Constructing Consensus Ontologies for the Semantic Web: A Conceptual Approach

LARRY M. STEPHENS, AUROVINDA K. GANGAM, and
MICHAEL N. HUHNS {stephens,gangam,huhns}@engr.sc.edu
Department of Computer Science and Engineering, University of South Carolina, Columbia, SC, 29208, USA

Abstract. Organizational knowledge typically comes from numerous independent sources, each with its own semantics. This paper describes a methodology by which information from large numbers of such sources can be associated, organized, and merged. The hypothesis is that a multiplicity of ontology fragments, representing the semantics of the independent sources, can be related to each other automatically *without* the use of a global ontology. That is, any pair of ontologies can be related indirectly through a *semantic bridge* consisting of many other previously unrelated ontologies, even when there is no way to determine a direct relationship between them. The relationships among the ontology fragments indicate the relationships among the sources, enabling the source information to be categorized and organized. An evaluation of the methodology has been conducted by relating numerous small, independently developed ontologies for several domains. A nice feature of the methodology is that common parts of the ontologies reinforce each other, while unique parts are deemphasized. The result is a *consensus* ontology.

Keywords: Merging ontologies; Semantic bridging; Reconciling ontologies

1. Introduction

Corporate information searches can involve data and documents both internal and external to the organization. The research reported herein targets the following basic problem: a search will typically uncover a large number of independently developed information sources—some relevant and some irrelevant; the sources might be ranked, but they are otherwise unorganized, and there are too many for a user to investigate manually. The problem is familiar and many solutions have been proposed, ranging from requiring the user to be more precise in specifying search criteria, to constructing more intelligent search engines, or to requiring sources to be more precise in describing their contents. A common theme for all of the approaches is the creation, use, and manipulation of ontologies for describing both requirements and sources [4, 8, 12, 14, 23, 20, 27, 30, 34, 36, 37, 40].

Unfortunately, ontologies are not a panacea unless everyone adheres to the same one, and no one has yet constructed an ontology that is comprehensive enough (in spite of determined attempts to create one [1, 21, 33], such as the Cyc Project [22], underway since 1984). Moreover, even if one did exist, it probably would not be adhered to, considering the dynamic and eclectic nature of the Web and other information sources.

There are three approaches for relating information from large numbers of independently managed sites: (1) all sites will use the same terminology with agreed-upon semantics (improbable), (2) each site will use its own terminology, but provide translations to a global ontology [9, 17] (difficult, and thus unlikely), and (3) each site will have a small, local ontology that will be related to those from other sites (described herein). We hypothesize that the small ontologies can be related to each other automatically *without* the use of a global ontology. That is, any pair of ontologies can be related indirectly through a *semantic bridge* consisting of many other previously unrelated ontologies, even when there is no way to determine a direct relationship between them. Our methodology relies on sites that have been annotated with ontologies [28]; such annotation is consistent with several visions for the Semantic Web [5, 6, 18]. The domains of the sites must be similar—else there would be no interesting relationships among them—but they will undoubtedly have dissimilar ontologies, because they will have been annotated independently.

Some researchers have attempted to merge a pair of ontologies in isolation, or merge a domain-specific ontology into a global, more general ontology [38]. Others have used this merging approach as a means for constructing a large global ontology [10]. Recently progress has been made in determining semantic similarity among separately developed entity classes [29] and refining the semantics of concepts in a thesaurus [16]. However, to our knowledge, no one has previously tried to reconcile a large number of closely related, domain-specific ontologies. We have evaluated our methodology by applying it to a large number of independently constructed ontologies; our preliminary results were first reported in [32]

2. The Information Interchange Problem

People need information in order to make decisions. The information can come from a variety of sources, and is assembled dynamically and uniquely for each specific problem. Also, the information must be fused and presented coherently, with inconsistencies identified and possibly resolved. Standards and conventions, such as XML, can help in making sources syntactically consistent. However, the information will likely be represented and conceptualized differently at each source.

Figure 1. In response to a user's request, agents assemble information from Web sites and other sources, each having its semantics defined by a different ontology, possibly represented in a formalism such as the OWL Web Ontology Language [11]

To address this, efforts are underway to standardize DTDs and XMLSchemas for various domains, based on ontologies. People developing Web pages will refer to these ontologies and use the concepts in the XML tags on their pages. Users will request groups of agents to assemble information from many sources, including Web pages. Agents will access the Web pages and be able to understand their content[1], because the ontologies provide the semantics for the tags used on the pages (see Figure 1).

A problem of semantic reconciliation will arise in two ways: (1) an individual source might refer to more than one ontology, and (2) Web pages retrieved from different sources will likely be based on different ontologies. In both cases, agents trying to form a coherent picture must be able to relate concepts that are explained in terms of different ontologies [35]. This requires that the ontologies be either merged or related.

