Consensus Ontologies in Socially Interacting MultiAgent Systems

Ergun Biçici
Koç University
Rumelifeneri Yolu 34450
Sariyer Istanbul, Turkey

September 28, 2007

Abstract

This paper presents approaches for building, managing, and evaluating consensus ontologies from the individual ontologies of a network of socially interacting agents. Each agent has its own conceptualization of the world within the multiagent system framework. The interactions between agents are modeled by sending queries and receiving responses and later assessing each other's performance based on the results. This model enables us to measure the *quality* of the societal beliefs in the resources which we represent as the *expertise* in each domain. The dynamic nature of our system allows us to model the emergence of consensus that mimics the evolution of language. We present an algorithm for generating the consensus ontologies which makes use of the authoritative agent's conceptualization in a given domain. As the expertise of agents changes after a number of interactions, the consensus ontology that we build based on the agents' individual views evolves. The resulting approach is concordant with the principles of emergent semantics. We provide formal definitions for the problem of finding a consensus ontology in a step by step manner. We evaluate the consensus ontologies by using different heuristic measures of similarity based on the component ontologies. Conceptual processing methods for generating, manipulating, and evaluating consensus ontologies are given and experimental results are presented. The presented approach looks promising and opens new directions for further research.

1 Introduction

Language and consequently terminologies evolve over time. The non-existence of a shared global conceptualization of a domain, which we can refer to when resolving misunderstandings, requires us to develop methods to find specialized, task and context oriented solutions. In this vein, several special purpose ontologies have been developed for different domains. However, access to most of these ontologies is not straightforward and they are proprietary [Lenat et al., 1990].

An *ontology* is a thesaurus [Scott, 1986], which answers the question of "what there is" [Quine, 1986] in a domain. Ontologies present a structure over the language we use to represent the world. Semantic Web's dream is to share, exploit, and understand knowledge on the web [Berners-Lee et al., 2001]. The existence of a single ontology that can cover all the required conceptual information for reaching semantic understanding is questionable because it would presume an agreement among all ontology experts. Therefore, semantic agreement among heterogeneous ontologies is not always possible. In the most extreme case, different ontologies may not even have some shared lexicon; hence making communication impossible.

Another problem is that various ontologies exists for the same domain but it is hard to decide which one provides the best conceptualization. The quality of the statements can also vary within each ontology. Thus, there is a need to find models of building consensus among diverse sources of statements. In this paper, we address the problem of building consensus ontologies which represent the consensus from multiple heterogeneous ontologies belonging to a number of agents interacting with each other.

Motivation. Forming a consensus ontology is important for multiple reasons. First of all, it provides us with a vocabulary to which agents can refer when they encounter misunderstandings in communication. Furthermore, it represents a unified world view supported by the members, which facilitates distributed knowledge management. In terms of language, building consensus ontologies can be regarded as an effort for reaching what Quine calls the "maxim of shallow analysis" [Quine and Ullian, 1978], the common ground of beliefs, which are no more particular or detailed than what is necessary for agreement. Any information system that makes use of different sources of knowledge needs to deal with the management of heterogeneous representations and conflicting statements. The questions that need to be addressed include: (i) How can a consensus ontology be generated and conflicting conceptualizations of the world be resolved? (ii) How can concepts that are conceptualized or referred differently be related? (iii) How

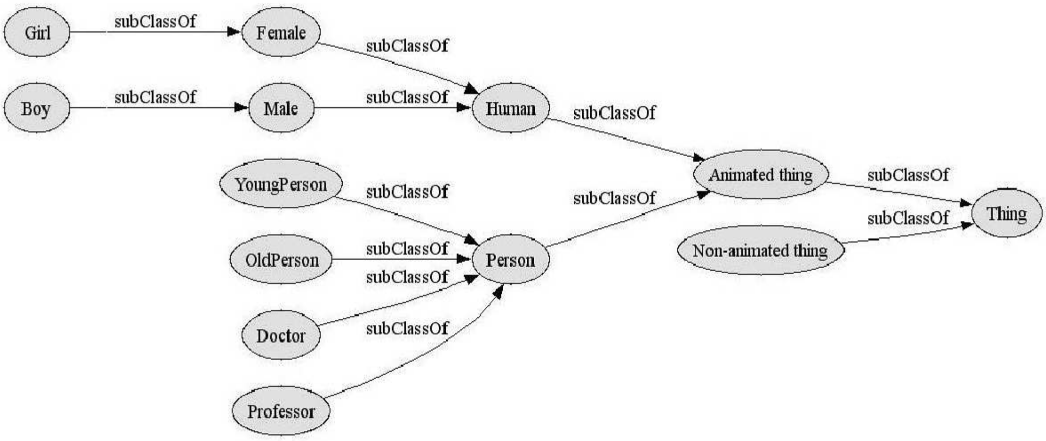

Figure 1: Sample ontology from the data set

can the goodness of the consensus ontology be evaluated?

The objective of finding a single, shared ontology is challenging, not only due to the difficulty of imposing a universal standard on ontologies, but also because of the virtual impossibility of reaching an agreed upon conceptualization among different sources. Stephens and Huhns [Stephens and Huhns, 2001] show the difficulties in reaching an agreement even for a general domain like "humans" (an example ontology from the Stephens and Huhns data is given in Fig. 1). We believe that reliably close approximations of consensus ontologies can be found by sound mathematical models and we regard our work in this direction.

Technical Challenges. Our goal is to reach semantic agreement among different world views shared by different agents. Some technical difficulties are as follows:

- *Conceptual mapping:* A concept belonging to the ontology of an agent need not necessarily be present in other ontologies due to the heterogeneity of conceptualizations. Therefore, we need to be able to find mappings between conceptual elements belonging to different ontologies.

- *Conflict resolution:* Finding consensus among sets of statements is not easy since they may contain conflicting elements with each other. As Arrow's *social choice impossibility theorem* [Arrow, 1963] states, there can be no general method for reaching a global preference order that will obey all of the preferences specified by the members of a society.

- *Consensus generation:* What is a good way to generate a consensus ontology, which can closely approximate a model of the consensus?
- *Consensus evaluation:* Measuring the goodness of the final consensus is not easy since each agent maintains an individual world view.

Contributions. The interactions in a social network enable us to model the societal beliefs in the *quality* of resources as *expertise* in a given domain. Our approach for building the consensus ontology is based on combining the beliefs of experts in each domain where expertise is gained by agents through social interactions. The framework that we use is based on the social interactions of agents in a referral based multiagent system. The system collaboratively builds the consensus ontology based on the evolving values for the expertise in each domain. The multiagent system framework provides us with a rich formalism with which we can model the social interactions and the dynamic nature of the environment.

The system that we have developed has the following contributions. First, we are able to model the emergence of consensual agreements among socially interacting agents. Second, we developed heuristic measures for evaluating the consensus ontology based on three different levels of abstraction. Third, we present a method of concept mapping based on the conceptual structures in the ontologies.

Related Work. The naive approach will assign each resource (which can be computationally represented as an RDF triplet [World Wide Web Consortium (W3C), 2004]) from each agent an equal weight such that the statements with the majority of the votes win. Thus, the triplets that are not voted enough are voted off or silenced. This statistical reinforcement formulation is done by Stephens and Huhns [Stephens and Huhns, 2001], which is likely to result in conflicting and non realistic set of statements. Aberer *et. al.* [Aberer et al., 2003] present a framework for query transformation and a method for detecting semantic agreements in which peers transform queries based on their local schema and their already existing mapping functions between schemas. The approach is named semantic gossiping. Emergent semantics [Aberer et al., 2004] is a recent term being used for the emergent phenomena regarding the semantic interoperability. The system we have ddevelopedis complying with all the principles of emergent semantics [Aberer et al., 2004] since: (1) the consensus ontology and each agent's own beliefs and own local ontology evolve via a process of forming consensus and a semantic handshake mechanism that happens among agents where agreement is effected by the quality, strength, and the trustworthiness of the statements and agents; (2,3) consensus ontologies emerge from the local interactions and the negotiations held among agents; (4) the consensus ontology is a dynamic and self-referential approximation of the consensus and evolves over time as the agents interact with each other and change their conversational context; (5) the consensus ontology building is effected by local interactions and aagreements which decentralizes the control to the interacting agents where autonomy is preserved; (6) it can be a model for peer-to-peer data management and result with more accurate global semantic agreements.

