An Ontological Approach for Querying Distributed Heterogeneous Information Systems in Critical Operational Environments

Atif Khan, John A. Doucette, and Robin Cohen

David R. Cheriton School of Computer Science, University of Waterloo
{atif.khan,j3doucet,rcohen}@uwaterloo.ca

Abstract. In this paper, we propose a decision making framework suited for knowledge and time constrained operational environments. We draw our motivation from the observation that large knowledge repositories are distributed over heterogeneous information management systems. This makes it difficult for a user to aggregate and process all relevant information to make the best decision possible. Our proposed framework eliminates the need for local aggregation of distributed information by allowing the user to ask meaningful questions. We utilize semantic knowledge representation to share information and semantic reasoning to answer user queries. We look at an emergency healthcare scenario to demonstrate the feasibility of our approach. The framework is contrasted with conventional machine learning techniques and with existing work in semantic question answering. We also discuss theoretical and practical advantages over conventional techniques.

1 Introduction

As electronic information systems become mainstream, society's dependence upon them for knowledge acquisition has increased. Over the years, the sophistication of these systems has evolved, making them capable of not only storing large amounts of information in diverse formats, but also of reasoning about complex decisions. The increase in technological capabilities has revolutionized the syntactic interoperability of modern information systems, allowing for a heterogeneous mix of systems to exchange a wide spectrum of data in many different formats. The successful exchange of raw information is, however, only the first step towards solving the bigger *semantic challenge* of information exchange. This is analogous to the "ontology challenge" defined by [15].

In recent years a focused effort in the semantic web domain has resulted in technological advancements, providing sophisticated tools for intelligent knowledge representation, information processing and reasoning. Domain specific knowledge can be managed by utilizing a diverse set of ontological solutions, which capture key domain concepts and the relationships between them. Knowledge regarding the domain can then be shared by publishing information in a domain specific ontology. A semantic reasoning engine can then be applied to a knowledge-base to answer complex user queries. The semantic reasoning process allows for enhanced knowledge discovery that may not be possible via consumption of the raw data alone. Latent relationships can be discovered by applying inference rules to the ontological knowledge-base.

Although the premise of the semantic web technology is sound in principle and the use of an ontology can significantly enhance how users consume and process information, practical implementation all but demands that the distributed heterogeneous knowledge be represented by local ontological representations [13]. Consequently, it is still difficult to share knowledge across diverse heterogeneous sources to answer specific questions. Furthermore, under adverse conditions (i.e. constraints on time, communication and/or knowledge), the usefulness of the aggregated data decreases sharply, since human agents are required to (manually) process and reason with the data.

For example, in a health care setting, a physician may need to consult various medical information systems in order to determine the best possible solution for a patient. Given ideal conditions, a physician will be able acquire and process information from various systems and make the ideal diagnosis. If the same scenario is now constrained by the available time, communication bandwidth and the skill level of the physician, the same quality of medical care may not be possible.

Motivated by this, we propose a framework where a user will pose questions directly (in natural language), rather than aggregate knowledge locally in an attempt to find the answer. The framework will

- Process the user query.
- Aggregate information from various sources.
- Create a semantic representation of the aggregated data.
- Process information using a semantic reasoner.

Each answer generated (in response to the user query) is backed up by a semantic proof. The semantic proof has the desirable property that it can be independently validated by any third party. Our approach does not require the exchange of large data-sets to make a decision, and consequently is more suitable for the above mentioned adverse scenarios.

2 Proposed Solution

We propose a framework for reliable information exchange between distributed heterogeneous parties, using semantic web technologies under constrained operational conditions. We observe that under normal circumstances, such an exchange can easily be accomplished using existing techniques. These techniques fail to be of practical use under adverse situations. For example, consider the following time and information constrained setting: A patient is in a critical life threatening situation, and is being treated by an emergency response (EMR)

