Assessing the Effectiveness of the DAML Ontologies for the Semantic Web

Andrew Burton-Jones[1], Veda C. Storey[1], Vijayan Sugumaran[2], Punit Ahluwalia[1]

[1] J. Mack Robinson College of Business
Georgia State University, Box 4015
Atlanta, GA 30302
{abjones, vstorey, hispaa}@cis.gsu.edu
[2] School of Business Administration
Oakland University
Rochester, MI 48309
sugumara@oakland.edu

Abstract: The continued growth of the World Wide Web makes the retrieval of relevant information for a user's query increasingly difficult. Current search engines provide the user with many web pages, but varying levels of relevancy. In response, the Semantic Web has been proposed to retrieve and use more semantic information from the web. Our prior research has developed an intelligent agent to automate the processing of a user's query while taking into account the query's context. The intelligent agent uses WordNet and the DARPA Agent Markup Language (DAML) ontologies to act as surrogates for understanding the context of terms in a user's query. This research develops a set of syntactic, semantic, and pragmatic constructs to assess the effectiveness of the DAML ontologies so that the intelligent agent can select the most useful ontologies. These constructs have been implemented in a tool called the "Ontology Auditor" for use by the intelligent agent.

1. Introduction

The explosive growth of the world-wide-web (WWW) makes it increasingly difficult for users to retrieve relevant web pages for queries. To address this problem, the Semantic Web has been proposed to extend the WWW by giving information well defined meaning. The Semantic Web relies heavily on ontologies to provide taxonomies of domain specific terms and inference rules for a body of knowledge that serves as a surrogate for semantics [BHL01].

Our prior research [SBS02, BPS02] developed a multi-agent system called ISRA (Intelligent Semantic-Web Retrieval Agent) that implements a heuristics-based methodology for intelligent retrieval of information from the web. ISRA takes into account the semantics in users' requests by parsing a natural language query and using information from WordNet [Mi90] and DAML ontologies (www.daml.org/ontologies) to serve as surrogate for the semantics of the terms in a user's query.

ISRA follows the tradition of query expansion in Information Retrieval (IR) [Vo94, Gr01]. Given a set of query terms, T1, ISRA expands the query using lexically related terms (from WordNet) and domain-related terms (from the DAML ontology library), to contextualize the query so that it can obtain more relevant results. The original query terms form root nodes in a semantic network. The network expands by adding related terms, and shrinks by removing terms of unwanted context. By doing so, ISRA iterates towards a more effective set of query terms, T2. The effectiveness of the methodology clearly depends on the quality of knowledge sources from which it obtains new terms. The DAML ontologies were developed specifically for the Semantic Web to provide domain knowledge. However, the ontologies have different degrees of completeness and applicability. They need to be evaluated so systems that use them can trust them.

The objective of this research is: to analyze the DAML ontologies, as representative of domain ontologies, to assess their effectiveness. This analysis is based upon principles from semiotics from syntactic, semantic, and pragmatic perspectives. The contributions of the research are to: a) provide a set of constructs for evaluating ontologies, and b) apply the constructs to assess the DAML domain ontologies.

2. Related Research

2.1 Semantic Web

The Semantic Web is an extension of the current web, in which the semantics of terms found in web pages will be explicitly defined using online machine-readable knowledge bases called ontologies [BHL01]. The aim of the Semantic Web effort is to create the infrastructure necessary for the web to become ‘machine-readable’ in the sense that it allows agents to interpret and reason about semantics on web pages (which is not possible on the current web) and, thus, perform complex and intelligent tasks [BHL01]. Researchers are developing new languages and authoring tools to specify Web page semantics [Fe01], determining how applications and intelligent agents might utilize the Semantic Web [He01], and investigating approaches for querying the Semantic Web [HH01]. The infrastructure to support the Semantic Web is being developed [Ca02, Ie01]. The DARPA ontology library contains approximately 200 ontologies developed for this purpose.

2.2 Ontologies for the Semantic Web

A barrier to the Semantic Web is the lack of high quality ontologies [He01]. An ontology is a set of terms, relations between terms, and inference rules for a topic [Gr93]. Ontologies and agents are predicted to evolve on the Semantic Web. Rather than having comprehensive ontologies (e.g., Cyc [GL94]), smaller, domain-specific ontologies will likely predominate [He01, SH01]. As Hendler [He01] predicts:

"The Semantic Web…will not primarily consist of neat ontologies that expert AI researchers have carefully constructed. I envision a complex Web of semantics ruled by the same sort of anarchy that rules the rest of the Web."