Campbell and Shapiro [Campell and Shapiro, 1998] attempt to find algorithms for determining the meanings of unfamiliar words by asking questions. Their approach resolves terminological mismatches with an ontological mediator. Matchmaking is a process used in semantic web applications for finding appropriate services for given queries using description logics reasoners [Li and Horrocks, 2003]. Building consensus ontologies facilitates knowledge sharing and has applications in service composition [Williams et al., 2003]. According to the categorization and the organization of the material presented in state of the art in ontology alignment [Euzenat et al., 2004], we are using both local and global methods for aligning concepts and generating consensus ontologies. The local methods that are employed in our work falls under the heuristic techniques that use terminological and structural techniques. The global methods that we employ involve compound and global similarity computations, learning methods that can fall under the category of semantic gossiping, and alignment extraction techniques via thresholding.

Noy [Noy, 2004] discusses techniques for finding correspondences between ontologies. For establishing the smallest set of concepts to be used in agent communication, previous work assumes that agents share some minimal common ground which can be used to learn new concepts [van Diggelen et al., 2004]. Algebraic methods for merging ontologies when mappings between ontologies are known are presented by Mitra and Wiederhold [Wiederhold and Mitra, 2001]. Formal concept analysis was used for merging ontologies employing instances and features of concepts defined in individual ontologies [Stumme and Maedche, 2001].

Sections. The next section investigates representations for the interoperability of semantic information and provides formal definitions for the problem of finding a consensus ontology in a step by step manner. Sect. 3 introduces the formal presentation of the problem of building consensus. We discuss several abstraction levels for comparing ontologies such as lexical, conceptual, or information retrieval. We also discuss methods for mapping concepts. Sect. 4 introduces social networks of agents and how they communicate and collaborate with each other from the perspective of building consensus. In Sect. 5, we present our methods for building consensus ontologies and in Sect. 6, we present our experiments and results. The last two sections present the future work and the conclusion.

2 Semantic Information and Consensus Ontologies

There has been extensive work on representing and employing information about semantics. In this section we discuss efforts that share close relations with our interpretation. We gain a broader perspective by investigating philosophical approaches and mathematical representations for semantic information. We also provide formal definitions for the problem of finding a consensus ontology in a step by step manner.

2.1 Resolving Misunderstandings at the Sign Level

Charles Peirce's semiotics acts as a model to derive a mathematical representation for the transmission of semantic information in a scenario close to the one used in agent communication. In simple terms, when I is the interpretant, S is the sign, and O is the object, I interprets S as a sign of O [Hookway, 1992]. This is a triadic mathematical representation for shared semantics excluding the instances of objects. Since the context is a necessary and essential part of translation and understanding, we assume that it is stored or derived by I.

Another way to represent meaning is through an ostensive perspective via using instances or examples. Quine asserts that sentences with the same meaning can be identified by specifying the circumstances under which two sentences have the same meaning [Quine, 1995]. An ostensive mathematical model for concepts and contexts exists through formal concept analysis (FCA) [Ganter and Wille, 1999]. The universe is viewed in a formal context which has the concept as its basic unit having instances and attributes as its building blocks. FCA represents a context as a triple (E, A, F) where E and A are sets of examples (extents) and attributes (intents) and $F : E \rightarrow A$ is a mapping function. All attributes thought with a concept is called its *intension* and all instances for which the concept can be predicated is called its *extension* [Stumme, 2002]. A concept in the context (E, A, F) is an $(\textit{extent}, \textit{intent})$ pair where $\textit{extent} \subseteq E$, $\textit{intent} \subseteq A$, $F(\textit{extent}) = \textit{intent}$, and $F^{-1}(\textit{intent}) = \textit{extent}$. Thus, $F^{-1}(F(\textit{extent})) = \textit{extent}$ and $F(F^{-1}(\textit{intent})) = \textit{intent}$.

In semiotics, context takes part in the translation of signs to objects, whereas in FCA, context is formed through an ostensive and comprehensive definition by the instances and the attributes of objects. Ostensive semiotics can be derived by combining these two ideas. That is, we can define objects with their extents and intents as it is done in FCA. Then, the interpretant would infer the context as it is done by FCA.

Triadic world refers to the Peircean interpretation of the world and the *dyadic world* refers to a world when the contextual information is immaterial or accepted as common sense. Semantic information about an object is stored as a tuple (S, O, I) in the triadic world. The shift from the dyadic to the triadic world require us to define the notion of context either via ostension as it is done in FCA or via other definitions such as the set of queries posed between the communicating pair, forming the conversational context.

Given two interpretants, I_i and I_j, trying to communicate semantic information about two different objects O_i and O_j signified by S_i and S_j correspondingly, the type of misunderstandings that might be encountered between the communicating pair can be classified as follows:

1. **Absence of semantic information:** An equivalent semantic information might not be present. An interpretant of 1930's will not be able to understand "neutrinos" as it will be untranslatable [Quine, 1995]; yet the listening pair might be able to interpret it.

2. **Syntactic misunderstandings:** Two equivalent objects might have different signs; thus the signs are *synonymic* (e.g. Morning star and Evening star). Examples include the use of different languages (e.g. liebe vs. love), spelling variations within languages (e.g. colour vs. color), misspellings in entry, mishandling of compound names (e.g. commonsense vs. common-sense), varying representational constraints for the same concept (e.g. local phone number representation vs. national), and synonymity.

3. **Conceptual misunderstandings:** Two syntactically equivalent signs might signify different objects, that is conceptual implications; thus they are *homonymic*. Examples include words having different senses and conceptual interpretations.

4. **Pragmatic misunderstandings:** The context in which an object is interpreted changes its semantics. This change can move two objects' semantic information closer or distant. For instance, "episode" has the same sense [Fellbaum, 1998] with "part" when used in the context of a play but in the context of medicine, it refers to the occurrence of an illness.

Even if none of these cases of misunderstandings exist, translation might not be known due to Quine's indeterminacy of translation thesis [Quine, 1970], which states that because of freedom of choice, the exact translation might not be determined from possibilities that arise from observations. When viewed as a learning problem, there may exist various functions that fit the observed data; yet it is hard to determine the exactly the same intended function among the vast amount of functions that are correct.

The additional advantage of a multiagent system is the ability of asking other agents when resolving misunderstandings. The process by which we use other agents' resources to find semantic mappings between a concept that we do not understand with a concept that we do may be called as forming a "semantic bridge" [Stephens and Huhns, 2001]. By using the bridging agents' resources, we can establish a link between a previously unknown object to an already known object that was in our agent's resources.

2.2 The Problem of Finding a Consensus Ontology

In this section, we present a general mathematical representation for ontologies and the consensus ontology and provide an initial formulation of the problem of finding a consensus ontology. Formal definitions are provided as needed in a top-down fashion.

- Let $\mathcal{C}$ represent a set of concepts and $<_{\mathcal{C}} \subseteq \mathcal{C} \times \mathcal{C}$ be the "`subClassOf`" relation, which relates two concepts having the subclass of relation defining a partial order over the set of concepts.
- A statement S in $\langle \mathcal{C}, <_{\mathcal{C}} \rangle$ is a 3-tuple $(c_s, <_{\mathcal{C}}, c_o)$ where $c_s, c_o \in \mathcal{C}$ represent the subject and the object respectively such that $c_s <_{\mathcal{C}} c_o$.
- Two statements, $S_i = (c_{s_i}, <_{\mathcal{C}}, c_{o_i})$ and $S_j = (c_{s_j}, <_{\mathcal{C}}, c_{o_j})$, conflict with each other when c_{s_i} is equivalent to c_{o_j} and c_{s_j} is equivalent to c_{o_i}.
- A set of statements, $\mathbb{S}$, is consistent when it is conflict-free.
- An ontology is a 2-tuple $\langle \mathcal{C}, <_{\mathcal{C}} \rangle$ which represents a consistent set of statements defined over $\mathcal{C}$. Thus, $\mathcal{O} = \{S_1, \ldots, S_n\}$.
- Let $\mathcal{O}^{\cap}$ be the ontology that represents the intersection of a given set of ontologies, $\mathbb{O}$. Then,

$$\mathcal{O}^{\cap} = \bigcap_{i=1}^{n} \mathcal{O}_i = \langle \bigcap_{i=1}^{n} \mathcal{C}_i, <_{\cap} \rangle,$$

where $\mathcal{O}_i \in \mathbb{O}$ for all i, $1 \leqslant i \leqslant n$ and $<_\cap$ defines an ordering consistent with all of the given set of orderings. $O^\cup$ is defined similarly.

