Diverse Web Ontologies: What Intelligent Agents Must Teach to Each Other

Andrew B. Williams and Costas Tsatsoulis

Department of Electrical Engineering and Computer Science
University of Kansas, Lawrence, KS 66045
{williams, tsatsoul}@ittc.ukans.edu

Abstract

This paper describes our ongoing research in multiagent learning among intelligent agents with diverse ontologies for World Wide Web distributed knowledge sharing and integration. Much research has been done on agent knowledge sharing through the use of common, pre-defined ontologies. However, several domains, including the World Wide Web, often precipitate intelligent agents selfishly inventing ontologies based on their utility for the task at hand. If such agents are able and willing to teach each other concepts based on their individualized ontologies, then the entire group of agents can accomplish their group and individual tasks more efficiently and with more creativity. We describe the hypotheses we are investigating and our proposed methodology for multiagent learning with diverse ontologies using the World Wide Web domain.

Introduction and Motivation

Everyday, World Wide Web users travel to previously visited Web sites using URL's they saved in a file, or bookmark list. Web browsers, such as Netscape Navigator, can be used to store these bookmarks in a graphical hierarchy of categories. Bookmarks are a form of indexing which are intended to be a navigation aid that helps reduce the cognitive loads associated with navigating the Web (Berghel 1996). A Web user can group Web page addresses which he or she thinks are similar or related into the same category. The user usually will create a label name for the categories to help him or her remember the kind of Web pages pointed to by that group of URL's.

Web search services, such as Yahoo! group similar and related Web page URL's into a hierarchy of concepts, or ontology. This type of ontology has thousands of categories and is manually indexed by Yahoo!'s ontologists. Several millions of dollars are spent to create and maintain this centralized database of Web page URL's. When a user visits Yahoo!'s site in order to search for a Web page, they can either follow the Web links in the ontology to find Web pages of interest or use its search engine which uses standard information retrieval techniques, such as term frequency and inverse document frequency indexing.

Individual users of the Web usually do not try to create bookmark categories, or ontologies, that match Yahoo!'s ontology. This would necessitate having to memorize several thousand categories and the relationships between them. Instead, Web users usually create their own ontology to help them categorize their Web pages under names that are easy for them to remember. So instead of using Yahoo's ontology concept description for Beethoven music, "Entertainment:Music:Genres:Classical:Composers: Classical Period:Beethoven," a normal user may put it in a bookmark category such as, "My Favorite CD's:Beethoven Jams."

In order to allow users to locate and share similar or related Web pages, we are developing a multiagent system for intelligent agents with diverse Web ontologies. We envision this type of intelligent agent eventually representing each of the millions of Web users. These agents will use the manually created ontologies of their respective users to teach each other what other agents know according to their own points of view. As these agents learn what other agents know and where they are located, the amount of time to locate, share, and integrate knowledge known by other agents should decrease and the quality of information and knowledge shared should increase. This multiagent system will be used to investigate the theory and algorithms for learning ontologies in a multiagent system. We believe that this research can be used in the future to develop millions of personalized Web agents that learn how individual users index the Web. This group of agents will then teach each other new, similar, or related concepts in order to solve the task of finding relevant information and knowledge.

Related Work

Previous research has focused on using a pre-defined, common ontology to share knowledge between agents by using a common set of ontology description primitives such as KIF (Genesereth and Fikes 1992) and Ontolingua (Gruber 1993). However, the approach of using global ontologies has problems due to the multiple and diverse needs of agents and the evolving nature of ontologies (Mineau 1992). Gruber (1991) also identifies this issue by raising the question on how group consensus can be reached on "what to represent" given that agents have commitments to different tasks, representation tools, and

domains.

Weinstein and Birmingham (1997) are conducting research in agent communication with *differentiated ontologies*, concepts that are not shared, but inherit structure from concepts that are shared. They are using rough mapping to identify syntactic and semantic similarity between graphs of concepts with description logic. Unlike most approaches, they do not translate to a central, shared language, but allow agents to communicate directly. However, they assume that the unshared terms inherit from terms in shared ontologies while we do not assume our agents use shared ontologies. Also, their research is not focused on multiagent learning, improving the group performance of the agents through experience.