This would not be a problem if there were an accepted methodology for ontology creation. However, there is currently no accepted way to develop ontologies [MS01]. Many ontologies "embody systematic errors or massive ontological unclarities …predestined to yield an end-result that is of dubious merit" [Sm03]. Guarino and Welty [GW02] suggest that many ontology developers fail to understand basic ontological relationships. Although some metrics for ontology development have been suggested [CG00], much more work is needed in this regard [We02].

Agents that use ontologies containing incomplete, inaccurate, or misleading knowledge cannot perform tasks successfully. A poor quality ontology can reduce *efficiency* by requiring superfluous ontologies to be read and can reduce *effectiveness* by providing the agent with poor information. Metrics are needed to evaluate the quality of ontologies, to support their design, and inform their use.

3. Metrics Suite For Evaluating Domain Ontologies

Weber [We02] distinguishes between *formal* ontologies, used to describe reality in general, and *material* ontologies, used to describe specific aspects of reality. Material ontologies are:

- Application ontologies - specify the definitions needed to model the knowledge required by an application
- Domain ontologies - specify conceptualizations specific to a domain
- Generic ontologies - specify conceptualizations generic to several domains, and
- Representation ontologies - specify conceptualizations that underlie knowledge representation formalisms (e.g., frames).

This section presents a metrics suite to evaluate material *generic*, *domain*, or *application* ontologies. To be valid, the metric suite must be derived from theory. One way to evaluate material ontologies is to use a formal ontology as a benchmark [GW02]. In conceptual modeling, for example, [CK94] and [WW95] use Bunge's [Bu77] formal ontology to evaluate various representation ontologies (e.g., ER, OO, etc). While formal ontologies can be very useful, the problem is that there is no way to validate the particular formal ontology chosen as the benchmark [We02, Sm03]. Furthermore, they are often too high-level or philosophical to capture pragmatic issues. This research, therefore, adopts Stamper's et al. [St00] semiotic framework, a more general theoretical framework derived from linguistics, and explicitly includes pragmatic issues.

Semiotics studies the properties of signs, for example, whether the sign used for "Chair" is good or bad, clear or unclear. Ontologies use symbols, or signs, to describe terms. For example, a *Computer Science* DAML ontology (http://www.daml.org/ontologies/64) includes:

```
<Class ID="Chair">
    <label>chair</label>
    <subClassOf resource="#AdministrativeStaff" />
    <subClassOf resource="#Professor" />
</Class>
```

Several signs are manifest in this script. The terms “Class” and “subClassOf” are signs with meaning in the DARPA Agent Markup Language (DAML). The terms “Chair” and “Professor” are signs for things in the real world the ontology describes. Stamper et al. [St00] provide a 6-level semiotic framework to support the analysis of signs.

- *Social*: meaning of signs in regard to its potential and actual social consequences
- *Pragmatic*: relationships between signs and their consequences
- *Semantic*: meaning of signs or the mapping between signs and what they represent
- *Syntactic*: relationship among signs including their formal logical arrangement
- *Empirical*: communication properties of signs including channel capacity, noise, entropy
- *Physical*: physical representation of signs in hardware, components, etc.

Overall Metric	Metrics Suite	Attributes	Description
Ontology Quality	Syntactic Quality	Lawfulness	Correctness of syntax
		Richness	Breadth of syntax used
	Semantic Quality	Interpretability	Meaningfulness of terms
		Consistency	Consistency of meaning of terms
		Clarity	Average number of word senses
	Pragmatic Quality	Comprehensiveness	Number of classes and properties
		Accuracy	Accuracy of information
		Relevance	Relevance of information for a task
	Social Quality	Authority	Extent to which other ontologies rely on it
		History	Number of times ontology has been used

Table 1: Proposed Metric Suite for Ontological Auditing

Table 1 proposes a suite of metrics for evaluating ontologies based upon the semiotic framework. The suite consists of metrics for syntactic, semantic, pragmatic, and social quality. Metrics for physical and empirical quality are not included as they are more applicable to implementation details. The metrics can be used by a person (real auditor/developer) or machine (virtual auditor). The suite is general enough to assess ontologies irrespective of how they are represented and implemented. By including multiple dimensions, the metrics can be weighted according to the requirements of specific applications or domains. The intent is to use the metrics to evaluate ontologies *before* they are used. Agents, such as ISRA, can use the evaluations to decide which ontologies to use.

Ontology engineers can use the metrics as design principles when building ontologies. A complete evaluation system would have two additional dimensions: 1) a tool to evaluate the quality of specific elements of an ontology for use in a specific task, and 2) a learning mechanism to adjust the evaluations after using the ontologies. Table 2 details how each metric for the *a priori* evaluation is calculated. Overall quality (Q) is a weighted function of its syntactic (S), semantic (E), pragmatic (P), and social (O) qualities (i.e., $Q = b1 \times S + b2 \times E + b3 \times P + b4 \times O$). The weights sum to unity, and currently, are set by the user, the application, or else assumed equal. Ongoing research is investigating empirically derived weights.