- Let $\mathcal{O}_i^*$ represent the set of all possible statements that are consistent with a given ontology $\mathcal{O}_i$ and let $\mathbb{O}^* = \bigcup_{i=1}^n \mathcal{O}_i^*$.
- A consensus ontology, $\mathcal{O}_C$, is an ontology which represents the consensus of a given set of ontologies.
- Given a set of ontologies, $\mathbb{O}$, let $\mathcal{O}_C$ represent the consensus ontology. Then,
$$\mathcal{O}^\cap \subseteq \mathcal{O}_C \subseteq \mathcal{O}^\cup \subseteq \mathbb{O}^*.$$
- A consensus ontology, $\mathcal{O}_C$, is *complete* with respect to an ontology $\mathcal{O}_i$ when all the statements of $\mathcal{O}_i$ are in $\mathcal{O}_C$. Thus, $\mathcal{O}_i \subseteq \mathcal{O}_C$.
- A consensus ontology, $\mathcal{O}_C$, is *consistent* with respect to an ontology $\mathcal{O}_i$ when none of the statements in $\mathcal{O}_i$ conflict with $\mathcal{O}_C$.
- A consensus ontology is *strong* when it is both complete and consistent with respect to an ontology.

Definition 2.1 (**Strong Consensus**)
Given a set of ontologies, $\mathbb{O}$, find an ontology $\mathcal{O}_C$ such that $\mathcal{O}_C$ is strong with respect to all ontologies in $\mathbb{O}$. □

The requirements for consistency and completeness are in practice hard to satisfy. Each ontology might be consistent in itself but it will likely to be inconsistent with others. Thus, there may be conflicting statements overall and which one to choose is not easy to decide. Even if the statements are not conflicting with each other, the union might not represent a consensus since not every statement is present in all ontologies and therefore they might not be supported by all. In a similar manner, although defining a consensus as the intersection of statements, the statements that are shared by all ontologies, will represent a consensus, it might lead to an empty set.

Let *HeuristicValue* be a function which returns a value for the goodness of a consensus ontology.

Definition 2.2 (**Learned Consensus**)
Find an ontology $\mathcal{O}_C$ such that *HeuristicValue* is maximized. □

$\mathcal{O}_C = \bigcap_{i=1}^{n} \mathcal{O}_i$ (*Initialization*)
$\mathbb{S} = \bigcup_{i=1}^{n} \mathcal{O}_i - \mathcal{O}_C$
while $\mathbb{S} \neq \emptyset$ **do**
    `Statement` = *FindBestStatement*$(\mathcal{O}_C, \mathbb{S}, \mathbb{O})$
    if *`Statement`* **then**
        $\mathbb{S}$ = $\mathbb{S}$ − `Statement` ;
        $\mathcal{O}_C$ = $\mathcal{O}_C$ ∪ `Statement` ;
    end
end
return($\mathcal{O}_C$) ;

Algorithm 1: Inductive Algorithm for Learning Consensus

Consensus learning may have two different *HeuristicValue* function definitions among others:

- Let *HeuristicValue* be a function where *HeuristicValue* : $\mathcal{O}_C \times \mathbb{O} \rightarrow [0, 1]$. Find $\mathcal{O}_C$ such that *HeuristicValue*$(\mathcal{O}_C, \mathbb{O})$ is maximal.
- Let *HeuristicValue* be a function where *HeuristicValue* : $\mathcal{O}_C \times \mathbb{O}^* \rightarrow [0, 1]$. Find $\mathcal{O}_C$ such that *HeuristicValue*$(\mathcal{O}_C, \mathbb{O}^*)$ is maximal.

Algorithm 1 presents a top-down, inductive learning approach to building a consensus ontology. A consensus ontology is reached by means of search through the space of possible formal statement configurations. At each step, the current best statement which has the highest *HeuristicValue*$(\mathcal{O}_C, \mathbb{O})$ is chosen by the *FindBestStatement* function and added to the consensus ontology. A number of heuristic measures that can be used for creating a *HeuristicValue* function is given in Table 1.

3 Consensus Ontology Generation, Management, and Evaluation

This section presents our formal introduction and framework to the problem of finding a consensus ontology among a given set of ontologies. Also, conceptual processing methods for building, managing, and evaluating consensus ontologies are given.

3.1 Problem Formulation

We define an ontology as a 2-tuple $\langle \mathcal{C},\ <_{\mathcal{C}} \rangle$ where $\mathcal{C}$ represents the set of concepts and $<_{\mathcal{C}} \subseteq\ \mathcal{C} \times \mathcal{C}$ is the "`subClassOf`" relation, which relates two concepts having the subclass of relation. $C_1 <_{\mathcal{C}} C_2$ denotes that C_1 is a subconcept of C_2 under the conceptual hierarchy of $<_{\mathcal{C}}$. A multiagent system (MAS) is a set of agents, $\mathbb{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$, where agents interact by asking each other questions and evaluating the answers they receive. Each agent $\mathcal{A}_i$ has an ontology $\mathcal{O}_i = \langle \mathcal{C}_i,\ <_{\mathcal{C}_i} \rangle$ and a lexicon, $\mathcal{L}_i$, which defines the set of allowable terms.

We use $\mathcal{O}^{\cap}$ to denote the ontology which represents the intersection of a given set of ontologies, $\mathbb{O}$, where $\mathcal{O}_i \in \mathbb{O}$ and $<_{\cap}$ defines an ordering consistent with all of the given set of orderings (equation 14, see Table 1). In Table 1, we define a number of heuristics for evaluating ontologies and their elements. We base some of our heuristics on Maedche's [Maedche, 2002] representation. We compare ontologies at three different levels of abstraction: lexical (equations 1, 2, and 3), conceptual (equations 4, 5, 6, and 7), and based on the measures of information retrieval (equations 8, 9, 10, 11, 12, and 13).

We compare two ontologies at *the lexical level*, by averaging over the syntactic similarities of their lexicon (equation 2). The string matching heuristic that we use, SM, is defined based on the edit distance, *ed* (equation 1). The $|.|$ operator used in the equations corresponds to the length of the lexical term or the size of the lexicon depending on the context. The similarity of the lexicon of the consensus ontology to the lexicons of component ontologies, $\mathbb{L}$, can be computed by averaging over all component ontologies (equation 3). Since $\overline{SM}$ is asymmetric, we take the arithmetic mean.

At *the conceptual level*, we use the similarity between the conceptual taxonomies of two given ontologies. The conceptual similarity between two concepts C_i and C_j is approximated by calculating the similarity between their ancestor sets (AS) (equation 4). Based on AS, we calculate the taxonomic similarity (TS) between two conceptual hierarchies $<_{\mathcal{C}_i}$ of $\mathcal{O}_i$ and $<_{\mathcal{C}_j}$ of $\mathcal{O}_j$ for a given lexical term (equation 5). When there exists a lexical entry L that is in $\mathcal{L}_i$ but not in $\mathcal{L}_j$, then we search for the maximum overlap among all those lexical entries of $\mathcal{L}_j$ (TS_2). We define the average taxonomic similarity between two ontologies, $\overline{TS}$ (equation 6), and compute the average similarity of the taxonomy of the consensus ontology compared to the taxonomies of component ontologies by averaging over all component ontologies (equation 7).

We can view building the consensus ontology task within the scope of information retrieval, where there exists a set of target elements that we are trying to

$SM(L_i, L_j) = max\left(0, \frac{min(\|L_i\|,\|L_j\|) - ed(L_i,L_j)}{min(\|L_i\|,\|L_j\|)}\right)$	(1)
$\overline{SM}(\mathcal{L}_1, \mathcal{L}_2) = \frac{1}{\|\mathcal{L}_1\|} \sum_{L_i \in \mathcal{L}_1} max_{L_j \in \mathcal{L}_2} SM(L_i, L_j)$	(2)
$\overline{\overline{SM}}(\mathcal{L}_C, \mathbb{L}) = \frac{1}{\|\mathbb{L}\|} \sum_{\mathcal{L}_i \in \mathbb{L}} \frac{\overline{SM}(\mathcal{L}_C,\mathcal{L}_i) + \overline{SM}(\mathcal{L}_i,\mathcal{L}_C)}{2}$	(3)
$AS(C_i, <_{\mathcal{C}}) = \{C_j \in \mathcal{C} \mid C_i \ <_{\mathcal{C}} \ C_j \ \vee \ C_i = C_j\}$	(4)
$TS(L, \mathcal{O}_i, \mathcal{O}_j) = \begin{cases} TS_1(L, \mathcal{O}_i, \mathcal{O}_j), & \text{if } L \in \mathcal{L}_j \\ TS_2(L, \mathcal{O}_i, \mathcal{O}_j), & \text{if } L \notin \mathcal{L}_j \end{cases}$	(5)
$\overline{TS}(\mathcal{O}_i, \mathcal{O}_j) = \frac{1}{\|\mathcal{L}_i\|} \sum_{L \in \mathcal{L}_i} TS(L, \mathcal{O}_i, \mathcal{O}_j)$	(6)
$\overline{\overline{TS}}(\mathcal{O}, \mathbb{O}) = \frac{1}{\|\mathbb{O}\|} \sum_{\mathcal{O}_i \in \mathbb{O}} \frac{\overline{TS}(\mathcal{O},\mathcal{O}_i) + \overline{TS}(\mathcal{O}_i,\mathcal{O})}{2}$	(7)
$precision(\mathcal{O}, \mathcal{O}_C) = \frac{\|elements(\mathcal{O}) \cap elements(\mathcal{O}_C)\|}{elements(\mathcal{O})}$	(8)
$recall(\mathcal{O}, \mathcal{O}_C) = \frac{\|elements(\mathcal{O}) \cap elements(\mathcal{O}_C)\|}{elements(\mathcal{O}_C)}$	(9)
$FMeasure(\mathcal{O}, \mathcal{O}_C) = \frac{2 \times recall(\mathcal{O},\mathcal{O}_C) \times precision(\mathcal{O},\mathcal{O}_C)}{recall(\mathcal{O},\mathcal{O}_C) + precision(\mathcal{O},\mathcal{O}_C)}$	(10)
$\overline{Precision}(\mathcal{O}_C, \mathbb{O}) = \frac{1}{\|\mathbb{O}\|} \sum_{\mathcal{O}_i \in \mathbb{O}} precision(\mathcal{O}_i, \mathcal{O}_C)$	(11)
$\overline{Recall}(\mathcal{O}_C, \mathbb{O}) = \frac{1}{\|\mathbb{O}\|} \sum_{\mathcal{O}_i \in \mathbb{O}} recall(\mathcal{O}_i, \mathcal{O}_C)$	(12)
$\overline{FMeasure}(\mathcal{O}_C, \mathbb{O}) = \frac{1}{\|\mathbb{O}\|} \sum_{\mathcal{O}_i \in \mathbb{O}} FMeasure(\mathcal{O}_i, \mathcal{O}_C)$	(13)
$\mathcal{O}^{\cap} =< \bigcap_{i=1}^{n} \mathcal{C}_i, <_{\cap}>$	(14)

