Developing Consensus Ontologies for the Semantic Web

Larry M. Stephens, Aurovinda K. Gangam, and Michael N. Huhns
Department of Computer Science and Engineering
University of South Carolina, Columbia, SC, 29208, USA

Abstract

This paper describes a methodology for associating, organizing, and merging large numbers of independently developed information sources. 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. The methodology has been tested by merging small, independently developed ontologies for the domains of *Humans*, *Buildings*, and *Sports*. The methodology, which reinforces common parts of the component ontologies and deemphasizes unique parts, produces a *consensus* ontology.

1 Introduction

A search for information will typically uncover a large number of independently developed information sources—some relevant and some irrelevant. A common theme for refining searches is the creation, use, and manipulation of ontologies for describing both requirements and sources [2, 4, 6, 7, 9, 13, 16]. 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—even given ongoing attempts to create one such as [1, 10] and the Cyc Project [11], 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.

This paper describes a methodology for merging and, therefore, relating small, independently developed ontologies automatically *without* the use of a global ontology. It is assumed that the sites have been annotated with ontologolical information [14]—a representation consistent with several visions for the Semantic Web [3, 8]. 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.

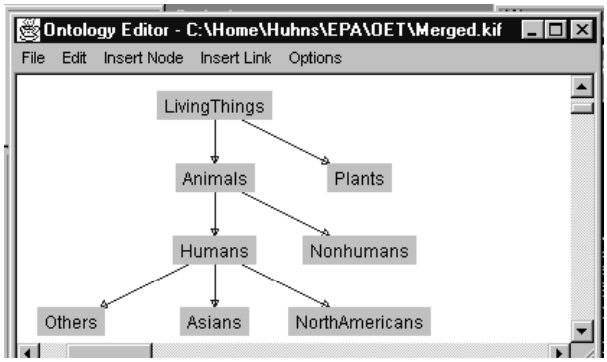

Figure 1: A typical small ontology used to characterize an information source about people (all links denote subclasses)

2 Experimental Methodology

To assess the methodology, we asked each student in a group of 54 computer science graduate students to construct a small ontology for the domain of *Humans-People-Persons.* A second group of 28 students constructed small ontologies for the *Buildings* domain, and a third group of 25 students developed ontologies for the *Sports* domain. The ontologies were written in OWL [5] and contained at least 8 classes organized with at least 4 levels of subclasses; a sample ontology is shown in Figure 1. In this and all other figures the directed link is from *superclass* to *subclass*.

We merge the files in each of the three domains using the syntactic and semantic information available in the component ontologies. The syntactic information is derived from the names of the nodes, for which we employ various string-matching techniques including detection of plural endings. The semantic information includes the meaning of the subclass link in the ontologies, prefixes that indicate antonyms, and evolving sets of synonyms for matching nodes. The synsets, which are used to track the progress of

merging and to monitor correctness, are seeded from WordNet [12]. The details of the node-merging algorithm are given in the Appendix.

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 is matched as ontologies are merged. We also maintain reinforcement values for class-subclass links. The original work reported in [15] was dependent on the ontology sequencing; the work reported herein uses an algorithm that is commutative with respect to the ordering of component ontologies.

The enhanced algorithm also identifies and removes circularities in the merged ontologies, enforces disjoint-class definitions that are specified in the component ontologies, and identifies noun "classifiers," such as *Apartment* in *Apartment-Building* to determine subclass relationships. For noun-classifier identification, we use the heuristic of matching the shorter node name (*Building*—the candidate superclass) with the ending of the longer string (*ApartmentBuilding*—the candidate subclass).

The identification of noun-noun pairs is not straightforward if there is no space, hyphen, or case change between the nouns. The string "OfficeBuilding" is not recognized by WordNet, which correctly identifies both "office building" and "office-building." Ontology builders need a set of conventions for entering noun-classifier knowledge. We prefer the use of "camel-case," which allows words to be easily extracted. Without such conventions, extraction becomes difficult. From "warmbloodedanimal," one might extract "war," "warm," "arm," "blood," "loo," "ode," "animal," "ma," and "mal" to name a few.

3 Results

In the *Humans-People-Persons* domain, the component ontologies described 864 classes, while the merged ontology shown in Figure 2 contained 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.

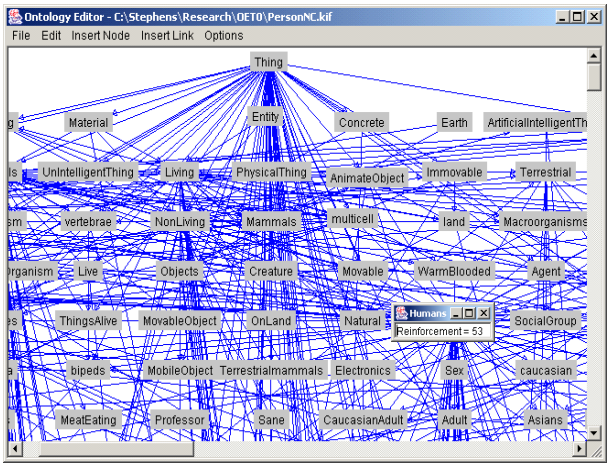

Figure 2: A portion of the ontology formed by merging 54 independently constructed ontologies for the domain Humans/People/Persons. The entire ontology has 389 concepts related by 696 subclass links.

