in groups based on some notion of similarity. Clustering is considered as a form of Unsupervised Learning. The applications of clustering use a given similarity metric and, as a result of the grouping of data points into clusters, attempt to use this information (*Implicit semantics*) to learn something about the interactions between the clustered entities. The sort of information sought from the clustered data points may range from simple similarity judgments as in Query -By- Example (QBE) document retrieval systems or systems aimed at extracting more *Formal Semantics* from the underlying data as is the aim of Semi-Automatic Taxonomy Generation.

Semi-Automatic Taxonomy Generation (ATG): As described in [Kashyap et al 2003] the aim of Automated Taxonomy Generation is to hierarchically cluster a document

corpus and extract from the resulting hierarchy of clusters a *sequence* of clusters that best captures all the levels of specificity/generality in the corpus, where this sequence is ordered by the value of the specificity/generality measure. This is then followed by a node label extraction phase where each cluster in the sequence is analyzed to extract from it a set of labels that best captures the topic its documents represents. These sets of labels are then pruned to reduce the number of potential labels for nodes in the final output hierarchy.

4.3.2 Association Rule Mining: An example of an association rule is given in [Agrawal et al 1993][Agrawal and Srikant 1994] as follows: 90% percent of the transactions in a transaction database that involve the purchase of bread and butter together also have the purchase of milk involved. This is an example of an application where occurrence patterns of attribute values in a relational database (*Implicit Semantics*) are converted in association rules (*Formal Semantics*).

4.4 Analytical applications

These come under the purview of applications that support complex query processing. It would be reasonable to hypothesize that search engines of the future will be required to answer analytical or discovery style queries [Guha et al 2003][Anyanwu and Sheth 2002]. This is in sharp contrast with the kinds of information requests today's search engines have to deal with, where the focus is on retrieving resources from the web that may contain information about the desired keyword. In this current scenario the user is left to sift through vast collections of documents and further analyze the returned results. In addition to querying data from the Web, future search engines will also have to query vast metadata repositories. We discuss one such technique in the following section.

4.4.1 Complex Relationship Discovery

As described in [Anyanwu and Sheth 2002] "Semantic Associations capture complex relationships between entities involving sequences of predicates, and sets of predicate sequences that interact in complex ways. Since the predicates are semantic metadata extracted from heterogeneous multi-source documents, this is an attempt to discover complex relationships between objects described or mentioned in those documents. Detecting such associations is at the heart of many research and analytical activities that are crucial to applications in national security and business intelligence." The datasets that Semantic Associations operate over are RDF/RDFS graphs. The Semantics of an edge connecting two nodes in an RDF/RDFS graph are *Implicit*, in the sense that there is no explicit interpretation of the semantics of the edge other than the fact that it is a predicate in a statement (except for *rdfs:subPropertyOf* or edges that represent data type properties – for which there is Model-Theoretic(Formal) semantics). Hence the RDF/RDFS graph is composed of a combination of *Implicit* and *Formal* semantics. The objective of Semantic Associations is therefore to find all contextually relevant edge sequences that relate two entities. This is in effect, an attempt to combine the *Implicit* and *Formal* semantics of the edges in the RDF/RDFS graph in conjunction with the Context of the query to determine the *multifaceted (multivalent) semantics* of a set of "connections" between entities. We view this *multivalent semantics* as a form of

Powerful semantics. In the context of search, Semantic Associations can be thought of as a class of research searches or discovery style searches.

Conclusions

We have identified three types of semantics and in the process assorted key capabilities required to build practical Semantic application involving Web resources. We have also qualified each of the listed capabilities with one or more types of semantics, as in Table 1. This table reveals some very basic problems that need to be solved for an application to be termed “Semantic”. It is clear from this table that *Entity Disambiguation*, *Question Answering Capability*, *Context-based retrieval* and *Navigational and Research (Discovery) style query capability* require the use of all three types of semantics. Therefore by focusing research efforts in representation mechanisms for *powerful (soft) semantics* in conjunction with fuzzy/probabilistic computational methods supporting techniques that use Implicit and Formal semantics it might be possible to solve some of the difficult but practically important problems. In our opinion the current view taken by the Semantic Web community is heavily biased in favor of *Formal Semantics*. It is clear however, that the focus of effort in pursuit of the Semantic Web vision needs to move towards an approach that encompasses all three types of semantics in representation, creation methods and analysis of knowledge. If the capabilities that we identified do in fact turn out to be fundamental capabilities that make an application Semantic, these capabilities could serve as a litmus test or a standard against which other applications may be measured to determine if they are “Semantic applications”.