Table 1: Heuristic measures for evaluating ontologies and their elements

retrieve, the consensus ontology, and a larger set that we choose from, the set of component ontologies. Equations 8-13 give the definitions for our information retrieval measures where the function *elements*$(\mathcal{O})$ returns the set of class lexical terms in the ontology $\mathcal{O}$. Precision corresponds to the proportion of selected lexical terms that the system got right (equation 8) whereas recall corresponds to the proportion of the lexical terms that the system selected (equation 9). Equations 11, 12, and 13 calculate the averages for precision, recall, and F-Measure values correspondingly. The closer the values are to 1, the better.

3.2 Mapping Concepts

This section presents our method of mapping concepts from different ontologies. Given two ontologies $\mathcal{O}_i$ and $\mathcal{O}_j$ with lexicons $\mathcal{L}_i$ and $\mathcal{L}_j$, let $L_i \in \mathcal{L}_i$ and $L_j \in \mathcal{L}_j$. A *mapping* function, $\mathbf{m}$, between $L_i \in \mathcal{L}_i$ and $L_j \in \mathcal{L}_j$ is a function whose domain is $\mathcal{L}_i$ of $\mathcal{O}_i$ and whose range is $\mathcal{L}_j$ of $\mathcal{O}_j$. Then, under the mapping $\mathbf{m}$, we can use L_j whenever we use L_i. Our method for concept mapping is given in Algorithm 2. The function *OCM* returns the level of *ordered conceptual match* between two concepts corresponding to the lexical entries in their respective ontologies. This function is based on the taxonomic similarity that we have defined. *OCM* is defined since subgraph isomorphism is known to be NP-complete [Garey and Johnson, 1979]. We have set the threshold levels for the concept mapping as 0.6, 0.3, and 0.5 for α_1, α_2, and α_3 correspondingly. Our experiments verify that this selection gives us good results. $\mathbf{m}(L_i) = L_j$ states that concept topic names L_i and L_j match with the mapping function $\mathbf{m}$.

Table 2 lists definitions for concept matching. We adopt the mathematical representation used in [Maedche, 2002] for formal ontologies. The relation $\mathcal{F} \subseteq \mathcal{L}_{\mathcal{C}} \times \mathcal{C}$ denotes *references* for concepts. Let for $L \in \mathcal{L}_{\mathcal{C}}$:

$$\mathcal{F}(L) = \{C \in \mathcal{C} \mid (L, C) \in \mathcal{F}\} \text{ and for } \mathcal{F}^{-1}(C) = \{L \in \mathcal{L}_{\mathcal{C}} \mid (L, C) \in \mathcal{F}\}.$$

We define abstractions for upwards cotopy (*UC*, equation 15), lexical concept match (*LCM*, equation 16), concept match (*CM*, equation 17), ordered upwards cotopy (*OUC*, equation 18), ordering match (*OM*, equation 19), and ordered concept match (*OCM*, equation 20). The context of a given concept may also be based on its downward cotopy; but we do not consider the downward cotopy, since we cannot get a total ordering between the elements of the set. *LCM* ignores the depth of the hierarchy considered in different ontologies. Highly specialized ontologies might use various levels when representing the same hierarchical compo-

$UC(C_i, <_{\mathcal{C}}) = \{C_j \in \mathcal{C} \mid C_i <_{\mathcal{C}} C_j) \vee C_i = C_j\}$	(15)
$LCM(L_i, \mathcal{O}_i, L_j, \mathcal{O}_j) = CM(\mathcal{F}(L_i), \mathcal{O}_i, \mathcal{F}(L_j), \mathcal{O}_j)$	(16)
$CM(C_i, \mathcal{O}_i, C_j, \mathcal{O}_j) = \frac{\lvert \mathcal{F}_i^{-1}(UC(C_i, <_{\mathcal{C}_i})) \cap \mathcal{F}_j^{-1}(UC(C_j, <_{\mathcal{C}_j}))\rvert}{\lvert \mathcal{F}_i^{-1}(UC(C_i, <_{\mathcal{C}_i})) \cup \mathcal{F}_j^{-1}(UC(C_j, <_{\mathcal{C}_j}))\rvert}$	(17)
$OUC(C_i, <_{\mathcal{C}}) = \{C_j \in \mathcal{C} \mid C_i <_{\mathcal{C}} C_j) \vee C_i = C_j\}_{\leqslant_{<_{\mathcal{C}}}}$	(18)
$OM(A_{\leqslant_A}, B_{\leqslant_B}) = \sum_{i=1}^{n-1} a_i \leqslant_A a_{i+1} \Leftrightarrow \mathbf{m}(a_i) \leqslant_B \mathbf{m}(a_{i+1})$	(19)
$OCM(C_i, \mathcal{O}_i, C_j,\ \mathcal{O}_j) = \frac{OM(OUC(C_i, <_{\mathcal{C}_i}), OUC(C_j, <_{\mathcal{C}_j}))}{min(\lvert OUC(C_i, <_{\mathcal{C}_i})\rvert, \lvert OUC(C_j, <_{\mathcal{C}_j})\rvert)}$	(20)

Table 2: Methods for mapping concepts

sition. For instance, given two ontologies, $\langle \mathcal{C}_i,\ <_{\mathcal{C}_i}\rangle$ and $\langle \mathcal{C}_j,\ <_{\mathcal{C}_j}\rangle$, such that:

$$\{C_i <_{\mathcal{C}_i} B,\ B <_{\mathcal{C}_i} A\}\ \subseteq <_{\mathcal{C}_i},\ \text{and}$$

$$\{C_j <_{\mathcal{C}_j} Y,\ Y <_{\mathcal{C}_j} B,\ B <_{\mathcal{C}_j} X,\ X <_{\mathcal{C}_j} A\}\ \subseteq <_{\mathcal{C}_j},$$

then the concept match between C_i and C_j becomes: $CM(C_i, \mathcal{O}_i, C_j, \mathcal{O}_j) = \frac{3}{5}$. This discrepancy might increase when comparing two concepts from ontologies belonging to two different agents with different expertise levels.

One way to overcome this is to define similarity based on the compliance of the hierarchical order in which concepts are positioned in the two hierarchies, which, in a way, provides us a scaling of the similarity measure. Based on such a measure, C_i and C_j should have a perfect match. Thus, we define an ordered concept set as follows: an *ordered set* is an n-tuple, denoted by $\{a_1, a_2, \ldots, a_n\}_{\leqslant}$, such that there exists a total order, $\leqslant$, defined on the elements of the set. Based on ordered sets, we can define a new type of mapping, that we call *monotone mapping*.