Syntactic Quality (S) measures the quality of the ontology according to the way it is written. *Lawfulness* refers to the degree to which an ontology language's rules have been complied. Not all ontology editors have error-checking capabilities; without correct syntax, the ontology cannot be read and used. *Richness* refers to the proportion of features in the ontology language that have been used in an ontology (e.g., whether it includes terms and axioms, or just terms). Richer ontologies are more valuable to the user (e.g., agent). Ongoing research is testing the value of adjusting this metric by the frequency of use of each feature.

Semantic Quality (E) evaluates the meaning of terms in the ontology library. Three attributes are proposed: interpretability, consistency, and clarity. *Interpretability* refers to the meaning of terms (e.g., classes and properties) in the ontology. This is achieved by checking that the words used by the ontology exist in an online lexical database (e.g., WordNet; [Fe98]) (http://www.cogsci.princeton.edu/~wn). Preferably, the knowledge provided by the ontology can map into meaningful concepts in the real world. *Consistency* is whether terms have a consistent meaning in the ontology. For example, if an ontology claims that X is a subclass_of Y, and that Y is a property_of X, X and Y have inconsistent meanings and are of no semantic value. *Clarity* is whether the context of terms is clear. For example, if an ontology claims that class "Chair" has property "Salary," an agent must know this describes academics, not furniture.

Pragmatic Quality (P) refers to the ontology's *usefulness* for users or their agents, irrespective of syntax or semantics. Three criteria are used. *Accuracy* is whether the claims an ontology makes are 'true.' This is difficult to determine automatically without a learning mechanism. Currently, a domain expert assesses accuracy. *Comprehensiveness* is a measure of the size of the ontology. Larger ontologies are more likely to be complete representations of their domains, and provide more knowledge to the agent. *Relevance* is whether the ontology satisfies the agent's specific requirements. This requires some knowledge of the agent's needs prior to evaluation. This metric is coarse because it checks for whether the ontology contains the *type* of information the agent uses (e.g., property, subclass, etc) rather than the *specific* semantics needed.

Social quality (O) reflects that agents and ontologies exist in communities. The *authority* of an ontology is the number of other ontologies that link to it (define their terms using its definitions). More authoritative ontologies signal that the knowledge they provide is accurate or useful. *History* is the number of times the ontology is accessed. Ontologies with longer histories are considered more dependable.

Attributes	Determination
Overall Quality (Q)	$Q = b_1 \cdot S + b_2 \cdot E + b_3 \cdot P + b_4 \cdot O$
Syntactic Quality (S)	$S = b_1 \cdot S_L + b_2 \cdot S_R$
Lawfulness (S_L)	Let X be total syntactical rules. Let X_b be total breached rules. Let N_S be the number of statements in the ontology. Then $S_L = X_b / N_S$.
Richness (S_R)	Let Y be the total syntactical features available in ontology language. Let Z be the total syntactical features used in this ontology. Then $S_R = Z/Y$.
Semantic Quality (E)	$Q = b_1 \cdot E_I + b_2 \cdot E_C + b_3 \cdot E_A$
Interpretability (E_I)	Let C be the total number of terms used to define classes and properties in ontology. Let W be the number of terms that have a sense listed in WordNet. Then $E_I = W/C$.
Consistency (E_C)	Let I = 0. Let C be the number of classes and properties in ontology. $\forall C_i$, if meaning in ontology is inconsistent, I+1. $\therefore$ I = number of terms with inconsistent meaning. Ec = I/C.
Clarity (E_A)	Let C_i = name of class or property in ontology. $\forall$ C_i, count A_i , (the number of word senses for that term in WordNet). Then $E_A = A/C$.
Pragmatic Quality (P)	$Q = b_1 \cdot P_O + b_2 \cdot P_U + b_3 \cdot P_R$
Comprehensiveness (P_O)	Let C be the total number of classes and properties in ontology. Let V be the average value for C across entire library. Then $P_O = C/V$.
Accuracy (P_U)	Let N_S be the number of statements in ontology. Let F be the number of false statements. $P_U = F/N_S$. Requires evaluation by domain expert and/or truth maintenance system.
Relevance (P_R)	Let N_S be the number of statements in the ontology. Let S be the type of syntax relevant to agent. Let R be the number of statements within N_S that use S. $P_R = R / N_S$.
Social Quality (O)	$Q = b_1 \cdot O_T + b_2 \cdot O_H$
Authority (O_T)	Let an ontology in the library be O_A. Let the set of other ontologies in the library be L. Let the total number of links from ontologies in L to O_A be K. Let the average value for K across ontology library be V. Then $O_T = K/V$.
History (O_H)	Let the total number of accesses to an ontology be A. Let the average value for A across ontology library be H. Then $O_H = A/H$.