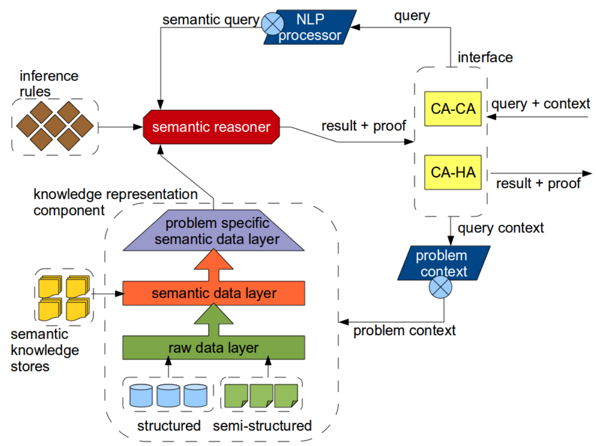

Fig. 1. System architecture

team member. Under these conditions the EMR team member may not be able to provide the best personalized care, because of difficulty accessing patient medical records in a timely manner or correctly interpreting those records.

Our proposed framework builds on top of the semantic web technologies. We use ontological models for knowledge representation. We acknowledge the fact that diverse heterogeneous information will be represented by an array of local or domain ontologies. Therefore, our framework provides support for working with multiple data-sets represented by different ontological models. Given the almost infinite amount of information in the world, we utilize a problem context to identify and limit the amount of knowledge that needs to be processed. We create a problem specific information model using this context. We utilize a semantic reasoner that takes as its input a knowledge-base, a set of inference rules and a user query. The reasoner generates a two part result-set, where the first element is the answer to the provided user query and the second element is a semantic proof.

We will now discuss the details of the various components of our proposed framework along with some examples.

2.1 System Architecture

We present a flexible architectural style for our proposed framework. Previous approaches utilizing similair frameworks tend to be domain specific (e.g. [17]). In contrast, our approach is domain indipendent. We now illustrate the salient components of our design (Refer to Fig 1).

System Interface The system interface component facilitates interaction by allowing a user to pose a query to the system. The user may also provide a query specific context. We provide support for two types of user communications based on the following two user classifications (i) a computational agent (CA) – represents an artificially intelligent automated system and (ii) a human agent (HA) – representing a human being. The first type of agent communication is between two CAs. A local CA receives a query (and a context) from a remote CA. This type of communication represents distributed automated systems interacting with each other. The second type of communication utilizes a local CA and a remote HA. This allows human beings to pose queries to a local system. For each query, the interface receives a response from the reasoning module, and forwards this response to the remote user.

The system interface component provides a queryable abstraction around the heterogeneous knowledge stores, so that the actual data (utilized for answering the query) does not have to be transmitted. This characteristic of the framework facilitates knowledge sharing under adverse conditions.

Knowledge-Representation The knowledge-representation component of our framework follows a multi-tiered design that is capable of accepting data from a wide array of heterogeneous sources. It also utilizes the problem-context (generated from the user query context) to limit the amount of data which must be processed to answer the query.

The raw data layer provides a useful abstraction to deal with all non-semantic data sources. These data-sources are composed of structured data (such as in the case of distributed relational database systems) and semi-structured data (such as content repositories and web pages). We assume that this raw data does not have any semantic capabilities built into it.

Information from the raw data layer is then annotated using appropriate ontologies. This semantic data layer provides the appropriate abstraction. It is important to note that we do not constrain the choice of the ontologies used. The main goal here is to be able to convert raw data into its semantically equivalent representation. The semantic data layer is also capable of incorporating data from other semantic data repositories.

The problem-specific semantic layer provides a normalization of the semantic data layer. The main goal of this layer is to provide mappings between various ontological representations of the data in use. For example a single semantic concept (such as *name*) that may be defined by different ontologies can be normalized and represented by a concept from a single consistent ontology.

Reasoning and Inference The reasoning layer is responsible for processing the various inputs from other modules such as the semantic query (representing the initial user query), the inference rules, and the knowledge-base from the problem-specific semantic layer. It utilizes a semantic reasoner [27] to reason about the user query over the selected knowledge-base. The reasoner generates a two part result-set. The first element of the result-set contains the answer to the user query. The second element contains a semantic proof in support of the response.