Definition 3.1 (Monotone mapping)
A mapping $\mathbf{m} : \mathcal{L}_i \to \mathcal{L}_j$, whose domain is the lexicon of $\mathcal{O}_i = \langle \mathcal{C}_i,\ <_{\mathcal{C}_i}\rangle$ and range is the lexicon of $\mathcal{O}_j = \langle \mathcal{C}_j,\ <_{\mathcal{C}_j}\rangle$, is monotone or order-preserving, if for $L_i, L_j \in \mathcal{L}_i$, $\mathcal{F}_i(L_i) \leqslant_{\mathcal{C}_i} \mathcal{F}_i(L_j)$ implies $\mathcal{F}_j(\mathbf{m}(L_i)) \leqslant_{\mathcal{C}_j} \mathcal{F}_j(\mathbf{m}(L_j))$, where $\mathbf{m}(L_i), \mathbf{m}(L_j) \in \mathcal{L}_j$. $\square$

Given: *Two lexical entries* L_i *and* L_j *belonging to ontologies* $\mathcal{O}_i$ *and* $\mathcal{O}_j$ *respectively, find if their concepts match using the thresholds* α_1, α_2, *and* α_3.
if *SM*$(L_i, L_j) \geqslant \alpha_1$ **then**
  if *OCM*$(L_i, \mathcal{O}_i, L_j, \mathcal{O}_j) \geqslant \alpha_2$ **then**
    $\mathbf{m}(L_i) = L_j$
else if *OCM*$(L_i, \mathcal{O}_i, L_j, \mathcal{O}_j) \geqslant \alpha_3$ **then**
  $\mathbf{m}(L_i) = L_j$
else
  $\mathbf{m}(L_i) \neq L_j$

Algorithm 2: Concept mapping

Ordered concept match (*OCM*) is based on order-preserving mappings. $\leqslant_{<_C}$ term in the definition of *OUC* (equation 18) represents the total order based on the taxonomic hierarchy of concepts. Various techniques for representing order in RDF are presented by Melnik and Decker [Melnik and Decker, 2001]. The overlap between two ordered sets is given by the ordering match (*OM*), where $A_{\leqslant_A} = \{a_1, a_2, \ldots, a_n\}_{\leqslant_A}$, $B_{\leqslant_B} = \{b_1, b_2, \ldots, b_n\}_{\leqslant_B}$, and $\mathbf{m}$ is a mapping whose domain is $A_{\leqslant_A}$ and range is $B_{\leqslant_B}$. Simplest such mapping is the lexicographic equivalence function, which can be defined as: $\mathbf{m} = \{(x, y) \mid x \in A_{\leqslant_A},\ y \in B_{\leqslant_B},\ \textit{Lex}(x) = \textit{Lex}(y)\}$ where *Lex*() is a function from a set element to a lexical entity which signify the element. The path comparison technique [Euzenat et al., 2004] is similar in manner since it also compares the labels of objects as well as the sequence of labels of related entities.

4 Social Networks

A referral system is a multi-agent system in which agents cooperate by using referrals where a referral corresponds to a link to another agent stored by the models of agents. A social network refers to a set of agents which socially interact with each other by using queries and answers [Yolum and Singh, 2003]. Agents in our system have a number of policies to learn models of other agents that they interact with. These models store information about their expertise, the projected ability to produce correct answers, and their sociability, the projected ability to produce correct referrals.

The system differentiates between each agent's interests and expertise since these two aspects do not necessarily overlap. This enables us to model the change in each agent's expertise as they develop new interests and update their expertise correspondingly. Each agent poses a query based on its own interests. These queries are first sent to potentially expert agents in the neighborhood of an agent. Agents receiving a query may answer the query based on their confidence in their answer or refer to another agent that is more appropriate. The received answers are used for evaluating the expertise of the answering agent. We represent queries, answers, and interests as sets of $\langle$*term*, *expertiseValue*$\rangle$ tuples when we calculate the similarities between them. Query terms selected form a subset of the concepts chosen from the local ontology of a given agent. The set of queries posed between any two communicating pair of agents forms the conversational context.

Definition 4.1 (**Similarity**)
Given two sets of term-value mappings, a query Q and expertise E, the similarity of Q to E is found as follows:

$$Q \diamond E = \frac{\sum_i q_i \times e_j}{\sqrt{n \sum_{i=1}^{n} q_i^2}},$$

where n is the number of terms in the query, $q_i \in Q$ is a term in Q, and $e_j \in E$ is a term in E such that $\mathbf{m}(q_i) = e_j$. □

Definition 4.1 is similar to the cosine similarity measure that weighs expertise vectors with higher magnitude more. Each agent has an expertise level in a concept term from its ontology, defined in the range $[0, 1]$. Expertise levels are learned dynamically by the social network through query-answer interactions and assessments of the answers. As the interests of agents change, the contents of the questions asked change and progressively, this causes the evolution of the expertise levels and the consensual structure. Thus, the system we have developed can be referred to as a dynamically evolving semantic system based on social interactions.

Agent Communication. When two agents, $\mathcal{A}_i$ and $\mathcal{A}_j$, communicate, they may experience misunderstandings based on the discrepancies in their intended meanings. Given a lexical term L_i from $\mathcal{O}_i$ being used by $\mathcal{A}_i$ to communicate with $\mathcal{A}_j$, we might observe that L_i is not present in $\mathcal{O}_j$. In that case, we need to find the best matching concept from $\mathcal{O}_j$. In another case, two lexical terms L_i and L_j can be syntactically equivalent but conceptually different. We accept that two agents can reach a shared understanding when the lexical terms they use to communicate share the same meaning where the meaning is based on the terms themselves and their corresponding conceptual structures. We resolve these issues by using our concept mapping algorithm (Algorithm 2).

Given: *A set of agents, $\mathbb{A}$, sharing a set of ontologies, $\mathbb{O}$, find the consensus ontology, $\mathcal{O}_C$, represented by a consistent set of statements such that it represents a consensus for the MAS.*

$\mathcal{O}_C = \bigcap_{i=1}^{n} O_{\mathcal{A}_i}$
while $newLeafSetSize \neq LeafSetSize$ **do**
  $LeafSet = getLeaves(\mathcal{O}_C)$
  $LeafSetSize = |LeafSet|$
  for $C_{subj} \in LeafSet$ **do**
    $\mathcal{A}_{expert} = getDomainExpert(\mathbb{O}, C_{subj})$
    $expansionSet = getDomainConceptualization(\mathcal{O}_{\mathcal{A}_{expert}}, C_{subj})$
    for $C_{obj} \in expansionSet$ **do**
      $C'_{obj} = getBestMatchingConcept(\mathbb{O}, C_{obj})$
      if C'_{obj} **then**
        $add(\mathcal{O}_C, C_{subj}, C'_{obj})$
      else
        $add(\mathcal{O}_C, C_{subj}, C_{obj})$
    end
    $newLeafSet = getLeaves(\mathcal{O}_C)$
    $newLeafSetSize = |newLeafSet|$
  end
end

Algorithm 3: Building consensus based on domain expertise

5 Building Consensus Based on Domain Expertise

In this section, we present a consensus building algorithm based on the observation that an agent who is expert in a domain will likely be able to conceptualize the underlying structure better than others.

In Algorithm 3, we first initialize the consensus ontology to the intersection of the component ontologies. This forms the upper ontology model accepted by all agents in the MAS. For each concept in the leaf set, that is the set of concepts that are considered as leaves when the ontology is seen as a tree, we determine the expert agent in that domain. Given the set of agent ontologies from the MAS and a concept, the *getDomainExpert* function returns the agent, $\mathcal{A}_{expert}$, which is the expert in the domain corresponding to the concept. Based on $\mathcal{A}_{expert}$'s conceptualization of the domain, we find an expansion set, *expansionSet*, which contains the set of concepts that are subclasses of the domain. For each concept C_{obj} in the set, we try to find a matching concept from the component ontologies which has a higher expertise level. For a given set of component ontologies and a concept, the *getBestMatchingConcept* function returns the best matching concept, C'_{obj}, from all ontology models which has the best expertise level greater than the expertise level of C_{obj}. If the expertise level of C'_{obj} is not greater than the expertise level of C_{obj}, then this function returns the empty set.

5.1 Randomized Induction Algorithm for Building Consensus

In this section, we present a method based on heuristic search in the space of RDF statement triples for finding the consensus ontology as local agreement among multiple component ontologies. We seek to find the best consensus ontology, $\mathcal{O}_C$, by adding statements to the initial consensus, which is set to $\mathcal{O}^\cap$. To prevent local minima, we use an approach based on randomized algorithms in which we can randomize the statement selection up to a level so that we are allowed to make bad moves.

Our general approach to consensus building is based on simulated annealing [Russell and Norvig, 1995]. In the inner loop, we pick a random statement and check to see if it improves the heuristic value. If it does, we add the statement to our current consensus ontology. Otherwise, with some probability, $p = e^{\frac{\Delta E}{T}}$, we add the statement. p decreases exponentially with the badness of the move, ΔE. Also, the parameter T determines the likelihood of us allowing bad moves. *schedule* determines the value of T based on a function of the number of cycles that has already been completed.