In relating a given Web page to an ontology, we have found that the Web page typically has many concepts that are not included in the ontology. The concepts could be ignored, with the result that agents will not be able to access and use them, the ontology could be augmented to include them, or other ontologies could be found that already include them. The first approach loses semantics, the second requires the ability or permission to extend an existing ontology, and the third, generally the best, still requires relationships to be established among the several ontologies chosen to represent the Web page.

For example, a user, interested in a comparison of the conductivity of aluminum versus copper wire, might initiate a simple search on the term "conductor." A standard search engine could return a ranked list of 1,980,000 Web pages, as Google^{TM} recently did, some of which would concern orchestra and railroad conductors. The methodology we describe below would construct a merged ontology from the small ontologies associated with each of the first 100 or so pages. The merged ontology, centered on the term "conductor" and revealing the three mostly disjoint sub-ontologies for its three word senses, would be presented to the user, as shown in Figure 2. Based on this, the user could

[1] By *understand*, we mean that valid or reasonable inferences can be made about a concept, where the results of the inferencing are not explicitly represented a priori. For example, if a Web page describing a Taurus specifies it as an instance of a Sedan in a MotorVehicle ontology, and if this ontology describes a Sedan as a kind of PassengerVehicle, then it can be inferred that a Taurus is a PassengerVehicle.

Figure 2. A merged ontology refines the domain concepts needed by users to satisfy their requests

Figure 3. Ontologies can be made to relate to each other like pieces of a jigsaw puzzle

select a node to retrieve a page, or iterate by selecting a node from which to initiate a refined search.

3. Reconciling Independent Ontologies

In agent-assisted information retrieval, a user will describe a need to his agent, which will translate the description into a set of requests, using terms from the user's local ontology. The agent will contact on-line brokers and request their help in locating sources that can satisfy the requests. The agents must reconcile their semantics in order to communicate about the request. This will be seemingly impossible if their ontologies share no concepts. However, if their ontologies share concepts with a third ontology, then the third ontology might provide a semantic bridge to relate all three. Note that the agents do not have to relate their entire ontologies, only the portions needed to respond to the request.

The difficulty in establishing a bridge will depend on the semantic distance between the concepts, and on the number of ontologies that comprise the bridge. Our methodology is appropriate when there are large numbers of small ontologies—the situation we expect to occur in large and complex information environments. Our metaphor is that a small ontology is like a piece of a jigsaw puzzle, as depicted in Figure 3. It is difficult to relate two random pieces of a jigsaw puzzle until they are constrained by other puzzle pieces. We expect the same to be true for ontologies.

Two concepts can have the following seven mutually exclusive relationships between them: *subclass*, *superclass*, *equivalence*, *partOf*, *hasPart*, *sibling*, or *other*. If a request contains three concepts, for example, and the request must be related to an ontology containing 10 concepts, then there are $7 \times 3 \times 10 = 210$ possible relationships among them. Only 30 of the 210 are correct, because each of the three concepts in the request has exactly *one* correct relationship with each of the 10 concepts in the source's ontology.

The correct ones can be determined by applying constraints among the concepts within an ontology, and among multiple ontologies. Once the correct relationships have been determined, we make use of *equiva-*

lence and *sibling* or, where those do not exist, the most specific *superclass* or *partOf*.

In Figure 3, the ontology fragment on the left is represented as *partOf(Wheel, Truck)*, while the one on the right is represented as *partOf(Tire, APC)*. There are no obvious equivalences between these two fragments. The concept *Truck* in the first ontology could be related to *APC* in the second by *equivalence*, *partOf*, *hasPart*, *subclass*, *superclass*, or *other*. There is no way to decide which is correct. When the middle ontology fragment *partOf(Wheel, APC)* is added, there is evidence that the concepts *Truck* and *APC*, and *Wheel* and *Tire* could be *equivalent*.

This example exploits the existence of the relation *partOf*, which is common to all three ontologies. Other domain-independent relations, such as *subclassOf*, *instanceOf*, and *subrelationOf*[2], will be necessary for the reconciliation process. Moreover, the reflexivity, symmetry, asymmetry, transitivity, irreflexivity, and antisymmetry properties are needed for relating occurrences of the relations to each other [31]. Domain concepts and relations can be related to each other by converse/inverse, composition, (exhaustive) partition, part-whole with 6 subtypes [7, 39], and temporal attitude. There must be some minimum set of these fundamental relations that are understood and used by all local ontologies and information system components.

In attempting to relate two ontologies, a system might be unable to find correspondences between concepts because of insufficient constraints and similarity among their terms. However, trying to find correspondences with other ontologies might yield enough constraints to relate the original two ontologies. As more ontologies are related, there will be more constraints among the terms of any pair, which is an advantage. It is also a disadvantage in that some of the constraints might be in conflict. We make use of the preponderance of evidence to resolve these statistically.