Next, we constructed a *consensus ontology* by eliminating weakly reinforced nodes and links. In filtering the merged file, we sorted the links by their reinforcement values and found that, for the most part, the strongly reinforced nodes were associated with strongly reinforced links. This finding, while not surprising, makes constructing a consensus ontology more efficient.

The consensus ontology for the domain of *Humans* consists of 20 classes related by 25 subclass links (see Figure 3). The class *Humans* 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. In this figure, all nodes are reinforced at least 5 times and all links, except as noted, reinforced at least 3 times. The weakly reinforced link *Female–Women* could be omitted but illustrates the transitive closure considerations, which are discussed next.

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. However, this approach can introduce results that clearly violate a consensus view. In Figure 3, *Humans* has subclass *Fe-*

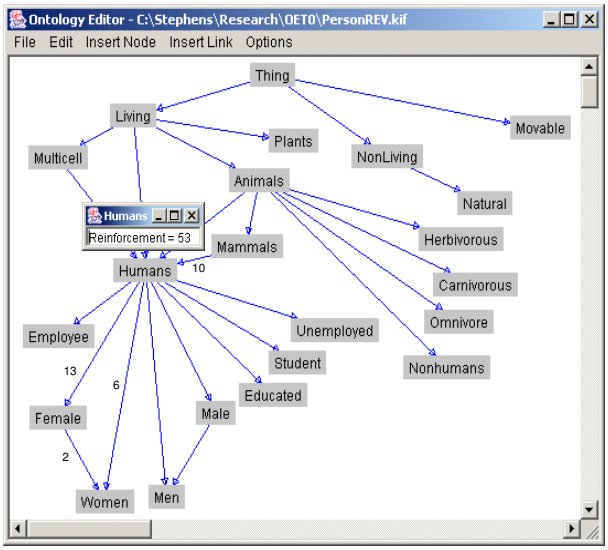

Figure 3: The consensus ontology for the *Humans* domain formed by merging concepts with common subclasses and superclasses from 54 component ontologies. The resultant ontology contains 20 concepts related by 25 subclass links.

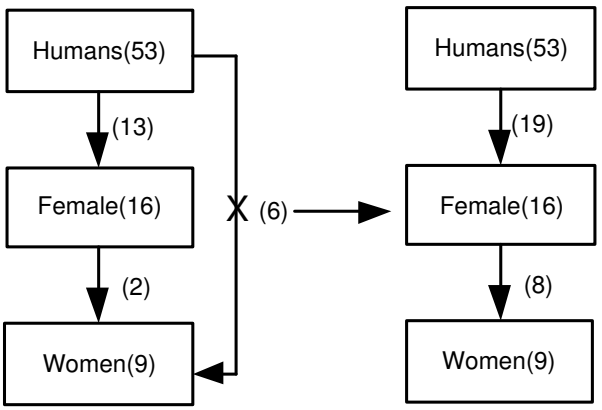

Figure 4: The consensus is that the concept *Women* is more strongly linked to *Humans* than *Female*. Removing the direct link from *Humans* to *Women* and reinforcing remaining links violates that consensus. Node and link reinforcements are shown in parentheses.

male with reinforcement 13, *Female* has subclass *Women* with reinforcement 2, and the direct subclass link from *Humans* to *Women* has reinforcement 6. Removing the direct link and reinforcing the remaining links (as in Figure 4) would give the *Female–Women* link a reinforcement value of 8—much stronger than the consensus view indicates. Our conclusion was to abandon this procedure and leave link reinforcement values unchanged.

Figures 5 and 6 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.

4 Discussion, Limitations, and Conclusions

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

Our results could be useful in the following scenario: suppose a user, interested in a comparison of the conductivity of aluminum versus copper wire, initiates a simple search on the term *conductor*. A recent Google^{TM} search for *conductor* returned a ranked list of 1,980,000 Web pages, some of which concern orchestra and railroad conductors. Our methodology could be used to 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 7. Based on this, the user could select a node to retrieve a page, or iterate by selecting a node from which to initiate a refined search.