Approach

Our approach addresses the current weaknesses to sharing knowledge among distributed agents by introducing a theory for learning ontologies, which includes semantic concept learning and interpretation, semantic concept sharing, and semantic concept translation. This theory combines agent communication and machine learning with two novel methodologies: a) recursive semantic context rule learning and b) concept cluster integration. Our agents use semantic concept learning and interpretation to learn and evaluate knowledge from other agents. Recursive semantic context rule learning is used to increase the interpretation quality of new concepts defined with different vocabularies. Concept cluster integration is used to discover how agents' concepts are related even though they are defined by diverse ontologies. Semantic concept sharing describes how agents interact to discover similar or related concepts from each other using concept-based queries. Semantic concept translation is used to teach agents what another agent's concepts mean from their own point of view. Our approach demonstrates how agents perform knowledge integration and translation without a common ontology by using machine learning and agent communication to teach each other about each other's ontologies, or views of the world, as it relates to their own.

Ontological Diversity

The DARPA Knowledge Sharing Effort (KSE) (Finin, Labrou, and Mayfield 1997) realized that various knowledge-based programs could not share knowledge because they were based on differing ontologies. Researchers associated with the KSE effort sought to be able to re-use knowledge bases by creating common ontologies in order to facilitate sharing knowledge. According to Gruber and Olsen (1994), the ontology vocabulary defines the ontological commitments among agents that are agreements to use the shared vocabulary in a coherent and consistent manner. This is where our approach for sharing knowledge diverges from the KSE approach.

We argue that different agents may invent their own ontologies based on their utility pertaining to their goals and needs and, as a result, the agents will not share a common, pre-defined ontology. There is a lack of a commitment to a common conceptualization by this type of agent. Genesereth and Nilsson (1987) described this essential ontological diversity of artificial intelligence: any conceptualization of the world is accommodated, and is invented by an agent based on its utility. Due to this ontological diversity, individual agents may create ontologies suitable for their own problem solving needs even if they are describing the same world or domain. An agent may not want to commit to an ontology a priori in order to facilitate future communication and sharing of its knowledge. The agent may selfishly create its own ontology in order to explain concepts relevant to its own problem solving needs within its domain. In order to share knowledge, a group of these ontologically diverse agents must learn to semantically interpret and integrate differing vocabularies in their ontologies. For a group of agents, the vocabularies for their ontologies may consist of word labels assigned to different semantic concepts or the individual words that make up an instance of a semantic concept, or class of instances. For our purposes, we describe an ontology as everything an agent knows and can learn about the world. Therefore, an ontology consists of the agent's known objects, classes of objects, interrelationships between objects in the world, and the learning and reasoning mechanisms required to learn and share its ontological knowledge.

Distributed Collective Memory

Agents can share a centralized memory, a distributed memory, or a hybrid memory (Garland and Alterman 1996). A distributed collective memory is defined as a set of base memory objects in the world which can be globally accessed but are selectively stored and conceptualized by individual agents. Agents may share a distributed collective memory but not share a common ontology to use when conceptualizing their known portion of the world. In this situation, it is assumed that no one agent knows and has conceptualized all of the objects existing in the world. Problems arise when these agents wish to share knowledge.

This phenomenon of agents sharing the same base objects distributed in a collective memory but having diverse ontologies can be seen in the World Wide Web domain, among others. Agents, representing human users of Web browsers, seek to find information located on the World Wide Web related to their individual needs and interests. Once an individual user, or agent, finds information that satisfies an interest or need, often he or she will save it in a bookmark, or hotlist. Web browsers allow the user to organize these bookmarks into a hierarchy using graphical folders. These bookmark hierarchies become a type of ontology for the user. They create concept categories by naming a bookmark folder. Within each bookmark folder, the user can place and name other folders or bookmarks. The bookmarks contain the name, description, and location of a Web page that fits into a particular concept category. Individual users create this

concept taxonomy, or ontology, based on their own view of the world (i.e. conceptualization). If a user were to interpret and categorize these bookmarks according to a common, pre-defined global ontology, it would necessitate having to know hundreds, if not thousands, of categories. This obviously is impractical and cumbersome for the average user. Each user creates these Web page ontologies based on their individual needs and interests. An agent created to find other agents who share concepts and concept instances but conceptualize them differently would not be able to use a global ontology to share knowledge. Often, users invent novel concept names such as, "Jimmy's Hype Hip Hop Tunes", or "McGwire's Latest Hits" in their personalized Web ontology. In this type of situation which occurs among Web users daily, an agent would need another mechanism besides using a global ontology to find other agents with similar or related concepts. A Web agent that searches other agent's bookmarks would not be able to perform a syntactic match of ontology concept names in order to find similar or related concept categories.