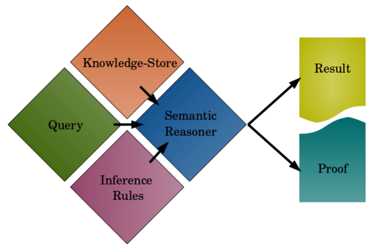

Fig. 2. Semantic Reasoning and inference

A semantic proof has the desirable property that it can be validated by any party. In a heterogeneous multi-agent distributed environment, knowledge changes with time. Therefore, the same query may not result in the same answer at a different time. Having a semantic proof generated for each user query allows the validation an answer against the knowledge-base representation (that was aggregated by the problem-specific semantic layer) at any given instant in time by any party.

Motivation In this section we consider two simple scenarios for knowledge sharing under adverse conditions, constrained by lack of time and lack of knowledge. The purpose of these examples is to highlight the various components of our proposed architecture and their interactions with one other. Fig 3 depicts a semantic model capturing the high level entities for a medical scenario. This semantic model represents the normalized view of the information gathered from various distributed sources. The model describes not only the entities, but also the semantic relationships between these entities.

The main entities defined in our model are patients, health care providers, drugs, diseases and various medical conditions. For the sake of simplicity, we define various simple relationships between these entities. The main relationship is the IS_A relationship (sometimes called "subsumption"). For example a doctor IS_A health care provider which IS_A person. Similarly Insulin IS_A allopathic drug which IS_A drug. In addition to the IS_A relationship, we also define several other varieties of attribute-value relationship. For example the disease Ulcer has a *condition* called Bleeding, the drug Nitroglycerin has a *contraindication* to the drug Viagra (Fig. 3). Using the triple notation [22] we capture the semantic model in a triple-store.

Example Scenario Consider a hypothetical scenario where an emergency response team member would like to administer Warfrain (an anticoagulant drug) to Alice in order to treat her for potential blood clotting. Alice is currently early in her pregnancy. The EMR member has had no past interactions with Alice, and is not aware of her medical condition and history. We add the following two constraints to this scenario to incorporate the (adverse) time and knowledge factors.

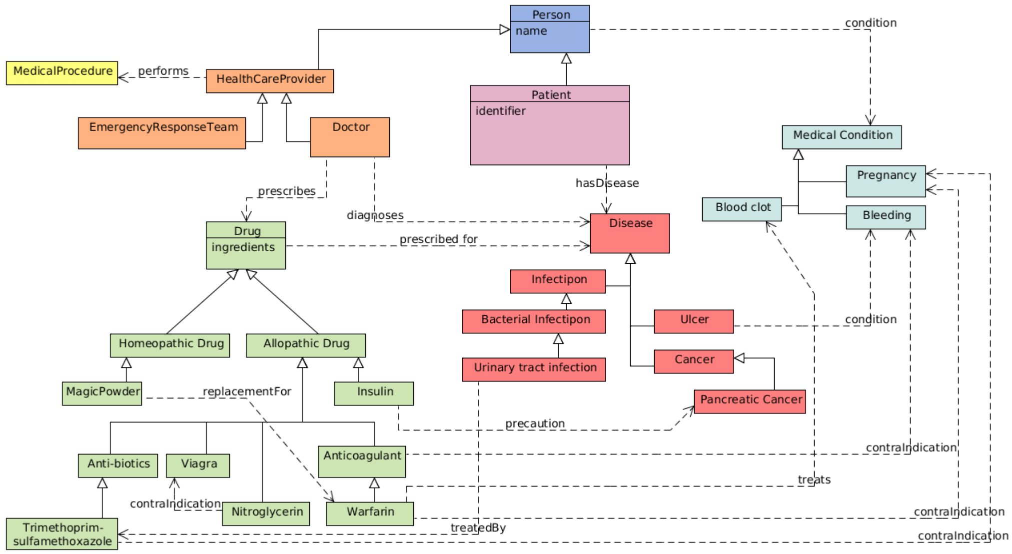

Fig. 3. Semantic model for medical question answering

– The current conditions prevent the EMR person from accessing and reviewing Alice's medical records.
– Alice's blood clot condition needs to be treated urgently.