In Algorithm 4, the *HeuristicValue* function is any heuristic measure that estimates the level of overlap based on the given component ontologies. We choose to use the taxonomic overlap measure, $\overline{\overline{TO}}$, which corresponds to the taxonomic overlap among its arguments. This is due to our data set which contains mostly taxonomic relations and due to the fact that taxonomic relations are more important than non-taxonomic ones. *RandomNeighboringStatement*$(\mathcal{O}_C, \mathbb{S})$ is a function which returns a randomly chosen neighboring statement, $S_k \in \mathbb{S}$, of the current consensus ontology such that $\mathcal{O}_C \cup S_k$ is consistent. By neighboring statements to an ontology, we mean the set of statements that can be added to extend a given ontology such that the consistency is preserved.

Given: *A set of ontologies, $\mathbb{O}$, find the consensus ontology represented by a consistent set of statements, $\mathcal{O}_C$, such that it has maximum HeuristicValue.*

$\mathcal{O}_C = \bigcap_{i=1}^{n} \mathcal{O}_i$ (*Initialization*)
$\mathbb{S} = \bigcup_{k=1}^{n} \mathcal{O}_k \; - \; \mathcal{O}_C$
$t = 0$; *(Temperature)*
$e = 0$; *(Energy)*
for *t* $\leftarrow 1$ *to* ∞ **do**
  $T \leftarrow$ *schedule*$[t]$
  if $T = 0$ **then**
    return $\mathcal{O}_C$
  $S_k =$ *RandomNeighboringStatement*$(\mathcal{O}_C, \mathbb{S})$
  $\Delta E =$ *HeuristicValue*$(\mathcal{O}_C \cup S_k, \mathbb{O}) \; - \;$ *HeuristicValue*$(\mathcal{O}_C, \mathbb{O})$
  if $\Delta E > 0$ **then**
    $\mathcal{O}_C = \mathcal{O}_C \cup S_k$
  else
    $\mathcal{O}_C = \mathcal{O}_C \cup S_k$ with probability $e^{\frac{\Delta E}{T}}$
end

Algorithm 4: Building consensus by simulated annealing

6 Experiments and Results

We have experimented with a number of agents ranging from 5 to 1000, having various numbers of differing ontologies ranging from 2 to 53. The expertise levels of agents are initialized to a measure of the depth of the domain within each agent's ontology. The results of our experiments are given in Table 3. By making use of the criterion we introduced in [Biçici, 2006b] and in [Biçici, 2006a], we evaluate a consensus ontology based on how well it agrees with the component ontologies. The evolving nature of the consensus ontology that is generated among 500 agents using 53 different ontologies can be seen in Figs. 3, 4, and 5, which are ordered according to their F-Measure performances. Each figure represents the consensus ontology that is generated at some stage of the evolution.

In our experiments, we attempted to address the variance in the performance of the consensus ontology with respect to the number of agents involved and the number of differing ontologies used. We present our results in Table 3 where *AvgSynSim* and *AvgTaxSym* corresponds to average syntactic and taxonomic sim-

Number of ontologies		Number of agents 5	10	25	50	100	250	500	1000
2	*AvgSynSim*		0.3856		0.3856	0.3856	0.3856	0.3856	0.3856
	AvgTaxSim		0.2890		0.2890	0.2890	0.2890	0.2890	0.2890
	FMeasure		0.5417		0.5417	0.5417	0.5417	0.5417	0.5417
5	*AvgSynSim*	0.1258			0.1249	0.1231	0.1267	0.1267	0.1240
	AvgTaxSim	0.2011			0.1997	0.1970	0.2025	0.2025	0.1984
	FMeasure	0.2433			0.2472	0.2550	0.2393	0.2393	0.2511
10	*AvgSynSim*		0.0710		0.0783	0.0783	0.0759	0.0979	0.0963
	AvgTaxSim		0.1666		0.1678	0.1678	0.1674	0.1962	0.1777
	FMeasure		0.2234		0.1893	0.1893	0.2006	0.1993	0.2384
25	*AvgSynSim*			0.0266	0.0264	0.0265	0.0266	0.0261	0.0262
	AvgTaxSim			0.1278	0.1289	0.1283	0.1278	0.1305	0.1300
	FMeasure			0.1239	0.103	0.1135	0.1239	0.0716	0.0821
53	*AvgSynSim*				0.0162	0.0141	0.0131	0.0144	0.0141
	AvgTaxSim				0.1181	0.1188	0.1164	0.1281	0.1188
	FMeasure				0.0794	0.0884	0.0938	0.0831	0.0884

Table 3: Evaluation results for the consensus built

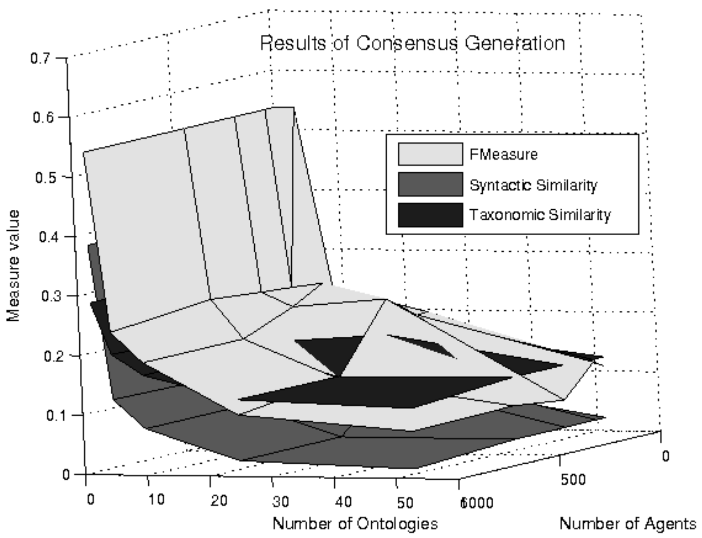

Figure 2: Results plotted in 3D

ilarity scores correspondingly. The resulting graph when the results are plotted in 3D is given in Fig. 2. The results show that the performance increases some as we decrease the number of agents collaborating towards the consensus and it increases greatly as we decrease the number of different ontologies being used by the agents.

We have also experimented with the threshold values used in the similarity measures to find the best setting for building consensus with our system. Under the setting with 50 agents sharing 5 different ontologies, we have found that the α values that are used in our concept mapping algorithm (Algorithm 2) with values of 0.6, 0.3, and 0.5 for α_1, α_2, and α_3 correspondingly gave the best results for the syntactic and taxonomic match measures. F-Measure is maximized when α_1, α_2, and α_3 are set to 0.5, 0.2, and 0.3. We chose to use 0.6, 0.3, and 0.5 for the presented experiments which gave good results overall. All concept mapping algorithms need to balance the weights given for the lexicon, which may be regarded as the pointers to the real concepts, and the weights given for the conceptual structures themselves. The α values represent that if there exists a high lexical match value for a lexical term, then we also check for a level of structural match via ordered conceptual match. But if the lexical match is not at a satisfactory level, then we further require a higher level structural match that could indicate a conceptual mapping possibility.

One research question that needs to be further answered is the existence of plateaus where the consensus ontology might reach after some time and whether there are some phenomena that leads to such plateaus. We have not yet experimented with techniques that can help us identify such regions if they do exist. Consensus plateaus can help us shed new light on the various phenomena that appear in emergent semantics, their mechanisms, and their relationships.

7 Future Work

In the current version of the system, only the consensus ontology is allowed to evolve whereas individual agents' ontologies remain unchanged. Allowing each agent to change its ontology based on queries might be a better alternative for simulations.

The final consensus ontology that is built can be refined based on some heuristics. One such heuristic is the *coherence continuum*. If there is alternation of the expert agent chosen for domains that are consecutively ordered based on the subclass relation, such as in $C_1 \subseteq C_2 \subseteq C_3$, then we may choose to refine the consensus ontology so that all the domains are chosen from the alternating agent's recommendation. For example, if A_1 is the expert for domains C_1 and C_3 and A_2 is the expert for C_2, then to preserve the coherence continuum we can discard the conceptualization of A_2, which can be considered as an interposer.

Another refinement can be done in choosing good domain experts. We can choose to store domain expert histories which can later be used to select experts from when the expertise of the best agent in the current domain is not as good as the agents who are experts in the upper levels of the consensus ontology. This retrospective approach assumes that an expert agent chosen for a given concept term is likely to be good in its subconcepts. However, in the real world, this assumption can easily be challenged. For instance, an expert in programming need not necessarily be good in *LISP* programming itself.