Semantic Concept Learning and Interpretation

The multiagent learning system we are developing contains a machine learning component to learn interpretation rules from an agent's Web ontology. The agent learns semantic concept description rules for each of the concepts in its ontology. An example of a semantic concept description may consist of the following:

- Rule 1: If (NCAA present) and (hoops present) and (NBA absent) then concept is "My college basketball links"
- Rule 2: If (March present) and (Madness present) and (championship present) then concept is "My college basketball links"

The discriminating features in the rules' pre-conditions are the semantic descriptors. A concept interpreter is made up of the agent's concept interpretation rules plus the rule-based inference engine used to interpret new Web pages representing a semantic concept. For each learned interpretation rule there is an associated degree of certainty represented by the percentage that this rule successfully classified a concept in the training set.

We represent each Web page, or semantic object, as a vector of boolean features. Each word or HTML tag in a Web page is used as a feature. We tokenize each agent's Web pages within their ontologies to find a vocabulary of unique tokens for each agent. This vocabulary is used to represent a Web page by a vector of ones and zeroes corresponding to the presence or absence of a token in a Web page. This combination of a unique vocabulary and vector of corresponding ones and zeroes is called a *concept vector*.

If an agent with a semantic concept description represented by rules 1 and 2 is presented a Web page containing the words, *NCAA* and *hoops* then it will interpret it as describing the *college basketball* concept. The agent will use its semantic concept descriptions along with a rule-based inference engine to interpret new Web pages. The interpretation accuracy for an agent's set of concept descriptions will be tested using the agent's known set of concepts and concept objects. The percentage accuracy for each concept tested on its training data will be known as the concept's *positive interpretation threshold*. A concept's *negative interpretation threshold* is equal to one minus the positive interpretation threshold. In order for a new group of similar Web pages to be interpreted as belonging to a concept, it must be interpreted by a concept description above its interpretation threshold. In this case it can be said that the agent *knows* the concept. If this same group of similar Web pages is interpreted below a concept's negative interpretation threshold, then an agent does *not* know the concept. If the interpretation percentage value of a new group of Web pages is between the positive and negative interpretation thresholds then the agent *may* know a concept. The interpretation percentage value, or *interpretation value*, for a concept is the percent the agent's interpretation rules successfully interpret the Web pages as belonging to a particular concept. For example, suppose there is a new group of ten new Web pages. If the concept description rules interpret three of them as belonging to concept X, four of them belonging to concept Y, and three of them belonging to concept Z, then the interpretation values will be 0.3 for concept X, 0.4 for concept Y, and 0.3 for concept Z.

It is possible that a group of similar Web pages may be interpreted as belonging to more than one concept. In this case, the concept description that yields the highest interpretation accuracy will be chosen as the target concept, or the concept the Web pages belong to.

Recursive Semantic Context Rule Learning

A region of interpretation uncertainty may arise due to the different vocabularies within agents' ontologies. We address this issue by proposing a novel algorithm, recursive semantic context rule learning, which will be described in this section. This interpretation uncertainty situation arises when an agent attempts to interpret a new group of similar Web pages and the interpretation value falls between the positive and negative interpretation thresholds for a single concept description. In this situation the agent *may* know the concept these Web pages belong to. In this case, the agent will use recursive semantic context rule learning to attempt to increase the interpretation value above a concept's positive interpretation threshold. This will be done to determine if the agent actually does *know* the concept.

Semantic context rules are interpretation rules learned using a concept's semantic descriptors as target concepts to learn from. In this case, the semantic descriptor becomes a *sub-concept* for the original target concept. The semantic descriptors for rule 1 are *NCAA* and *basketball*. An agent's training set of Web pages is re-processed to find every Web page that contains the tokens *NCAA* and *basketball*,

and are then input into the semantic concept learner to learn semantic context rules for these semantic concept descriptors. The semantic context rules learned for this example may look like this:

- Rule 1.1: If (collegiate present) and (sports present) and (administration absent) then concept is "NCAA"
- Rule 1.2: If (round ball present) and (three-pointer present) then concept is "hoops"

It is important to note that all of the semantic concept descriptions and semantic context rules are specific to an individual agent's ontology. Since this is true, an agent has a vocabulary, or set of unique tokens that is limited by the Web pages it possesses knowledge for. We hypothesize that selectively learning semantic context rules will increase an agent's interpretation value and improve the likelihood that it may know a concept. This is because it attempts to learn rules for semantic descriptors that are key to interpreting Web pages according to its ontology. Once these semantic context rules are learned, the rule-based inference engine will process the new group of Web pages using these rules. If the preconditions of these semantic context rules are satisfied, the agent will infer that the corresponding semantic context is present. Using the above example, let us say that an agent has a group of Web pages for which it attempts to satisfy the semantic description rule 1. The Web pages have the semantic descriptors *hoops* present and *NBA* absent but has no knowledge of the *NCAA* descriptor. The agent uses a meta-rule, which states that a semantic context rule should be learned if an interpretation rule is only lacking one pre-condition being satisfied. This meta-rule specifies to the agent to re-process the Web pages using the missing semantic descriptor as a target concept to learn. This new semantic context rule (Rule 1.1) along with the existing semantic description rules will be re-applied to the new group of Web pages. If the tokens *collegiate*, *sports*, and *administration* are present in one of the Web pages, then the agent infers that the semantic context for the this sub-concept, *NCAA*, is present and infers that the interpretation rule for the concept, *college basketball*, is satisfied.