Instead of aggregating information related to this scenario (such as Alice's medical records, drug interaction guidelines and such), the EMR person would launch a natural language query such as "can Alice be given Warfrain?" against a medical information system based on our framework. The system would identify Alice and Warfrain, and would compile the required information from various heterogeneous sources. The compiled knowledge is then translated into its' semantic representation. Fig. 4 shows a simplified contextual model based on the global knowledge store presented in Fig. 3. The semantic reasoner will consume this information along with the rules and semantic (user) query, and will generate a result and a proof as follows:

User Query

:Alice *:canNotBeGiven* :Warfrain.

Inference Rule

{?PATIENT :condition ?CONDITION. ?DRUG :contraIndication ?CONDITION. } => {?PATIENT *:canNotBeGiven* ?DRUG}.

Semantic Reasoning & Proof

{{:Alice :condition :Pregnancy} ***e:evidence*** <knowledge-base#_27>.
{:Warfrain :contraIndication :Pregnancy} ***e:evidence*** <knowledge-base#_22>}
=>
{ {:Alice :canNotBeGiven :Warfrain} ***e:evidence*** <rules#_9>}.
Proof found in 3 steps (2970 steps/sec) using 1 engine (18 triples) }.

Based on the facts and the inference rules, the semantic reasoner concludes that Alice can not be given Warfrain since she is pregnant and the Warfrain has a *contraindication* relationship with Pregnancy. The N3 representation of the user query, inference rules and semantic proof are shown above.

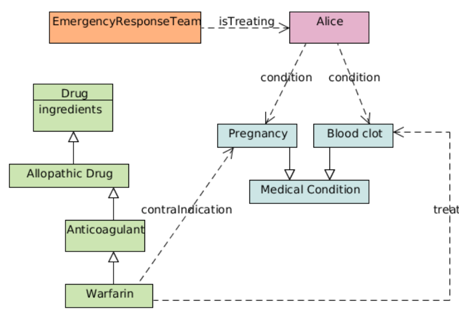

Fig. 4. Example Scenario: Should Alice be given Warfrain?

The scenario discussed above has been kept simple for ease of understanding. A more realistic knowledge-base would be quite rich in semantic concepts and a large number of relationships between the concepts. Similarly there will be a larger array of rules defined to provide the required level of inferencing capabilities to a complex semantic model.

3 Framework Realization

It is important to note that we are proposing a framework that can have many different realizations based on given system requirements. For example certain implementations can omit the natural language query interface if the interacting components are artificially intelligent machine agents. Similarly, different semantic reasoners can be used to achieve implementation goals of performance. Our proposed framework identifies the critical system components and their interactions.

Our realization of the system was solely focused on validating the proposed framework. As semantic knowledge representation and reasoning represent the most important components of the proposed framework, our proof-of-concept realization was focused around the workings and validation of these components. We used N3 [22] notation to represent all knowledge (raw facts), inference rules and the system queries. Considering that N3 utilizes triple format to represent knowledge, any other representation capable of using the triple notation would be compatible with our approach.

We utilized the Euler proof mechanism [27] for semantic reasoning in our realization. Our choice was mainly driven by Euler's support for N3 notation and its support for the Java programming language (since our application was written in Java). The Euler project also provided an extensive set of examples where the OWL rules and concepts were already translated into N3 representation.

4 Related Work

In this section we establish both theoretical and practical concerns motivating the use of ontologies in question answering, and discuss previous work incorporating ontologies into knowledge querying.

4.1 Ontology-Free Approaches to Querying

There is considerable recent work suggesting that conventional querying techniques, though extremely powerful, might not be suitable for use in environments where queries are frequent, time-dependent, and arbitrarily complex. The principal reason for this is that, in the absence of semantic reasoning and inference rules, all information available for querying must be available in explicit form. This poses a problem in domains where there exists an enormous amount of information, precluding of the possibility explicit codification. For example, in the medical domain, there are hundreds of thousands of codified relationships between various concepts [28], but the hierarchical nature of these relationships means that the number of *implicit* relationships can be much higher. For example, viral pneumonia is explicitly defined as a type infectious pneumonia, but implicitly it is also a type of lung disease.