Table 2: Determination of Metric Values

4. Implementation Of Ontology Auditor Agent

The metrics were implemented in an automated ontology auditor, shown in Figure 1. The auditor is an agent in that it operates autonomously to assess the goodness of an ontology before that ontology is used by ISRA. The auditor agent is comprised of three components: a) search component, b) rating component, and c) publishing component. The ISRA system interfaces with the ontology auditor agent and can request it to evaluate ontologies in a particular domain. The auditor agent returns the scores for the ontologies and the ISRA system can choose the appropriate ontologies to use.

The auditor agent carries out a three-step process. First, the *search component* searches for ontologies in specified domains (e.g., the DARPA ontology library) based on their common ontology-language file extensions (e.g., file.daml.) Second, the *rating component* assesses each ontology using online web services (e.g., WordNet) and rules for each metric. The rating component gives a rating for each metric and an overall average rating. It does not, however, give a recommendation. Third, the *publishing component* publishes its assessment of the ontology in a designated location so other agents can read it. The Ontology Auditor Agent has been implemented in C++ and applied to the DAML ontologies. The auditor agent utilizes WordNet web service to determine word senses of terms. The agent also uses a knowledge base that contains the ontology metrics and rules to be used in evaluating ontologies.

4.1 Search Component

The search component continually evaluates ontologies and adds them to the evaluated ontologies. It can also evaluate new ontologies on demand, e.g., if ISRA requests knowledge on a domain that is not covered by the auditor's published list of evaluated ontologies. It contains meta-information about the ontologies and their domains.

4.2 Rating Component

The rating component contains a module for each metric, which are described below.

Lawfulness Module Lawfulness is measured by searching for instances of incorrect syntax used within the ontology. For example, the lawfulness module can retrieve web pages containing DAML ontologies from http://www.daml.org/ontologies/uri.html. Using a DAML markup checker (http://www.daml.org/validator/), these pages are then parsed for syntactic errors and the number of errors detected reported. For example, the Calendar ontology (http://www.daml.org/ontologies/134) returns the following error:

```
ParseException: {E201} Syntax error when processing <EOF>.
Input to RDF parser ended prematurely.
```

Richness Module The number of daml properties used to describe each ontology provides a measure of richness. The score counts the different features used. The module imports data from a table of daml features used in each ontology at http://www.daml.org/ontologies/features. An ontology lacking in richness is the *Instance* ontology (http://www.daml.org/ontologies/77), which only uses two types of terms (*subclass* and *type*). In contrast, the *Research Information* ontology (http://www.daml.org/ontologies/221) contains 21 different types of terms, including class and subclass, property and subproperty, intersection, inverse, disjoint, domain, range, and cardinality.

Interpretability Module Interpretability is measured by checking WordNet to determine if the terms in an ontology are meaningful. The daml pages are parsed for classes and properties. If the class or property names are phrases (e.g., "LiquefiedGasCarrierWithTankOnDeck") these are modified before searching WordNet.