It is possible that one of the rule preconditions is missing in the Web page's newly learned semantic context rule. In this case, a meta-rule instructs the agent to recursively learn another rule for some of the semantic context rules' preconditions. The agent's meta-rules are responsible for guiding this recursive semantic context rule learning to increase interpretation accuracy without overly taxing the agent's computation resources with an exhaustive rule search.

Concept Cluster Integration

An agent may not definitively *know* or *not* know a concept but may have inferred that it *may* know several concepts. That is, a new group of Web pages may be interpreted with an interpretation percentage lying between the positive and negative interpretation for more than one semantic concept. In this case, another novel algorithm, *concept cluster integration*, is used to learn how an agent's ontology is *related* to the target concept describing the new group of Web pages. For example, an agent may infer from its interpretation rules that it *may* know the new group of Web pages as possibly belonging to the concepts *college basketball*, *university athletics*, or *sports scouts*. This would happen if the agent's interpretation rules for each of these concepts each have an interpretation value between the positive and negative interpretation values. We hypothesize that the concept cluster integration taught by agents in a multiagent system will improve their ability to solve problems as a group.

For this example, an agent's concept cluster integration component instructs the agent to re-process the Web pages with the concept objects for college basketball, university athletics, and sports scouts grouped together as one concept cluster. The agent then uses its semantic concept learner to learn new interpretation rules for the reformulated concepts. Then the agent re-inputs the new group of Web pages to its concept interpreter. If these Web pages have a new interpretation value above the positive threshold for the newly formed concept cluster, then the agent can infer that it knows concepts that are related to the semantic concept represented by the new group of Web pages. On the other hand, if the interpretation value is below the negative threshold, then the agent infers that its newly clustered concept is not related to the new group of Web pages. Also, this algorithm may try to cluster these groups in different combinations and repeat the above process. The newly inferred knowledge that this agent knows a *related* concept is stored in its knowledge base in the form of a rule which indicates which related concepts were clustered and the name of the new concept this new concept cluster is related to. This process of integrating ontologies can be used to help the group, as a whole, learn how their individual ontologies relate to each other. We believe this will help to decrease future communication costs as this type of group learning increases. Also, individual agent information precision and recall should increase.

Semantic Concept Sharing

Agents can learn to locate and share knowledge using concept-based queries. An agent may wish to find other agents that know similar or related concepts in order to learn new knowledge. Although these agents may all have access to the same base semantic objects (i.e. Web pages) none of them has global knowledge of all of the objects due to their huge number and their rapid growth. Yet each individual agent in this multiagent system can constantly learn new concepts or gather Web pages and learn to interpret them according to their own ontology. They can learn to locate agents with similar or related concepts using an agent interaction defined as a concept-based query. A *concept-based query* (CBQ) occurs when one agent sends example concept vectors along with the concept name and vocabulary vector to other neighboring agents, determines by the agents' responses who knows similar or related

concepts, and learns new knowledge. This new knowledge can be in the form of new similar or related concept or knowledge regarding another agent's ontology. For this concept-based query scenario, a neighboring agent is another agent that the querying agent knows where to locate.

When agents share knowledge using a concept-based query there is a querying (Q) agent and a responding (R) agent. The Q agent sends out a CBQ to its neighbors. The R agents use their concept interpreters to determine if they think they know similar or related concepts and send their response to the Q agent. This response is either positive or negative along with the concept name and type. A positive or negative response corresponds to an interpretation value above the positive or negative interpretation threshold, respectively. The concept name corresponds to the bookmark folder the Web pages belong to. The concept *type* indicates whether the answer to the query is a similar or related concept. If an R agent has a positive response to the CBQ, it will request permission to send examples of its similar or related concept back to the Q agent. The Q agent can then verify whether the R agents actually know a similar or related concept by using its own concept interpreter on the examples R sends to it. During low activity periods, the Q agent will re-run its semantic concept learner on its updated set of semantic concept objects as a form of truth maintenance.