4. Experimental Methodology

We conducted experiments in three domains. We asked one group of 54 graduate students in computer science to construct a small ontology for the Humans/People/Persons domain, a second group of 28 students to construct a small ontology for the Buildings domain, and a third group of 25 students to construct a small ontology for the Sports domain.

[2] Examples of subrelations are (1) *on* is a subrelation of *above* in spatial relations, (2) *daughterOf* is a subrelation of *childOf* in familial relations, and (3) *cityLocation* is a subrelation of *countryLocation* in geographic relations.

Figure 4. A typical small ontology used to characterize an information source about people (all links denote subclasses)

The ontologies were written in OWL [11] and were required to contain at least 8 classes with at least 4 levels of subclasses; a sample ontology is shown in Figure 4. In this and all other figures the directed link is from *superclass* to *subclass.*

We merged the files in each of the three domains using all syntactic and semantic information available in the component ontologies. Our system merges the component files one-at-a-time into a resultant merged file. For each node in the resultant file, we maintain a *reinforcement* value, which indicates how many times the node has been matched as ontologies are merged. We also maintain reinforcement values for class-subclass links. The details of the methodology are presented in Section 5 and the results in Section 6. The methodology includes

- string matching for node names, including plural-pairs.
- checking for antonyms and synonyms,
- making the algorithm commutative,
- removing circularities in the merged ontologies,
- incorporating disjoint-class definitions, and
- identifying noun “classifiers,” such as “Apartment” in “ApartmentBuilding” to determine subclass relationships.

5. Ontology-Merging Techniques

Our objective is to make use of all of the syntactic or semantic information that is available to achieve the best possible merger of the component ontologies. The syntactic information available is the names of the nodes, for which we employ various string-matching techniques. The semantic information includes the semantics associated with a subclass link in the ontologies, prefixes that indicate antonyms, and evolving synonym sets.

5.1. Substring Matching

Our principle technique for merging two ontologies relies on simple string and substring matching. The name of a node from one ontology is systematically compared to each of the nodes from another ontology using the following prioritized rules:

- If an exact match is found, then the comparisons cease and a value of 1.0 is assigned as a match.

- If the node names are antonyms of each other, then the merging attempt is aborted. We detect antonyms formed by prefixes such as *anti*, *dis*, *im*, *in*, *non*, and *un*. In general, antonym checking prevents some mergers and produces a correspondingly larger number of total classes compared to uninformed string matching. Antonyms are a convenient way to subdivide concepts or domains into subconcepts and opposites, and were widely used in the student-produced ontologies. For example, it is typical that "People" might be divided into "Students" and "NonStudents," or "Citizens" and "NonCitizens."

- If the names are not identical, then we check for plural pairs that follow the traditional rules of grammar such as building–buildings, calf–calves, knife–knives, and thesis–theses. The match value is set to 1.0 as if the node names were identical.

- If the shorter string is wholly contained at the *end* of the longer string, then the nodes are not merged but the node with the shorter string name is asserted to be a super class of the node having the longer name. The use of *noun classifers* is discussed further in Section 5.7. For example, the string "Animal" matches the end of the string "WildAnimal," so "Animal"is assumed to a superclass of "WildAnimal."

- Otherwise, the match value is based on the extent to which the *leading* substring of the shorter name matches the *leading* substring of the longer name. For example, the first five characters of "Animal" and "Animate" are identical, and a match value of 5/7 = 0.71 is assigned.

The best match for each node is found, and if its match value exceeds a threshold value (set at 0.50 for our experiments), a successful match is declared. The merging technique works *inter* ontology, not *intra* ontology. For example, in the final ontology for the "People" domain, there are distinct classes for "Animal" and "Animate"—strings that would be expected to merge—because one of the component ontologies made a distinction between the two concepts.

Figure 5. The best substring match common to both *Student* and *Rodentia* is *dent*, an undesirable match.

Figure 6. Many-to-one matching examples in which the ontology on left is merged into the ontology on right. The result depends on the order of merging–an undesirable effect. The numbers in parentheses indicate the reinforcement of a node after merging

We investigated using "dot-plot" matching [24]; however, dot-plots matches were not restrictive enough to give good results for cases involving the interior portions of the strings. For example, "Student" partially matches with "Rodentia" (Figure 5).

5.2. Merged-Name List and Synonyms

For each node, we maintain a list of all names that nave been merged in reinforcing the concept. Originally this was done to allow a progression of name variations to merge; however, now we use it to confirm that node merges make sense. This "synset" approach disclosed that the dot-plot string matching was too inexact, permitting, for example, the successive "wandering" matches of Ag*ent*, Stud*ent*, *Ent*ity, and Rod*enti*a.

We also initialized synsets for some of the nodes with synonyms obtained from WordNet [26]. The use of synonyms increases the number of nodes that are merged and increases node reinforcement values. For example, from WordNet we include the synonyms "Person" and "Human," which would not have been found using string-matching techniques.