Also, the investigation regarding the existence of consensus plateaus appears promising and postures like a fruitful avenue for the continuance of this research.

8 Conclusion

We have studied the generation, management, and evaluation of consensus ontologies among agents having differing ontologies within the multiagent system framework. The system that we have developed has the capability of modeling the emergence of consensual agreements among socially interacting agents. We have also developed measures for evaluating the consensus ontology based on three different levels of abstraction and heuristic methods for conceptual processing. Interactions between agents based on queries and their assessments allow us to model the quality of resources.

We have provided formal definitions for the problem of finding a consensus ontology in a step by step manner. Conceptual processing methods for building, managing, and evaluating consensus ontologies are given and experimental results are presented. We have presented a method of concept mapping based on the conceptual structures in the ontologies. An algorithm for generating the consensus ontologies using the authoritative agent's conceptualization is presented and another method is developed based on heuristic search in the space of RDF statement triples for finding the consensus ontology as local agreement among multiple component ontologies

The system that we have developed can handle arbitrary ontologies having both taxonomic and non-taxonomic relations. The dynamic emergence of consensus mimics the evolution of language. The resulting system that we have developed is concordant with the principles of emergent semantics. The presented approach looks promising and opens new directions for further research including the investigation of consensus plateaus in systems with the characteristics of emergent semantics. We expect that this research will help us understand and formalize the tradeoffs between approaches to building consensus which can later determine inference mechanisms that can be in place.

Acknowledgements

The research reported here was supported in part by North Carolina State University. The author would like to thank Munindar P. Singh, James Lester, and Jon Doyle for helpful discussions and for their guidance and support during the term of this research.

References

[Aberer et al., 2003] Aberer, K., Cudré-Mauroux, P., and Hauswirth, M. (2003). Start making sense: The chatty web approach for global semantic agreements. *Journal of Web Semantics*, 1(1):89–114.

[Aberer et al., 2004] Aberer, K., Cudré-Mauroux, P., Ouksel, A. M., Catarci, T., Hacid, M.-s., Illarramendi, A., Kashyap, V., Mecella, M., Mena, E., Neuhold, E. J., Troyer, O. D., Risse, T., Scannapieco, M., Saltor, F., Santis, L. D., Spaccapietra, S., Staab, S., and Studer, R. (2004). Emergent semantics principles and issues. In Lee, Y.-J., Li, J., Whang, K.-Y., and Lee, D., editors, *Proceedings of the 9th International Conference on Database Systems for Advanced Applications (DASFAA 2004)*, pages 25–38. Springer.

[Arrow, 1963] Arrow, K. J. (1963). *Social Choice and Individual Values*. John Wiley and Sons, New York, second edition.

[Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic Web. *Scientific American*, 284(5):34–43.

[Biçici, 2006a] Biçici, E. (2006a). Consensus ontology generation in a socially interacting multiagent system. In *COMPSAC '06: Proceedings of the 30th Annual International Computer Software and Applications Conference (COMPSAC'06)*, pages 279–284, Washington, DC, USA. IEEE Computer Society.

[Biçici, 2006b] Biçici, E. (2006b). Generating consensus ontologies among socially interacting agents. In *AMTA '06: International Workshop Agents and Multiagent Systems, from Theory to Application*, Québec City, Québec, Canada.

[Campell and Shapiro, 1998] Campell, A. and Shapiro, S. C. (1998). Algorithms for ontological mediation. In Harabagiu, S., editor, *Use of WordNet in Natural Language Processing Systems: Proceedings of the Conference*, pages 102–107. Association for Computational Linguistics, Somerset, New Jersey.

[Euzenat et al., 2004] Euzenat, J., Bach, T. L., Barrasa, J., Bouquet, P., Bo, J. D., Dieng, R., Ehrig, M., Hauswirth, M., Jarrar, M., Lara, R., Maynard, D., Napoli, A., Stamou, G., Stuckenschmidt, H., Shvaiko, P., Tessaris, S., Acker, S. V., and Zaihrayeu, I. (2004). D2.2.3: State of the art on ontology alignment. Technical report. NoE Knowledge Web project delivable.

[Fellbaum, 1998] Fellbaum, C., editor (1998). *WordNet: An Electronic Lexical Database*. The MIT Press.

[Ganter and Wille, 1999] Ganter, B. and Wille, R. (1999). *Formal Concept Analysis: Mathematical Foundations*. Springer Verlag.

[Garey and Johnson, 1979] Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the theory of of NP-Completeness*. Freeman and Company, San Francisco.

[Hookway, 1992] Hookway, C. (1992). *Peirce*. Routledge, London.

[Lenat et al., 1990] Lenat, D. B., Guha, R. V., Pittman, K., Pratt, D., and Shepherd, M. (1990). Cyc: Toward programs with common sense. *Communications of the ACM, CACM*, 33(8):30–49.

[Li and Horrocks, 2003] Li, L. and Horrocks, I. (2003). A software framework for matchmaking based on semantic web technology. In *Proceedings of the 12th International Conference on World Wide Web*, pages 331–339. ACM Press.

[Maedche, 2002] Maedche, A. (2002). *Ontology Learning for the Semantic Web*. Kluwer Academic Publishers, Boston.

[Melnik and Decker, 2001] Melnik, S. and Decker, S. (2001). Representing order in RDF.

[Noy, 2004] Noy, N. F. (2004). Semantic integration: A survey of ontology-based approaches. *SIGMOD Record*, 33(4):65–70.

[Quine and Ullian, 1978] Quine, W. V. and Ullian, J. S. (1978). *The Web of Belief*. McGraw-Hill, New York, NY, second edition.

[Quine, 1970] Quine, W. V. O. (1970). On the reasons for indeterminacy of translation. *Journal of Philosophy*, 67(6):178–183.

[Quine, 1986] Quine, W. V. O. (1986). *Philosophy of Logic*. Harvard University Press, second edition.

[Quine, 1995] Quine, W. V. O. (1995). *From Stimulus To Science*. Harvard University Press.

[Russell and Norvig, 1995] Russell, S. and Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

[Scott, 1986] Scott, D. S. (1986). Capturing concepts with data structures. In *Proc.DS-2, IFIP TC-2 Conference on Knowledge and Data, Portugal*.

[Stephens and Huhns, 2001] Stephens, L. M. and Huhns, M. N. (2001). Consensus ontologies: Reconciling the semantics of web pages and agents. *IEEE Internet Computing*, 5(5):92–95.

[Stumme, 2002] Stumme, G. (2002). Formal concept analysis on its way from mathematics to computer science. *Lecture Notes in Computer Science*, 2393:2–19.

[Stumme and Maedche, 2001] Stumme, G. and Maedche, A. (2001). FCA-MERGE: Bottom-Up merging of ontologies. In Nebel, B., editor, *Proceedings of the seventeenth International Conference on Artificial Intelligence (IJCAI-01)*, pages 225–234, San Francisco, CA. Morgan Kaufmann Publishers, Inc.

[van Diggelen et al., 2004] van Diggelen, J., Beun, R. J., Dignum, F., van Eijk, R. M., and Meyer, J.-J. (2004). Optimal communication vocabularies in the presence of heterogeneous ontologies. Utrecht, Netherland.

[Wiederhold and Mitra, 2001] Wiederhold, G. and Mitra, P. (2001). An algebra for semantic interoperability of information sources. In *IEEE Symp. on BioInformatics and Bioengineering*, pages 174–182.

[Williams et al., 2003] Williams, A. B., Padmanabhan, A., and Blake, M. B. (2003). Local consensus ontologies for b2b-oriented service composition. In *AAMAS*, pages 647–654.

[World Wide Web Consortium (W3C), 2004] World Wide Web Consortium (W3C) (2004). Resource description framework (RDF). URL: http://www.w3.org/RDF/.

[Yolum and Singh, 2003] Yolum, P. and Singh, M. P. (2003). Dynamic communities in referral networks. *Web Intelligence and Agent Systems*, 1(2):105–116.