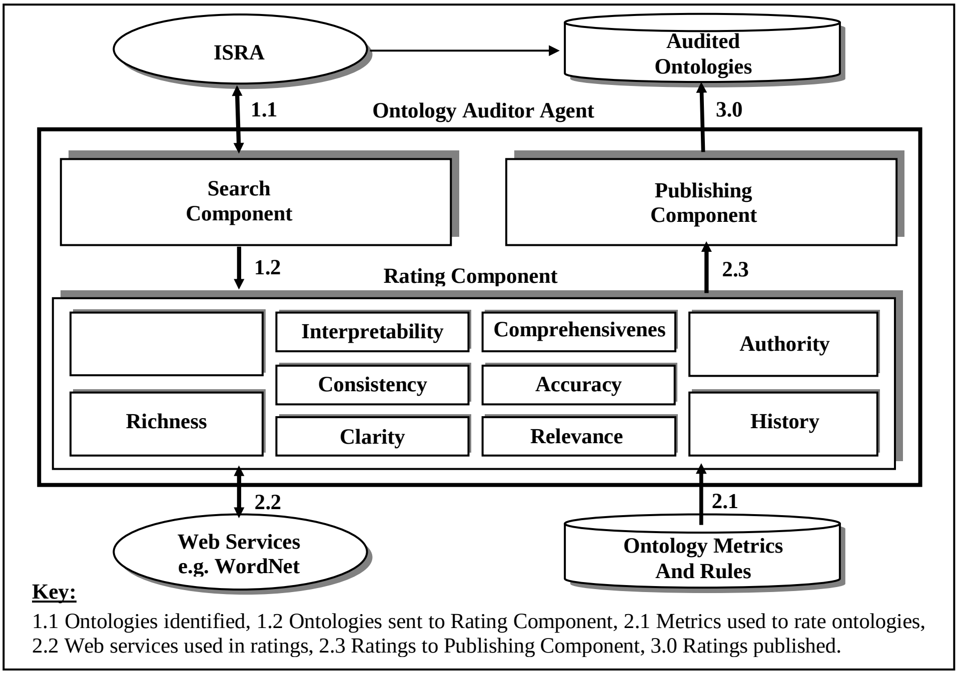


Figure 1: Ontology Auditor Agent Architecture

Consistency Module This module checks the internal consistency of ontologies. Inconsistencies occur when the same term is used in two or more ways in one ontology. For example, if term (X) is listed as a sub class of another term (Y), then it would be inconsistent if X also appeared as a super-class of Y elsewhere in the ontology. Similarly, if X is a property of Y, it should not also be a subclass of Y. Inconsistencies should be detected to avoid reaching incorrect inferences. In the *profiling* ontology (http://www.daml.org/ontologies/237) "gender" is listed as both a property and a class.

Clarity Module This is an extension of the interpretability module. Class and property names in WordNet are either single words (e.g., person) or phrases (e.g., firstName). The *clarity* metric checks for the number of senses in WordNet for the class or property name as a whole (whether a single word or phrase). Interpretability checks for the existence of the individual words (e.g., person, first, and name). An example of an 'unclear' word, is the class "break" (found in the *Agenda* ontology http://www.daml.org/ontologies/238) which is highly polysemous, having 15 different word senses in WordNet. Ideally, the ontology would use words with precise meanings (e.g., "intermission," which has only 2 senses) because automated approaches for resolving the context of polysemous words remains difficult [Mi96], labor consuming [LSG03], and unsatisfactory [GS01, GS02].

Comprehensiveness Module The total count of classes and properties in an ontology is reported. This is also an extension of the module used for interpretability. As shown in Table 3, most DAML ontologies are small (1-20 terms). However, some very large ontologies (e.g., Cyc, http://www.daml.org/ontologies/225) include over 2700 terms.

Accuracy Module This module tests whether knowledge given by the ontology is true. As an example, the Computer Science ontology (http://www.daml.org/ontologies/225) states that 'staff' is a subclass of a department and, therefore, inherits the department's properties ('has_staff,' 'has_courses,' and 'has_URL'). This is inaccurate; staff are part of (not a subclass of) a department, so should not inherit its properties. Accuracy is determined by checking knowledge in the ontology against existing knowledge known to be true. The accuracy module is not implemented at present, so requires a domain expert. We are investigating how this could be performed automatically via a truth maintenance system for ontologies that provide axioms.

Relevance Module The relevance module examines the degree to which the ontology provides information of the type needed by a specific application. ISRA, for instance, primarily uses class/subclass relationships in the DAML library. Other applications may require information on properties. This module identifies the type of knowledge the ontology provides that may be useful to different applications. It investigates four types of information: class/subclass, property, cardinality, and a broad category called 'set knowledge' that includes restrictions, inverse, union, disjoint, complement, etc.

Authority Module Authority is the number of references to an ontology from other ontologies. The daml page for each ontology is first parsed for references to other ontologies. The number of references to each ontology in the library is counted. An example of authority can be seen in the following definition of a class in the *Computer Science* ontology (http://www.daml.org/ontologies/65) that references "univ1.0.daml" for its definition of Faculty.

```
<Class ID="Faculty">
<equivalentToresource="http://www.cs.umd.edu/projects/
                 plus/DAML/onts/univ1.0.daml#Faculty"/>
```

History Module Implementing the history module requires either (a) the ontology library to publish frequency of access to each ontology (not yet produced by the DAML library), or (b) the agent to track the number of times it uses each ontology. This latter method will be used to implement history in the ISRA system.

4.3 Publishing Component

The results are stored in the Audited Ontologies database. The publishing component dynamically creates an html document that incorporates the assessment scores and related information using a predefined template. These pages can be accessed by human and software agents, and will be made available once all dimensions are implemented.

5. Analysis of DAML Ontologies

The prototype auditor was applied to the DAML ontologies, with all but the consistency, accuracy, authority, and history modules implemented. The results are shown Table 3.