5.3. Many-to-One Node Merging

Our original algorithm sought a best match for each node on the left with at most one from the right; however, a node in the right-hand ontology might match many nodes from the left. In Figure 6, in which the ontology on the left is merged into the ontology on the right, the resultant ontology depends on the order of merging. In the top portion of the figure, the discriminating semantics between "Animals" and "Animate" is lost.

5.4. One-to-One Node Merging

In one-to-one node merging, the nodes in one ontology merge with the best matched node from the other, regardless of the order of merging. For example, if two nodes from the left-hand ontology match with

Figure 7. One-to-one merging prevents several distinct nodes from collapsing onto a single node in the resultant ontology. For this merging order, the many-to-one technique would have produced an ontology with the leaf node "Animal" refinforced by 3.

Figure 8. Cycle removal eliminates the link between "Person" and "Animal." The weakly reinforced link between "Thing" and "Person" might be removed as shown in Figure 9.

a single node in the right-hand ontology, then only the best pair is matched; the other node is not merged, as shown in Figure 7. Although the nodes "Animals" and "Animate" both match well with "Animal," only "Animals" is merged with "Animal." If the order of merging is reversed (as in the bottom portion of Figure 6), then the number of nodes remains the same; however, the name of a merged node might be different. To avoid differences in names of merged nodes, we maintain a list of all names that have been merged for a given node, as discussed in Section 5.2.

With one-to-one merging, the amount of merging is less than for the many-to-one algorithm—an effect that generally produces a slightly larger merged ontology than the many-to-one technique; however, the merging is now commutative.

5.5. Cycle Removal

Cycles can appear in the final merged ontology if, via a direct subclass link or transitive closure, node A is both a *subclassOf* and *superClassOf* node B. Several of our student-produced ontologies had conflicting information that created cycles. When cycles are detected, they must be removed, either automatically or by seeking user intervention. Our algorithm removes the weakest reinforced link in a cycle and ensures that there is at least one path from all nodes to the root node. See Figure 8.

5.6. Transitive-Closure Link Removal

We considered removing from our merged ontologies all transitive closure class-subclass links, and reinforcing the remaining links. For example, if A has subclass B and B has subclass C, then it appears needless to assert explicitly that A has subclass C. Figure 9 shows the result of removing the weakly reinforced, redundant link from "Thing" to "Person" in Figure 8 and reinforcing the other links.

However, this approach can introduce results that clearly violate a "consensus" in a merged ontology. Suppose the links from A to B

Figure 9. The direct link between "Thing" and "Person" in the merged ontology of Figure 8 is removed, and the remaining links are reinforced.

Figure 10. The consensus is that the concept "Women" is more strongly linked to "Humans" than "Female." Removing the direct link from "Humans" to "Women" and reinforcing remaining links violates that consensus. Node and link reinforcements are shown in parentheses.

and B to C have reinforcement values that are much less than the reinforcement of the direct link from A to C. Removing the direct link and reinforcing the remaining links gives the impression that the consensus supports the weakly reinforced links. Our conclusion was to abandon the procedure and leave link reinforcement values unchanged. The relationships among "Humans," "Female," and "Women" in Figure 10, which are taken from our results, illustrate how using transitive closure can violate a clear consensus.

5.7. Noun Classifiers

Nouns are sometimes used as adjectives to restrict the meaning of other nouns. For example, the term "office building" suggests a special type of "building," and it is reasonable to infer that "office building" is a subclass of "building," even if the link is not explicit in an ontology. We have not fully automated this process, and use a hand-generated list of nouns that could be the target of classifiers in the domain of the ontologies. In addition, we use the heuristic of matching the shorter string name (the superclass) to the end of the longer string (the subclass) as noted in Section 5.1.

The identification of noun-noun pairs is not straightforward if there is no space or hyphen separating the nouns. WordNet does not recognize the string "OfficeBuilding" but easily finds the meaning of "office building" or "office-building." Ontology builders need a set of conventions for entering knowledge. We prefer the use of "camel-case," which allows words to be easily extracted. Without conventions for representing compound words, extraction becomes very difficult. From "warmblooded-animal," one might extract "war," "warm," "arm," "blood," "loo," "ode," "animal," "ma," and "mal" to name a few.

6. Discussion of Results

In the Humans/People/Persons ontology domain the component ontologies described 864 classes, while the merged ontology contained

Figure 11. A portion of the ontology formed by merging 54 independently constructed ontologies for the domain Humans/People/Persons. The entire ontology has 389 concepts related by 696 subclass links.

Figure 12. The consensus ontology for the "Humans" domain formed by merging concepts with common subclasses and superclasses from 54 component ontologies. The resultant ontology contains 20 concepts related by 25 subclass links.

389 classes in a single graph with a root node of the OWL concept `owl:Thing`. All of the concepts were related, i.e., there was some relationship (path) between any pair of the merged concepts (see Figure 11).