References

- [Agrawal et al 1993] R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In P. Buneman and S. Jajodia, editors, Proceedings of the 1993.
- [Agrawal and Srikant 1994] R. Agrawal, and R. Srikant. Fast Algorithms for Mining Association Rules, Proceedings of the 20th International Conference on Very Large Databases, Santiago, Chile, 1994
- [Aleman-Meza et al 2003] B. Aleman-Meza, C. Halaschek, S. Sahoo, [Terrorist Related Assessment using Knowledge Similarity](#), LSDIS Lab Technical Report, December 2003.
- [Anyanwu and Sheth 2002] Kemafor Anyanwu, Amit P. Sheth: [The p Operator: Discovering and Ranking Associations on the Semantic Web](#). SIGMOD Record 31(4): 42-47 (2002)
- [Atzeni et al 2004] Paolo Atzeni, Roberto Basili, Dorte H. Hansen, Paolo Missier, Patrizia Paggio, Maria Teresa Pazienza, Fabio Massimo Zanzotto Ontology-based question answering in a federation of university sites: the MOSES case study 9th International Conference on Applications of Natural Language to Information Systems (NLDB'04) Manchester (United Kingdom), June 2004

- [Barbara et al 1992] D. Barbará, H. Garcia-Molina and D. Porter. The Management of Probabilistic Data. IEEE Transactions on Knowledge and Data Engineering, Volume 4 , Issue 5 (October 1992), Pages: 487 - 502
- [Blair and Subrahmanian 1989] H.A. Blair, V.S. Subrahmanian. Paraconsistent logic programming. Theoretical Comp. Sci. 68, 1989, 135-154
- [Brin and Page 1998] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In Proceedings of the Seventh International World Wide Web Conference, 1998.
- [Cao 2000] T.H.Cao, Annotated Fuzzy Logic Programs, International Journal on Fuzzy Sets and Systems 113 (2000) 277-298
- [Castano et al 2001] S. Castano, V.D. Antonellis and S.D.C. Vimercati, Global viewing of heterogeneous data sources , IEEE Transactions on Knowledge and Data Engineering 13(2) (2001) 277-297
- [Dill et al 2003] Stephen Dill, Nadav Eiron, David Gibson, Daniel Gruhl, R. Guha, Anant Jhingran, Tapas Kanungo, Sridhar Rajagopalan, Andrew Tomkins, John A. Tomlin, and Jason Y. Zien. Semtag and seeker: bootstrapping the semantic web via automated semantic annotation. In Proceedings of the Twelfth International Conference on World Wide Web, pages 178--186. ACM Press, 2003.
- [Ding and Peng 2004] Zhongli Ding, Yun Peng. A Probabilistic Extension to Ontology Language OWL. Proceedings of the Hawai'i International Conference on System Sciences, January 5-8, 2004, Big Island, Hawaii.
- [Dubois et al 1994] D. Dubois, J. Lang, H. Prade. Possibilistic Logic. In Dov M. Gabbay et al. (Eds), Handbook of Logic in Artificial Intelligence and Logic Programming, Vol. 3, Oxford University Press, Oxford, 1994, pp 439 - 514
- [Etzioni et al 2004] Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, Alexander Yates: Web-scale information extraction in knowitall: (preliminary results). WWW 2004: 100-110
- Giugno and Lukasiewicz 2002] Rosalba Giugno, Thomas Lukasiewicz. P-SHOQ(D): A Probabilistic Extension of SHOQ(D) for Probabilistic Ontologies in the Semantic Web. Proceedings of the European Conference on Logics in Artificial Intelligence, 2002
- [Gruber 2003] T. Gruber. It Is What It Does: The Pragmatics of Ontology, invited talk at Sharing the Knowledge- International CIDOC CRM Symposium, March 26-27, Washington, DC, <http://tomgruber.org/writing/cidoc-ontology.htm>
- [Guha et al 2003] R. Guha, R. McCool, and E. Miller. Semantic search. In The Twelfth International World Wide Web Conference, Budapest, Hungary, May 2003.
- [Hammer and McLeod 1993] Joachim Hammer, Dennis McLeod: An Approach To Resolving Semantic Heterogeneity In A Federation Of Autonomous, Heterogeneous Database Systems, Journal for Intelligent and Cooperative Information Systems (1993)
- [Hammond et al 2002] B. Hammond, A. Sheth, and K. Kochut, [Semantic Enhancement Engine: A Modular Document Enhancement Platform for Semantic Applications over Heterogeneous Content, in Real World Semantic Web Applications](#), V. Kashyap and L. Shklar, Eds., IOS Press, December 2002, pp. 29-49.
- [Heinsohn 1994] Jochen Heinsohn: Probabilistic Description Logics. UAI 1994: 311-318.