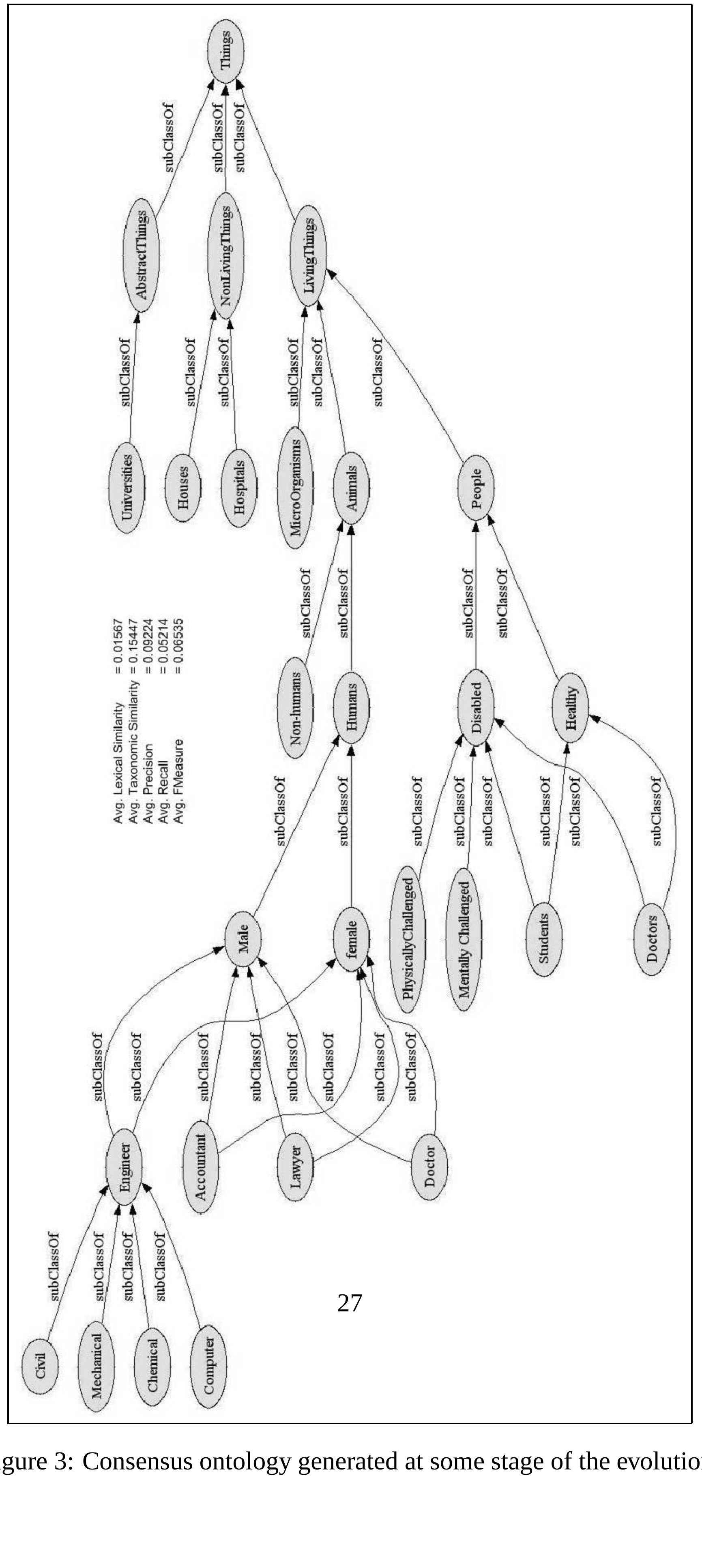

Figure 3: Consensus ontology generated at some stage of the evolution.

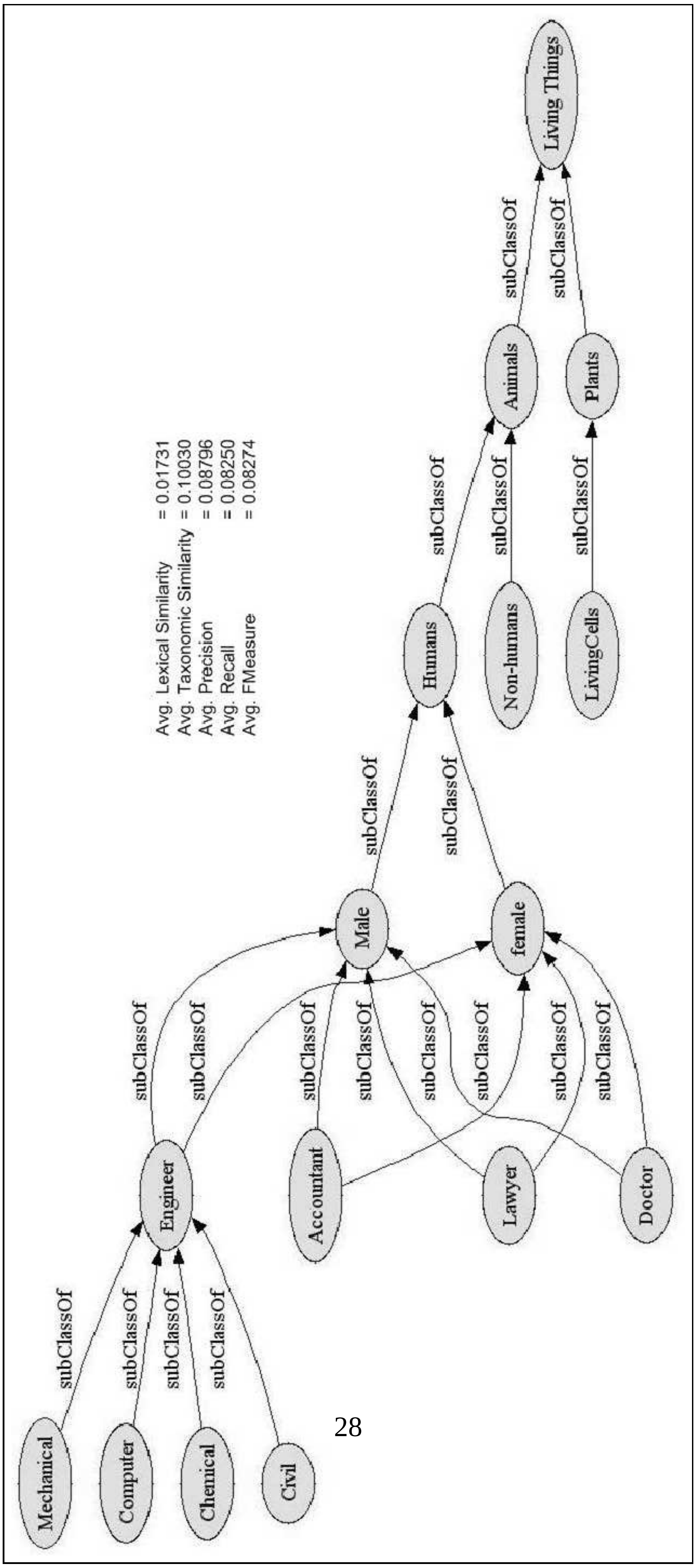

Figure 4: Consensus ontology generated at another stage of the evolution.

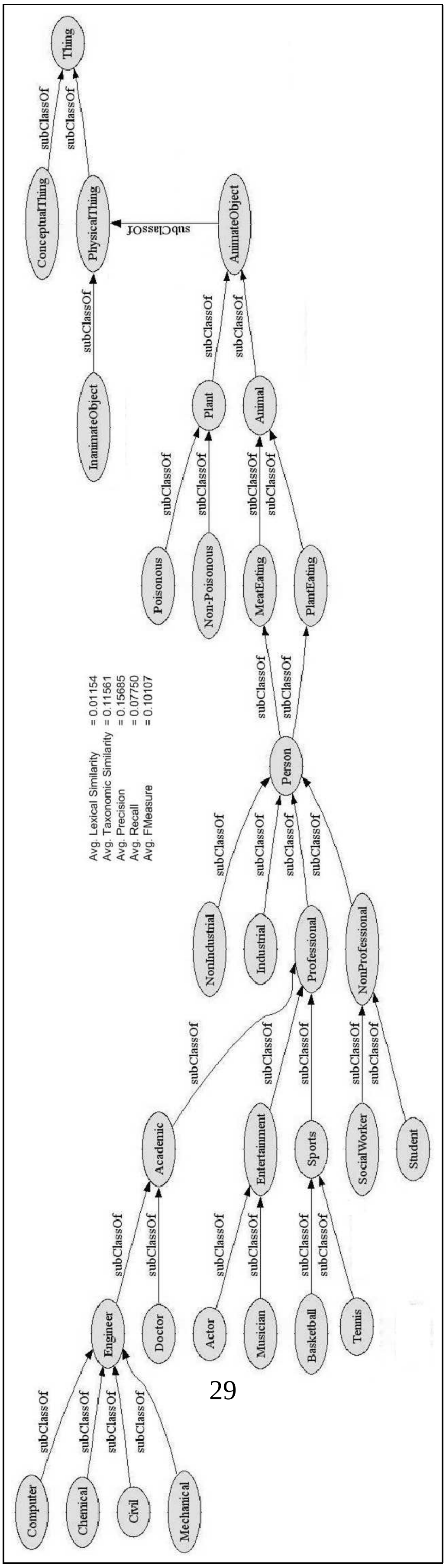

Figure 5: Consensus ontology generated at another stage of the evolution.