Our experiments and analysis are preliminary and ongoing. However, the results so far support the hypothesis that a multiplicity of ontology fragments can be related automatically without the use of a global ontology. We used the following simplifications in our work:

- We did not make use of the properties of the classes, as would be the case for a complete implementation of subsumption.
- Our string-matching algorithm did not use a thorough morphological analysis to separate

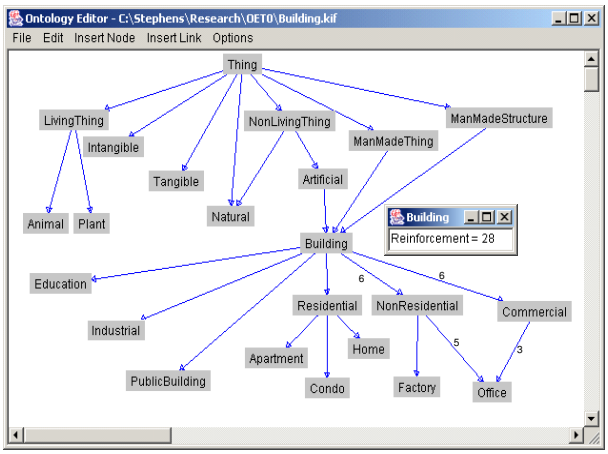

Figure 5: The consensus ontology for the *Building* domain contains 23 concepts and 26 links. *Office* is considered both *NonResidential* and *Commercial*. The concepts *Plant* (a subclass of *LivingThing*) and *Factory* (a subclass of NonLivingThing) appear in different branches of the ontology. The merged ontology is derived from 28 component ontologies.

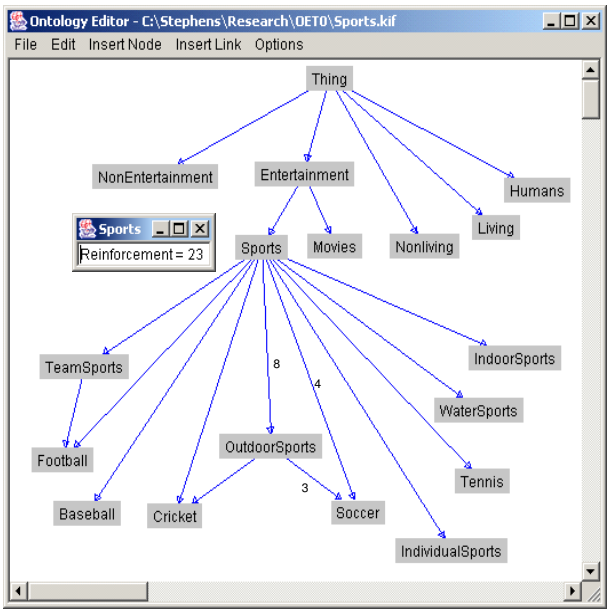

Figure 6: The consensus ontology for the *Sports* domain has 18 concepts and 20 links. *Soccer* is classified slightly more strongly as a subclass of *Sports* rather than of *OutdoorSports*.

the root word from its prefixes and suffixes. We do, however, handle singular and plural noun forms in most cases, and discriminate between obvious antonym pairs.

- Noun classifiers were detected by a string-matching heuristic. Breaks in compound nouns need to be identified in a more principled way, such has a blank space, hyphen, or case change. Unfortunately our data sets did not adhere to a uniform convention for compound noun representation.

- We used only subclass-superclass information, and have not yet made use of other important relationships, notably *partOf*.

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, 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.

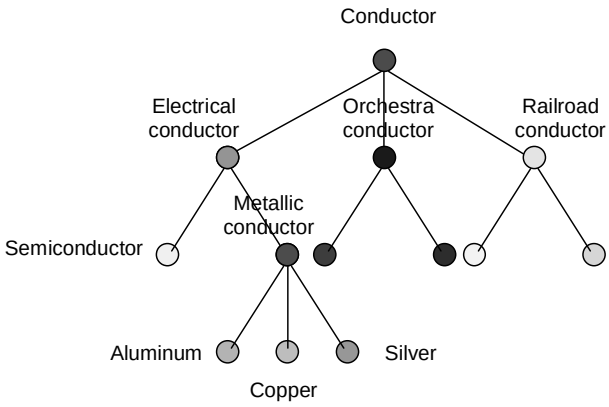

Figure 7: A merged ontology refines the domain concepts needed by users to satisfy their requests.

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] 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.

[3] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Scientific American*, 284(5), 34–43, 2001.

[4] B. Chandrasekaran et al., "What are ontologies, and why do we need them?," *IEEE Intelligent Systems*, 14(1), 20–26, 1999.

[5] M. Dean and G. Schreiber (eds.), "OWL Web Ontology Language 1.0 Reference," March 31, 2003, http://www.w3.org/TR/owl-ref/.

[6] 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.

[7] 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.

[8] 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.

[9] V. Kashyap and A. Sheth, *Information Brokering across Heterogeneous Digital Data: A Metadata-based Approach*, Kluwer Academic Publishers, Boston, 2000.

[10] 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.

[11] D. B. Lenat and R. V. Guha, *Building Large Knowledge-Based Systems*, Addison-Wesley, Reading, MA, 1990, http://www.cyc.com.

[12] G. A. Miller, "WordNet: A Lexical Database for English," *Communications of the ACM*, 1995 38(11), 39–41, 1995.

[13] 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.

[14] 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

[15] 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.

[16] C. Welty, "The Ontological Nature of Subject Taxonomies," in N. Guarino (ed.), *Formal Ontology in Information Systems*, IOS Press, Amsterdam 317–327, 1998.

Appendix: Node-Name Matching Algorithm

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. 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.