- [Han et al 2004] Hui Han, C. Lee Giles, Hongyuan Zha, Cheng Li, and Kostas Tsioutsoulouklis. "Two Supervised Learning Approaches for Name Disambiguation in Author Citations" , in Proceedings of ACM/IEEE Joint Conference on Digital Libraries (JCDL 2004), pages 296-305, 2004.
- [Handschuh et al 2002] Siegfried Handschuh, Steffen Staab, Fabio Ciravegna S-CREAM - Semi-automatic CREATION of Metadata In Proc. of the European Conference on Knowledge Acquisition and Management - EKAW-2002. Madrid, Spain, October 1-4, 2002. Springer, 2002.
- [Hjorland 1998] Birger Hjorland Information retrieval, text composition, and semantics. Knowledge Organization 25(1/2):16-31, 1998
- [Jaeger 1994] Manfred Jaeger: Probabilistic Reasoning in Terminological Logics. KR 1994: 305-316.
- [Kashyap et al 2003] V. Kashyap, C. Ramakrishnan, C. Thomas, D. Bassu, T. C. Rindflesch and A. Sheth TaxaMiner: [An Experimentation Framework for Automated Taxonomy Bootstrapping](#). Technical Report number UGA-CS-TR-04-005, Computer Science Dept., University of Georgia
- [Kashyap and Sheth 1996] Vipul Kashyap, Amit Sheth: [Semantic Heterogeneity in Global Information Systems: The Role of Metadata, Context and Ontologies](#), Cooperative Information Systems 1996
- [Kashyap and Shklar 2002] V. Kashyap and L. Shklar (Eds). Real World Semantic Web Applications Volume 92 Frontiers in Artificial Intelligence and Applications, 2002, 208 pp.
- [Kifer and Subrahmanian 1992] Michael Kifer and V.S. Subrahmanian. Theory of generalized annotated logic programming and its applications. Journal of Logic Programming, 12:335-367, 1992
- [Kleinberg 1998] J. Kleinberg. Authoritative sources in a hyperlinked environment. In Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 1998.
- [Koller et al 1997] Daphne Koller, Alon Levy, and Avi Pfeffer. P-CLASSIC: A tractable probabilistic description logic. Proc. Of the 14th Nat. Conf. on Artificial Intelligence (AAAI-97), 390-397, 1997
- [Kuramochi and Karypis 2004] M. Kuramochi and G. Karypis. Finding frequent patterns in a large sparse graph. In SIAM International Conference on Data Mining (SDM-04), 2004.
- [Lukasiewicz 2004] Thomas Lukasiewicz: Weak Nonmonotonic probabilistic Logics, Principles of Knowledge Representation and Reasoning (KR) 2004
- [Maedche and Staab 2001] Alexander Maedche, Steffen Staab: Ontology Learning for the Semantic Web. IEEE Intelligent Systems 16(2): 72-79 (2001)
- [Naing et al 2002] Myo-Myo Naing, Ee-Peng Lim, Dion Hoe-Lian Goh: Ontology-based Web Annotation Framework for HyperLink Structures. WISE Workshops 2002: 184-193
- [Omelayenko 2001] B. Omelayenko. Learning of Ontologies for the Web: the Analysis of Existent approaches. In Proceedings of the International Workshop on Web Dynamics, 2001.
- [Patil et al 2004] A. Patil, S. Oundhakar, A. Sheth, K. Verma, [METEOR-S Web service Annotation Framework](#), Proceeding of the World Wide Web Conference, New York, NY, May 2004, pp. 553-562.
- [Quan and Karger 2004] Dennis Quan and David R. Karger. How to Make a Semantic Web Browser in WWW 2004
- [Rahm and Bernstein 2001] Erhard Rahm, Philip A. Bernstein. A Survey of Approaches to Automatic Schema Matching. In VLDB Journal 10: 4, 2001