Next, we constructed a *consensus ontology* by counting the number of times classes and subclass links appeared in the component ontologies when we performed the merging operation. For example, the class "Human" and its matching classes appeared 53 times (one of the 54 students used the term "Sapiens(Man)," which failed to match the other nodes). The subclass link from Mammals (and its matches) to Humans (and its matches) appeared 10 times. We termed these numbers the "reinforcement" of concepts and links. We then removed from the merged ontology any classes or links that were not reinforced above a threshold level.

Finally, we applied an *equivalence heuristic* for collapsing classes that have common reinforced superclasses and subclasses: if all reinforced subclasses of X are also reinforced subclasses of Y, and all reinforced superclasses of X are also reinforced superclasses of Y, then *equivalence* holds between X and Y. This heuristic is similar to an inexact graph matching technique such as [25].

Figure 12 shows the collapsed consensus ontology for the domain of "Humans," now containing 20 classes related by 25 subclass links. All nodes are reinforced at least 5 times and all links, except as noted, reinforced at least 3 times. The weakly reinforced links "Female–Women" and "Male–Men" could be omitted but are included to illustrate the transitive closure trade-off.

Figures 13 and 14 show the results for the domains of "Buildings" and "Sports," which are based on 28 and 25 component ontologies, respectively. For these two domains, the reinforcement threshold for concepts and links is 3.

A consensus ontology is perhaps the most useful organization for information retrieval by humans, because it represents the way most people view the world and its information. For example, if most people wrongly believe that crocodiles are a kind of mammal, then most people

Figure 13. The consensus ontology for the "Building" domain contains 23 concepts and 26 links. "Office" is considered both "NonResidential" and "Commercial." The concepts "Plant" (a subclass of "LivingThing") and "Factory" (a subclass of NonLivingThing) appear in different branches of the ontology. The merged ontology is derived from 28 component ontologies.

Figure 14. The final consensus ontology for the "Sports" with 18 concepts and 20 links. "Soccer" is classified slightly more strongly as a subclass of "Sports" rather than "OutdoorSports."

would find it easier to locate information about crocodiles if it were placed in a mammals grouping, rather than where it factually belonged.

The information retrieval measures of precision and recall are based on some degree of match between a request and a response. The length of a semantic bridge between two concepts can provide an alternative measure of conceptual distance and an improved notion for relevance of information [2, 3, 15]. Previous measures relied on the number of properties shared by two concepts within the same ontology, or the number of links separating two concepts within the same ontology [13]. These measures not only require a common ontology, but also do not take into account the density or paucity of information about a concept. Our measure does not require a common ontology and is sensitive to the information available.

Although promising, our experiments and analysis so far are preliminary and ongoing. We used the following simplifications:

- We did not make use of properties of the classes, as would a complete implementation of subsumption.
- Our string-matching algorithm did not use a thorough morphological analysis to separate the root word from its prefixes and suffixes. We do, however, handle singular and plural noun forms in most cases, and discriminate between antonym pairs.
- We used only subclass-superclass information, and have not yet made use of other important relationships, notably *partOf* as suggested in Figure 3.

We are addressing some of these limitations in our continuing research. Moreover, our hypothesis, that a multiplicity of ontology fragments can be related automatically without the use of a global ontology, appears correct, but our investigation is continuing according to the following plan:

- We are improving the algorithm for relating ontologies, based on methods for partial and inexact matching, making extensive use of common ontological primitives, such as *subclass* and *partOf*. The algorithm will take as input ontology fragments and produce mappings among the concepts represented in the fragments. It will use constraints among known ontological primitives to control computational complexity. Some of these details are presented in Appendix A.

- We are developing metrics for success in relating ontologies, based on the number of concepts correctly related, as well as the number incorrectly matched. The *quality* of a match will be based on semantic distance, as measured by the number of intervening semantic bridges.

7. Conclusion

Imagine that in response to a request for information about a particular topic, a user receives pointers to more than 1000 documents, which might or might not be relevant. The technology developed by our research would yield an organization of the received information, with the semantics of each document reconciled. This is a key enabling technology for knowledge-management systems. The technique could be applied off-line by search engines such as Google^{TM}, thereby providing ontologies that do not exist today for refining queries.

Our premise is that it is easier to develop small ontologies, whether or not a global one is available, and that these can be automatically and *ex post facto* related. We are determining the efficacy of local annotation for Web sources, as well as the ability to perform reconciliation qualified by measures of semantic distance. The results of our effort will be (1) software components for semantic reconciliation, and (2) a scientific understanding of automated semantic reconciliation among disparate information sources.

References

1. E. Agirre, O. Ansa, E. Hovy, and D. Martnez, “Enriching very large ontologies using the WWW,” in *Proceedings of the Ontology Learning Workshop*, ECAI, Berlin, Germany, July 2000.
2. E. Agirre and G. Rigau, “A proposal for Word Sense Disambiguation using Conceptual Distance,” in *Proceedings of Recent Advances in NLP (RANLP95)*, Tzigov Chark, Bulgaria, 258–264, 1995.