5.1 Syntax

The results for *lawfulness* in Table 3 indicate that only eighteen percent of the DAML ontologies are free from syntax errors. Thirty-five percent of the ontologies contain 1-50 syntax errors, and over twenty percent contain greater than 50 errors. Further analysis indicates that as a proportion of the classes, properties, and instances per ontology, the average number of syntax errors per statement was 1.1 (i.e., over one syntax error per fact provided by the ontology). Improved tools would help prevent and detect syntax errors in these languages.

The results for *richness* show that over sixty percent of the ontologies use 1-10 different types of syntax and twenty percent use 10-20. The DAML web site provides 67 types of syntax. Some of these merely reflect language differences e.g., *subclass* available either using an rdf or daml syntax. Nevertheless, 51 distinct types of syntax are available. None of the ontologies uses even half of the available syntax. This indicates that either the ontologies are underdeveloped, or many of the syntactical features are unnecessary.

5.2 Semantics

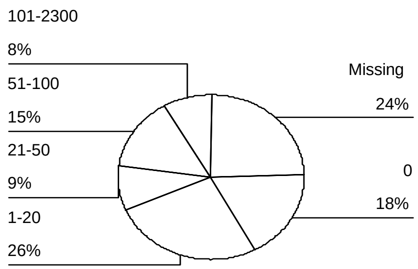
With respect to *clarity,* almost forty percent of the ontologies contained words that were clear (had between one and four senses on average). Nearly 25% percent of the ontologies, however, contained classes and properties that were highly polysemous, with an average of more than four senses for each class and property. Such ontologies could cause problems for agents needing knowledge about a specific sense of a term. Another 10% of the ontologies contained less than one sense per class or property name (i.e., included meaningless names) (as determined by WordNet). The results for *interpretability* further showed that a surprisingly large number of ontologies contained meaningless terms. Only approximately 20% of the ontologies had more than 80% of their terms existing in WordNet. Across the entire library, less than half of the words in the library (43%) had meaning in WordNet. Overall, ontology designers appear to frequently use terms that do not exist in common English. An intuitive explanation is that these terms are merely highly domain-specific. While the auditor could be expanded to consult domain specific thesauri, our analysis found that many ontologies simply use acronyms, non-words, misspelled words, or non-English words. Tools could help ontology engineers use precise semantics when constructing ontologies.

5.3 Pragmatics

While the DAML library contains some very large ontologies, most are small, with almost half containing only 1-20 terms (Table 3). Many represent knowledge about very narrow domains or capture small parts of larger domains. As per [SH01], agents will have to use these ontologies as a collective, rather than independent, sources of domain knowledge. ISRA uses information on classes and subclasses. As shown in Table 3, over 80 percent of the ontologies provide such information. The DAML library is also relevant for applications that require information on properties of classes. The ontologies are less relevant for applications that require knowledge about cardinality or 'sets'.

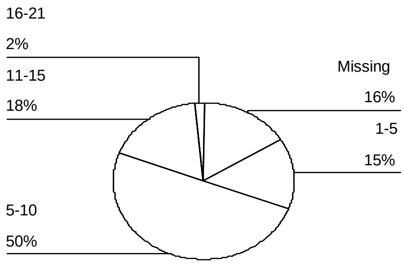
Syntax

Lawfulness



Values are the total number of syntax errors per ontology.

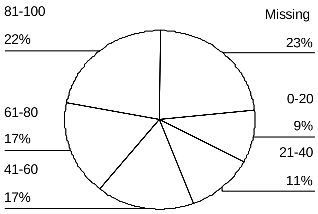
Richness



Values are the number of types of syntax used (max = 67).

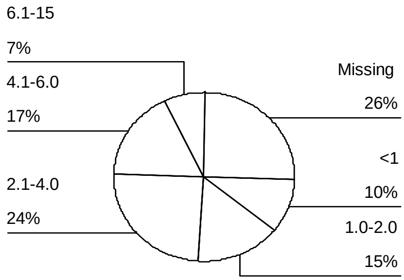
Semantics

Interpretability



Values are the % of words (used in class/property names) that exist in WordNet.

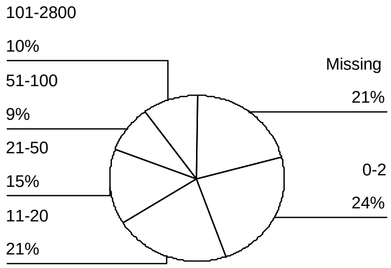
Clarity



Values are the average number of word senses for each ontology.

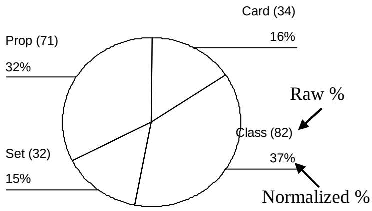
Pragmatics

Comprehensibility



Values are the sum of the classes and properties in each ontology.