If an R agent does not know a similar or related concept requested by the Q agent, it can forward the query to one of its neighbors. This routing of queries can continue throughout the network of agents until a positive response is located. The initial CBQ contains the maximum number of times it should be forwarded to neighboring agents.

Semantic Concept Translation

Semantic concept translation is required when two agents learn they have similar concepts with different names. Bond and Gasser (1988) stated that agents may have disparate references which lead them to refer to the same object or concept using different terms and viewpoints, i.e. diverse ontologies. Semantic concept translation knowledge can also help an agent determine the optimum places to send its CBQ. When a Q agent receives a positive response from an R agent, the Q agent determines whether it recognizes the concept name or whether it has to translate it. If the concept name agent Q used for its query matches the R agent's concept name, then no translation is required. However, if the R agent's concept name does not match but the examples it sends corresponds to the Q agent's concept it based its query on, then a translation rule must be learned by both the Q agent and the R agent. For example, the Q agent may send out a query based on its concept X, and the R agent may reply with a positive response using a concept name Y. The Q agent can then learn a rule, which states that the particular R agent knows Q agent's concept X as concept Y. Likewise, the R agent learns that the particular Q agent knows R agent's concept Y as concept X.

Agents can use these translation rules to decrease future communication costs. If an R agent receives a CBQ with a concept name Z and has a translation rule for concept Z that indicates which agent(s) knows this concept, it can redirect the query to the appropriate agent(s) with minimal communication costs.

Conclusion and Future Work

Our theory for learning ontologies among agents with diverse conceptualizations to improve group semantic concept search performance through experience was overviewed in this paper. This theory involves learning ontologies using semantic concept learning and interpretation, semantic concept sharing, and semantic concept translation. We introduce two novel algorithms, recursive semantic context rule learning and unsupervised concept cluster integration, to address how agents teach each other to interpret and integrate knowledge using diverse ontologies. We are in the latter stages of completing our multiagent learning system, which we will use to test the validity of this theory. We have developed a multiagent architecture using Java and CORBA along with a limited set of KQML performatives to run our experiments. This theory for teaching agents how to learn ontologies of other agents has application not only for the World Wide Web, but also for a domain such as collaborative design.

Acknowledgements

This investigation was supported in part by a University of Kansas General Research allocation #3129-20-0003-111.

References

Bond, A. H., and Gasser, L. eds. 1988. *Readings in Distributed Artificial Intelligence*, Morgan Kaufmann, San Mateo, CA.

Berghel, H. 1996. The Client's Side of the World Wide Web, *Communications of the ACM* 39(1), 30-40.

Finin, T., Labrou, Y., and Mayfied, J. 1997. KQML as an Agent Communication Language. In Jeffrey Bradshaw (ed.), Software Agents, MIT Press.

Garland, A. and Alterman, R. Multiagent Learning through Collective Memory. In Adaptation, Coevolution, and Learning in Multiagent Systems, Technical Report, SS-96-01, AAAI Symposium, Stanford, CA, March 25-27, Menlo Park, CA, AAAI Press, 33-38.

Genesereth, M. and Fikes, R. 1992. Knowledge Interchange Format Manual Version 3.0, Technical Report, Logic-92-1, Stanford Logic Group, Stanford University.

Genesereth, M. and Nilsson, N. 1987. *Logical Foundations of Artificial Intelligence*, Palo Alto, CA, Morgan Kauffman Publishers.

Gruber, T. 1991. The Role of Common Ontology in Achieving Sharable, Reusable Knowledge Bases. In James A. Allen, Richard Fikes, & Erik Sandewall (Eds.), *Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference*, Cambridge, MA, Morgan Kauffman, 601-602.

Gruber, T. 1993. A Translation Approach to Portable Ontology Specifications, *Knowledge Acquisition*, 5(2):199-220.

Gruber, T. and Olsen, T. 1994. An ontology for engineering mathematics. In J. Doyle, P. Torasso and E. Sandewall (eds.), *Fourth International Conference on Principles of Knowledge Representation and Reasoning*, Bonn, Germany, Morgan Kaufmann.

Mineau, G. 1992. Sharing Knowledge: Starting with the Integration of Vocabularies. In H. Pfeiffer and T. Nable (eds.), *Conceptual Structures: Theory and Implementation, Proceedings of the Seventh Annual Workshop*, Las Cruces, NM, July 8-10, Springer-Verlag, 34-45.

Weinstein, P. and Birmingham, W. 1997. Matching Requests for Agent Services with Differentiated Vocabulary, Student Abstract. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAA-97)*, Providence, Rhode Island, July 27-31, 850.