3. E. Agirre, X. Arregi, X. Artola, A. Daz de Ilarraza, and K. Sarasola, “A methodology for the extraction of semantic knowledge from dictionaries using phrasal patterns,” in *Proceedings of IBERAMIA’94. IV Congreso Iberoamericano de Inteligencia Artificial*, McGraw-Hill, 263–270, 1994.
4. J. L. Ambite, Y. Arens, E. Hovy, A. Philpot, L. Gravano, V. Hatzivassilogluo, and J. Klavens, “Simplifying Data Access: The Energy Data Collection Project,” *IEEE Computer*, 34(2), 47–54, 2001.
5. T. Berners-Lee *Weaving the Web*, Harper, San Francisco, CA., 1999.
6. T. Berners-Lee, J. Hendler, and O. Lassila, “The Semantic Web,” *Scientific American*, 284(5), 34–43, 2001.
7. R. Chaffin and D. Herrmann, ”The nature of semantic relations: a comparison of two approaches,” in *Relational Models of the Lexicon: Representing knowledge in semantic networks*, M. W. Evens, ed., Cambridge University Press, Cambridge, England, 289–334, 1988.
8. B. Chandrasekaran, J. R. Josephson, and V. R. Benjamins, “What are ontologies, and why do we need them?,” *IEEE Intelligent Systems*, 14(1), 20–26, 1999.
9. M. Ciocoiu and D. S. Nau, “Ontology-Based Semantics,” in *Proc. 7th International Conference on Principles of Knowledge Representation and Reasoning*, Breckenridge, CO, April 2000.
10. H. Dalianis and E. Hovy, “Integrating STEP Schemata using Automatic Methods,” in *Proceedings of the ECAI-98 Workshop on Applications of Ontologies and Problem-Solving Methods*, Brighton, England, August 24–25, 54–66, 1998.
11. M. Dean and G. Schreiber (eds.), “OWL Web Ontology Language 1.0 Reference,” March 31, 2003, http://www.w3.org/TR/owl-ref/.
12. S. Decker, M. Erdmann, D. Fensel, and R. Studer, “Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information,” in R. Meersman et al. (eds.), *Semantic Issues in Multimedia Systems, Proceedings of DS-8*, Kluwer Academic Publishers, Boston, 351–369, 1999.
13. H. S. Delugach, “An Exploration Into Semantic Distance,” in *Lecture Notes in Artificial Intelligence*, 754, Ch. 9, Springer-Verlag, Berlin, 1993.
14. A. Farquhar, R. Fikes, and J. Rice, “Tools for Assembling Modular Ontologies in Ontolingua,” in *Proceedings of AAAI-97*, AAAI Press, Menlo Park, CA, 436–441, 1997.
15. N. Foo, B. J. Garner, A. Rao, and E Tsui, “Semantic Distance in Conceptual Graphs,” in *Current Directions in Conceptual Structure Research*, Gerhotz L (ed.), Ellis Horwood, 149–154, 1992.
16. J. Geller, H. Gu, Y. Perl, and M. Halper, “Semantic refinement and error correction in large terminological knowledge bases,” *Data and Knowledge Engineering*, 45, 1–32, 2003.
17. T. R. Gruber, “A Translation Approach to Portable Ontology Specifications,” Knowledge Systems Laboratory Technical Report KSL 92-71, Stanford University, April 1993.
18. J. Heflin and J. Hendler, “Dynamic Ontologies on the Web,” in *Proc. 17th National Conference on AI (AAAI-2000)*, AAAI Press, Menlo Park, CA, 443–449, July 2000.
19. M. N. Huhns and L. M. Stephens, “Plausible Inferencing Using Extended Composition,” *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, IJCAI-89*, Detroit, MI, 1420–1425, August 1989.
20. V. Kashyap and A. Sheth, *Information Brokering across Heterogeneous Digital Data: A Metadata-based Approach*, Kluwer Academic Publishers, Boston, 2000.