Relevance



Values are the % of ontologies that provide these semantics. Raw % = % of ontologies, normalized % = ratio out of 100%.

Table 3: Results from Evaluation of DAML Library

5.4 Total Quality

Overall, the DAML ontologies:

- contain limited syntax, with about one syntax error per element of knowledge,
- contain semantics of varying precision, only half of which are common in English,
- are generally small, and
- primarily describe classes, subclasses, and properties.

Table 4 presents the analysis of overall quality. Although the results for some metrics still need to be calculated, and only equal weights have been used at this stage, the current results illustrate the wide variation in the quality of ontologies.

Metric	Dimension	Description	Low	Mean	High
Syntax	Lawfulness	% of correct syntax per class and prop	0.00	0.82	1.00
	Richness	% of available syntax used	0.04	0.17	0.41
	Total	*.5*L + .5*R*	*0.02*	*0.50*	*0.71*
Semantics	Interpretability	% of words used that exist in WordNet	0.00	0.63	1.00
	Clarity*	Average precision of words in ontology	0.07	0.78	1.00
	Total	*.5*I + .5*C*	*0.04*	*0.71*	*1.00*
Pragmatics	Comprehensibility	Size as % of the largest (capped at 500)	0.00	0.11	1.00
	Relevance	% providing subclass information	0.00	0.82	1.00
	Total	*.5*C + .5*R*	*0.00*	*0.47*	*1.00*
Total		***.5*Sy + .5*Se + .5*P***	***0.02***	***0.56***	***0.90***

* only includes words with one or more sense (0.0 represents extreme ambiguity, 1.0 represents no ambiguity).

Table 4: Total Quality

5.5 Implications of the Metrics Suite for Ontology Design

The metric suite could help ontology engineers when designing their ontologies to

- capture a more complete representation of their domain
- check the syntax of their ontologies
- ensure that the semantics they use are meaningful and precise
- develop an ontology so that it is relevant for many users/agents

The metric suite also has implications for developers of ontology languages. The DAML ontologies were developed specifically for the Semantic Web. Most DAML ontologies provide classes, subclasses, and properties in a domain. More detailed information is rarely provided. Information on cardinality, for example, is only found in approximately one third of the ontologies. More work is needed to determine why ontology designers are not using more advanced features. These features may not be necessary or developers may not understand them.

There is also a need for alternative evaluation methodologies. The methodology operates primarily at a domain or generic level. The evaluation could be extended by including a higher-level (formal) ontological evaluation to identify ontological inconsistencies [GW02]. Alternatively, the evaluation could be extended to include a more application-specific evaluation of the quality of *elements* of knowledge in an ontology for use in a *specific* task. Learning mechanisms could be used to update evaluations based on feedback from agents. More work is also needed on the weighting scheme. At present the auditor uses a simple additive weighting scheme. More testing will enable weights to be empirically derived. Finally, empirical tests are needed to verify the usefulness of the metrics by incorporating the assessment scheme into ISRA and to ensure that the proposed evaluation metrics are not merely rooted to specific ontologies, languages, or libraries.

6. Conclusion

A metric suite for ontology auditing and a prototype auditor have been developed to evaluate the effectiveness of ontologies for the Semantic Web. This appears to be one of the first attempts at comprehensive ontology evaluation. The metrics can assist practice by suggesting ontology design principles. The prototype auditor was used to assess the DAML ontology library. The results supported the usability of the metrics suite, found significant areas where developers need to improve the quality of their ontologies, and highlighted the need for more research on ontology evaluation.

References

[BHL01] Berners-Lee, T., Hendler, J., and Lassila, O. "The Semantic Web," Scientific American, May 2001, pp. 1-19.

[Bu77] Bunge, M. Treatise on Basic Philosophy: Volume 3: Ontology 1: The Furniture of the World, Reidel, Boston, 1977.

[BPS02] Burton-Jones, A., Purao, S., and Storey, V.C. "Context-Aware Query Processing on the Semantic Web," Proceedings of the 23rd International Conference on Information Systems, Barcelona, Spain, Dec. 16-19, 2002.

[Ca02] CACM "Special Issue on Ontology," Communications of the ACM (45:2), February, 2002, pp. 39-65.

[CK94] Chidamber, S.R., and Kemerer, C.F. "A Metrics Suite for Object-Oriented Design," IEEE Transactions on Software Engineering (20:6), 1994, pp. 476-493.

[CG00] Corcho, O., and Gomez Perez, A. "Evaluating Knowledge Representation and Reasoning Capabilities of Ontology Specification Languages," Proc. of the Workshop on Applications of Ontologies and Problem-Solving Methods, Berlin, 2000, pp. 1-9.