- [Ramakrishnan et al 2004] Ganesh Ramakrishnan, Soumen Chakrabarti, Deepa Paranjpe, Pushpak Bhattacharya. Is question answering and acquired skill? Proceedings of the 13th international conference on World Wide Web 2004
- [Sheth et al 2002] A. Sheth, C. Bertram, D. Avant, B. Hammond, K. Kochut, Y. Warke, [Managing Semantic Content for the Web](#), IEEE Internet Computing, July/August 2002, pp. 80-87.
- [Sheth et al 2003] Amit P. Sheth, Sanjeev Thacker, Shuchi Patel: [Complex relationships and knowledge discovery support in the InfoQuilt system](#). VLDB J. 12(1): 2-27 (2003)
- [Sheth and Ramakrishnan 2003] Amit P. Sheth, Cartic Ramakrishnan: [Semantic \(Web\) Technology In Action: Ontology Driven Information Systems for Search, Integration and Analysis](#). IEEE Data Eng. Bull. 26(4): 40-48 (2003)
- [Straccia 2004] Umberto Straccia. Uncertainty and Description Logic Programs: A Proposal for Expressing Rules and Uncertainty on Top of Ontologies, Technical Report 2004-TR-14
- [Straccia 1998] Umberto Straccia. A Fuzzy Description Logic, Proceedings of AAAI-98, 15th Conference of the American Association for Artificial Intelligence
- [Townley 2000] J. Townley, The Streaming Search Engine That Reads Your Mind, August 10, 2000. <http://smw.internet.com/gen/reviews/searchassociation/>
- [Uschold 2003] Michael Uschold: Where are the Semantics in the Semantic Web? In Artificial Intelligence, Fall 2003
- [Wang et al 2004] Jiying Wang, Ji-Rong Wen, Frederick H. Lochovsky, Wei-Ying Ma: Instance-based Schema Matching for Web Databases by Domain-specific Query Probing. VLDB 2004
- [Woods 2004] William A. Woods: "Meaning and Links: A Semantic Odyssey". Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR2004), June 2-5, 2004. pp. 740-742
- [Yen 1991] John Yen: Generalizing Term Subsumption Languages to Fuzzy Logic. IJCAI 1991: 472-477
- [Zadeh 2002] Lotfi A. Zadeh. Toward a perception-based theory of probabilistic reasoning with imprecise probabilities. In Journal of Statistical Planning and Inference 105 (2002) 233-264.
- [Zadeh 2003] Lotfi A. Zadeh. From Search Engines to Question-Answering Systems - The Need for New Tools. Proceedings of the 1st Atlantic Web Intelligence Conference, 2003
- [Cerebra] http://www.networkinference.com/Assets/Products/Cerebra_Server_DataSheet.pdf
- [TopQuadrant 2004] http://www.topquadrant.com/documents/TQ04_Semantic_Technology_Briefing.PDF