21. K. Knight and S. Luk, "Building a Large-Scale Knowledge Base for Machine Translation," *Proc. of the National Conference on Artificial Intelligence (AAAI)*, Seattle, WA, 773–778, 1994.
22. D. B. Lenat and R. V. Guha, *Building Large Knowledge-Based Systems*, Addison-Wesley, Reading, MA, 1990.
23. K. Mahalingam and M. N. Huhns, "An Ontology Tool for Distributed Information Environments," *IEEE Computer*, 30(6), 80–83, 1997.
24. J. V. Maizel, Jr. and R. P. Lenk, "Enhanced graphic matrix analysis of nucleic acid and protein sequences," *Proceeding of the National Academy of Sciences*, 78(12), 7665-9, 1981.
25. N. Manocha, D. Cook, and L. Holder, "Structural Web Search Using a Graph-Based Discovery System," *ACM Intelligence*, 12(1), 20–29, 2001.
26. G. A. Miller, "WordNet: A Lexical Database for English," *Communications of the ACM*, 1995 38(11), 39–41, 1995, http://www.cogsci.princeton.edu/˜wn/
27. M. Nodine, W. Bohrer, and A. Hee Hiong Ngu, "Semantic Brokering over Dynamic Heterogeneous Data Sources in InfoSleuth," *15th International Conference on Data Engineering*, Sydney, Australia, March 1999.
28. J. M. Pierre, "Practical Issues for Automated Categorization of Web Sites," *Electronic Proc. ECDL 2000 Workshop on the Semantic Web*, Lisbon, Portugal, 2000, http://www.ics.forth.gr/proj/isst/SemWeb/program.html
29. M. A. Rodríguez and M. J. Egenhofer, "Determining Semantic Similarity among Entity Classes from Different Ontologies," *IEEE Transactions on Knowledge and Data Engineering*, 15(2), 442–456, 2003.
30. A. P. Sheth and J. A. Larson, "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases," *ACM Computing Surveys*, 22(3), 183–236, 1990.
31. L. M. Stephens and Y. F. Chen, "Principles for Organizing Semantic Relations in Large Knowledge Bases," *IEEE Transactions on Knowledge and Data Engineering*, 8(3), 492–496, 1996.
32. L. M. Stephens and M. N. Huhns, "Consensus Ontologies: Reconciling the Semantics of Web Pages and Agents," *IEEE Internet Computing*, 5(5), 92–95, 2001.
33. K. Stoffel, M. Taylor, and J. Hendler, "Efficient Management of Very Large Ontologies," in *Proceedings of American Association for Artificial Intelligence Conference (AAAI-97)*, AAAI/MIT Press, 442–447, 1997.
34. W. Swartout and A. Tate, "Ontologies," *IEEE Intelligent Systems*, 14(1), 18–19, 1999.
35. S. Thacker, A. Sheth, and S. Patel, "Complex Relationships for the Semantic Web," in *Creating the Semantic Web*, D. Fensel, J. Hendler, H. Liebermann, and W. Wahlster (eds.), MIT Press, 2002.
36. M. Uschold and M. Gruninger, "Ontologies: Principles, Methods, and Applications," in *Knowledge Engineering Review*, 11, 96–137, 1996.
37. C. Welty, "The Ontological Nature of Subject Taxonomies," in N. Guarino (ed.), *Formal Ontology in Information Systems*, IOS Press, Amsterdam 317–327, 1998.
38. G. Wiederhold, "An Algebra for Ontology Composition," in *Proc. Monterey Workshop on Formal Methods*, U.S. Naval Postgraduate School, 56–61, 1994.
39. M. E. Winston, R. Chaffin, and D. Herrmann, "A Taxonomy of Part-Whole Relations," *Cognitive Science*, 11, 417–444, 1987.
40. K. T. Yao, I. Y. Ko, R. Eleish, and R. Neches, "Asynchronous Information Space Analysis Architecture Using Content and Structure Based Service Bro-

kering," in *Proceedings of the Fifth ACM Conference on Digital Libraries (DL 2000)*, San Antonio, Texas, June 2000.

Appendix A: Heuristics for Merging Component Ontologies

Using the relations of Section 3, our methodology is embodied in the following algorithm, similar to one used for plausible inferencing among Cyc relationships [19]:

Given ontologies A and B, both based on the OWL specification [11], having nodes $n_A(i), i = 1, 2, \ldots, N$ and $n_B(k), k = 1, 2, \ldots, M$ and relationship arcs $r_A(i1, i2)$ and $r_B(k1, k2)$,

- Perform string matching among $n_A(i)$ and $n_B(k), \forall i, k$, to determine candidate matches

- Perform synonym matching among $n_A(i)$ and $n_B(k), \forall i, k$, to determine additional candidate matches

- Discard matches where $n_A(i1)$ matches $n_B(k1)$ and $n_A(i2)$ matches $n_B(k2)$, but $r_A(i1, i2)$ is inconsistent with $r_B(k1, k2)$; matches that remain are presumed to represent the relation *equivalence*[3]

The algorithm can be made more intelligent with the use of the following steps:

- Add additional relations

  - **If** $n_A(i) \equiv n_B(k) \wedge$ $n_A(j)$ is a *subclass (superclass/hasPart/partOf)*[4] of $n_A(i)$ **then** $n_A(j)$ is a *subclass (superclass/hasPart/partOf)* of $n_B(k)$

  - **If** $n_A(i1) \subseteq n_A(i2) \subseteq n_A(i3) \wedge n_B(k1) \subseteq n_B(k2) \subseteq n_B(k3) \wedge n_A(i1) \equiv n_B(k1) \wedge n_A(i3) \equiv n_B(k3)$ **then** the relation between $n_A(i2)$ and $n_B(k2)$ is either *sibling, subclass, superclass*, or *equivalence*

  - **If** $n_A(i1)$ *partOf* $n_A(i2)$ *partOf* $n_A(i3) \wedge n_B(k1)$ *partOf* $n_B(k2)$ *partOf* $n_B(k3) \wedge n_A(i1) \equiv n_B(k1) \wedge n_A(i3) \equiv n_B(k3)$ **then** the relation between $n_A(i2)$ and $n_B(k2)$ is either *sibling, partOf, hasPart*, or *equivalence*[5]

[3] The relation *equivalence* is denoted by $\equiv$.