This motivates the use of machine learning techniques as a possible method of answering ad-hoc queries to an information system which may not encode all possible relationships. By taking a sufficiently large sample of the data, it may be possible to infer the answer to a user's query. For example, if a user asks whether a particular patient can be given a drug, a predictive classification system could be dynamically constructed and utilized to answer the query.

4.2 Motivations for Ontological Approaches to Querying

There are both theoretical and practical motivations for avoiding the use of traditional machine learning techniques to answer the kind of questions described above. Many machine learning algorithms, including popular decision trees (C4.5, ID3 [23]), maximum margin classifiers (e.g. Support Vector Machines

[7]), and clustering techniques (e.g. KNN [9]), operate by phrasing queries as optimization problems. For example, if a doctor wants to know whether their patient is likely to experience an adverse reaction to a drug, then a system might collect a large sample of patient records and use them to build a classification model. Although machine learning algorithms are often very effective in practice, there are theoretical reasons to suppose they might be less useful in time-critical domains where arbitrary queries are being made. "No Free Lunch" theorem (NFL) [30] shows that all optimization techniques are expected to produce identical mean performance across a set of arbitrary queries, in the absence of domain specific knowledge. This suggests that, over a large set of possible queries, no conventional machine learning technique is likely to answer all queries better than using completely random optimization strategies. In critical scenarios like ours, the possibility of receiving a poor result might be too large a risk for users to trust the system's answers.

There are also practical considerations, especially the opacity of the answers obtained using conventional query techniques. Continuing with our example above, what the doctor receives in response to a query about adverse reactions to a drug is a classification model based on a sample of patient data. The understandability of these models to computational laypeople varies from model to model. A support vector machine for example, is practically impossible for a layperson to understand, since it operates by building the maximally separating hyper-plane for a high-dimensional extrapolation of the given data. When the doctor asks "Why does the system believe my patient will have an adverse reaction?", she may not trust a system which answers "I put your patient's record into a 500 dimensional space, and it fell on this side of a line". This is true even if the system is highly reliable, because human users may have concerns about the ethics of entrusting life-saving decisions to a "black box". The system cannot easily explain its decision in terms of medical conditions and the relationships between them, and so it is impossible to tell whether the answer provided is based on sound reasoning, or an unfortunate hiccup in the algorithm's usual consistency.

4.3 Previous Ontological Approaches to Querying

There has been considerable previous work utilizing ontologies for answering queries, but the general focus is on preprocessing of queries to facilitate the use of conventional machine learning techniques. This is a reasonable approach insofar as it obviates the NFL issues described above by introducing domain specific knowledge into the optimization process. In medicine, for example, there has been a focus on isolating the queries used by doctors most frequently, and preprocessing them using semantic information[8,12]. By utilizing ontological information, previous researchers have created frameworks capable of automated contextualization of doctor queries. For example, a doctor whose patient has type I diabetes would have queries regarding that patient and "diabetes" automatically translated to instead include "Type I Diabetes" [21]. An alternative approach considers the incorporation of meta-data into search queries, which can be utilized to return more relevant documents during information retrieval [11]. Finally, recent research in question answering systems utilizes ontologies to translate doctors' questions into lists of relevant terms for an ordinary search engine [14].

The use of a semantic reasoner in place of a conventional machine learning algorithm to answer search queries offers several immediate advantages. First, because a semantic reasoner does not rely on optimization to construct a predictive model, it is not subject to the problems posed by the No Free Lunch theorems for optimization. This eliminates the need for extensive incorporation of a priori knowledge by the end user, as in [11]. Second, the opacity problem is solved by the ability of the framework to both provide a proof of its answer (i.e. the chain of reasoning used to determine the answer), and to formulate that proof in terms easily understood by a layperson (i.e. via conversion of triple formated data into simple natural language statements). For example, if our doctor wishes to ask "Why does the system believe my patient will have a reaction to this drug?", instead of being told, somewhat tautologically, that their patient fits the system's model of patients who had reactions, the doctor can be provided with a patient-specific proof based on medical evidence. By providing a semantic proof, the framework asserts that answer to a user's query is correct, based on the present data.