[Fe98] Fellbaum, C. (ed.). WordNet: An Electronic Lexical Database, MIT Press, Cambridge, MA, 1998.

[Fe01] Fensel, D., Harmelen, F.v., Horrocks, I., McGuinness, D.L., and Patel-Schneider, P.F. "OIL: An Ontology Infrastructure for the Semantic Web," IEEE Intelligent Systems (March/April), 2001, pp. 38-45.

[GS01] Gelbukh, A., and Sidorov, G. "Algorithm of Word Sense Disambiguation in an Explanatory Dictionary," Proceedings of the Workshop on Computational Lexicography, Great Britain, 2001, pp. 35-40.

[GS02] Gelbukh, A., and Sidorov, G. "Automatic Selection of Defining Vocabulary in an Explanatory Dictionary," In Computational Linguistics and Intelligent Text Processing, Lecture Notes in Computer Science N 2276, Springer-Verlag, 2002, pp. 300-303.

[Gr01] Greenberg, J. "Automatic Query Expansion via Lexical-Semantic Relationships," Journal of the American Society for Information Science (52:5), 2001, pp. 402-415.

[Gr93] Gruber, T.R. "A Translation Approach to Portable Ontology Specifications," Knowledge Acquisition (5), 1993, pp. 199-220.

[GW02] Guarino, N., and Welty, C. "Evaluating Ontological Decisions with OntoClean," Communications of the ACM (45:2), 2002, pp. 61-65.

[GL94] Guha, R.V., and Lenat, D.B. "Enabling Agents to Work Together," Communications of the ACM (37:7), 1994, pp. 127-142.

[HH01} Heflin, J. and Hendler, J. A Portrait of the Semantic Web in Action. IEEE Intelligent Systems, 16(2):54-59, 2001.

[He01] Hendler, J. "Agents and the Semantic Web," IEEE Intelligent Systems (March/April), 2001, pp. 30-36.

[Ie01] IEEE "Special issue on the Semantic Web," IEEE Intelligent Systems, (March/April) 2001, pp. 30-79.

[LSG03] Ledo-Mezquita, Y., Sidorov, G., and Gelbukh, A. "Tool for Computer-Aided Spanish Word Sense Disambiguation," In Computational Linguistics and Intelligent Text Processing, LNCS, N 2588, Springer-Verlag, 2003, pp. 279-282.

[MS01] Maedche, A., and Staab, S. "Ontology Learning for the Semantic Web," IEEE Intelligent Systems (March/April), 2001, pp. 72-79.

[Mi96] Miller, G.A. "Contextuality," In Mental Models in Cognitive Science, J. Oakhill and A. Garnham (eds.), Psychology Press, East Sussex, UK, 1996, pp. 1-18.

[Mi90] Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K.J. "Introduction to WordNet: An On-line Lexical Database," International Journal of Lexicography (3:4), 1990, pp. 235-244.

[Sm03] Smith, B. "Ontology and Information Systems," in Edward N. Zalta (ed.) Stanford Encyclopedia of Philosophy Stanford Encyclopedia of Philosophy, 2002, URL: http://ontology.buffalo.edu/ontology(PIC).pdf (Accessed March 28, 2003).

[St00] Stamper, R., Liu, K., Hafkamp, M., and Ades, Y. "Understanding the Role of Signs and Norms in Organizations--a Semiotic Approach to Information Systems Design," Behaviour & Information Technology (19:1), 2000, pp. 15-27.

[SH01] Stephens, L.M., and Huhns, M.N. "Consensus Ontologies: Reconciling the Semantics of Web Pages and Agents," IEEE Internet Computing (Sept-Oct), 2001, pp. 92-95.

[SBS02] Sugumaran, V., Burton-Jones, A., and Storey, V.C. "A Multi-Agent Prototype for Intelligent Query Processing on the Semantic Web," in Basu, A., and Dutta, S. (eds.) "Proceedings of the 12th Annual Workshop on Information Technology and Systems (WITS), Barcelona, Spain, Dec. 14-15, 2002.

[Vo94] Voorhees, E.M. "Query Expansion Using Lexical-Semantic Relations," Proceedings of the 17th Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland, 1994, pp. 61-69.

[WW95] Wand, Y., and Weber, R. "On the Deep Structure of Information Systems," Journal of Information Systems (5), 1995, pp. 203-223.

[We02] Weber, R. "Ontological Issues in Accounting Information Systems," In Researching Accounting as an Information Systems Discipline, S. Sutton and V. Arnold (eds.), American Accounting Association, Sarasota, FL, 2002.