[4] The relation *subclass* is denoted by $\subseteq$.

[5] This relation cannot be *subclass* (*superclass*), because if $n_A(i2) \subseteq n_B(k2)$, then at least one of the equivalence relations $n_A(i1) \equiv n_B(k1)$ or $n_A(i3) \equiv n_B(k3)$ must instead be *subclass* (*superclass*).

– Considering an additional ontology C introduces constraints that enable relations to be added as follows:

- **If** $n_A(i1) \equiv n_C(j1) \wedge n_C(j2) \equiv n_B(k2) \wedge n_A(i1) \subseteq n_A(i2) \wedge n_B(k1) \subseteq n_B(k2) \wedge n_C(j1) \subseteq n_C(j2) \wedge$ there are no known relationships between $n_A(i1)$ and $n_B(k1)$ $\wedge$ there are no known relationships between $n_A(i2)$ and $n_B(k2)$
  then the relationship between $n_A(i1)$ and $n_B(k1)$ and $n_A(i2)$ and $n_B(k2)$ is either *sibling, subclass, superclass*, or *equivalence*
- **If** $n_A(i1) \equiv n_C(j1) \wedge n_C(j2) \equiv n_B(k2) \wedge n_A(i1)$ *partOf* $n_A(i2) \wedge n_C(j1)$ *partOf* $n_C(j2) \wedge$ there are no known relationships between $n_A(i1)$ and $n_B(k1) \wedge$ there are no known relationships between $n_A(i2)$ and $n_B(k2)$
  then the relationship between $n_A(i1)$ and $n_B(k1)$ and $n_A(i2)$ and $n_B(k2)$ cannot be *other*.

Appendix B: An Example Input Ontology in OWL

```
<?xml version="1.0" ?>
 <rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:xsd="http://www.w3.org/2000/10/XMLSchema#"
   xmlns:owl="http://www.w3.org/2002/07/owl#"
   xmlns:exd="http://www.w3.org/TR/@@/owl-ex-dt#"
   xmlns:dex="http://www.w3.org/TR/@@/owl-ex#"
   xmlns="http://www.w3.org/TR/@@/owl-ex#">
   <owl:Ontology rdf:about="">
     <owl:versionInfo>$Id: student1.rdf,v 1.1 2002/07/29
         15:33:03 huhns Exp $</owl:versionInfo>
     <rdfs:comment>People Domain Ontology</rdfs:comment>
     <owl:imports
        rdf:resource="http://www.w3.org/2002/07/owl" />
   </owl:Ontology>

   <owl:Class rdf:ID="Living Things">
     <rdfs:label>Living Things</rdfs:label>
   </owl:Class>

   <owl:Class rdf:ID="Animals">
```

```
  <rdfs:subClassOf rdf:resource="#Living Things"/>
</owl:Class>

<owl:ObjectProperty rdf:ID="LifeSpan">
    <rdfs:domain rdf:resource="#Animals"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="Plants">
  <rdfs:subClassOf rdf:resource="#Living Things"/>
</owl:Class>

<owl:Class rdf:ID="Humans">
  <rdfs:subClassOf rdf:resource="#Animals"/>
</owl:Class>

<owl:ObjectProperty rdf:ID="Age">
    <rdfs:domain rdf:resource="#Humans"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="Sex">
    <rdfs:domain rdf:resource="#Humans"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="religion">
    <rdfs:domain rdf:resource="#Humans"/>
</owl:ObjectProperty>


<owl:Class rdf:ID="Non-humans">
  <rdfs:subClassOf rdf:resource="#Animals"/>
</owl:Class>

<owl:Class rdf:ID="Asians">
  <rdfs:subClassOf rdf:resource="#Humans"/>
</owl:Class>

<owl:Class rdf:ID="NorthAmericans">
  <rdfs:subClassOf rdf:resource="#Humans"/>
</owl:Class>

<owl:Class rdf:ID="Others">
  <rdfs:subClassOf rdf:resource="#Humans"/>
  <rdfs:comment>
```

```
      People from all other continents
    </rdfs:comment>
  </owl:Class>

</rdf:RDF>
```

Acknowledgements

This work was supported in part by the Advanced Research and Development Activity (ARDA). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the U. S. Government. This work was also supported by the US National Science Foundation under grant number IIS-0083362 and by the University of South Carolina under a grant from the Office of Research.