5 Future Work

Future work will take two directions. First, we plan to implement and benchmark a prototype system, and compare its performance with that of a system based around conventional machine learning techniques for question answering. Second, we plan to extend the framework by overlaying probabilistic models onto the ontological model, to provide a more precise answer to a users' queries. For example, a user who reports cracks in a bridge might be told that there is a 60% chance of bridge failure, rather than simply being told that the bridge *will* collapse if they drive over it. A drawback associated with this extension is the "curse of dimensionality" which arises when there are many possible *combinations* of factors that have different interactions. For example, a bent bridge might have a 30% chance of collapsing, but a bent and cracked bridge a 99% chance of collapsing. The problem worsens as additional factors are added, and each combination of factors in turn must be considered.

To avoid this problem, we plan to consider the introduction of heuristic techniques for providing estimated probabilities. For example, we might have the system take a random sampling of past bridges with both characteristics, and produce an observed probability estimate. Alternatively, the framework could provide "reasonable" bounds in the absence of additional information by assuming no interaction and a positive interaction of strength proportionate to the criticality of the task. Thus, if the bridge is only 3ft off the ground, estimates of the risk would tend to be more liberal (i.e. smaller interaction estimates) than if the bridge is 300ft off the ground. Neither scheme is ideal, and experimental validation might be required to determine appropriate estimates of risk.

6 Conclusion

In this paper we present a proposal for a general purpose ontology-based information exchange framework, intended for use in time critical, knowledge sparse scenarios. The framework utilizes ontologies to retrieve contextually relevant facts from external data sources; reason about those facts in the context of a problem-dependent rule base; and produce both answers and human readable proofs relevant to user queries.

The framework is demonstrated through two example scenarios with a prototype, and contrasted with existing work on semantic data mining (which tends to focus on pre- and post-processing, rather than rule discovery and query answering), and conventional, non-semantic machine learning approaches. Our framework eliminates the problems posed by the No Free Lunch theorem for optimization [30], and provides transparent answers which are easily understood by computational laypersons. Future work will focus on the implementation of a fully functional system, user studies of the system's effectiveness as compared with conventional techniques, and on incorporating probabilistic reasoning into the model.

Bibliography

1. Bettina Berendt, Andreas Hotho, and Gerd Stumme. Towards semantic web mining. In *In International Semantic Web Conference (ISWC*, pages 264–278. Springer, 2002.
2. Tim Berners-lee, Dan Connolly, Lalana Kagal, Yosi Scharf, and Jim Hendler. N3logic: A logical framework for the world wide web. *Theory and Practice of Logic Programming*, 8(03):249–269, 2008.
3. Abraham Bernstein, Foster Provost, and Shawndra Hill. Toward intelligent assistance for a data mining process: An ontology-based approach for cost-sensitive classification. *IEEE Transactions on Knowledge and Data Engineering*, 17:503–518, 2005.
4. Laurent Brisson and Martine Collard. How to semantically enhance a data mining process? In Will Aalst, John Mylopoulos, Norman M. Sadeh, Michael J. Shaw, Clemens Szyperski, Joaquim Filipe, and Jose Cordeiro, editors, *Enterprise Information Systems*, volume 19 of *Lecture Notes in Business Information Processing*, pages 103–116. Springer Berlin Heidelberg, 2009.
5. Silvana Castano and Alfio Ferrara. Knowledge representation and transformation in ontology-based data integration. In *Knowledge Transformation for the Semantic Web*, pages 107–121. 2003.

6. Harry Chen, Filip Perich, Dipanjan Chakraborty, Tim Finin, and Anupam Joshi. Intelligent agents meet semantic web in a smart meeting room. *Autonomous Agents and Multiagent Systems, International Joint Conference on*, 2:854–861, 2004.
7. Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.
8. Manning PR Covell DG, Uman GC. Information needs in office practice: are they being met? *Annals of Internal Medicine*, 103:596–9, 1985.
9. T. M. Cover and P. E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory J*, 13:21–27, 1967.
10. Randy D. Smith David W. Embley, Douglas M. Campbell. Ontology-based extraction and structuring of information from data-rich unstructrued documents. In *Proceedings of the 1998 ACM International Conference on Information and Knowledge Management, Bethesda, Maryland, USA*, 1998.
11. Dina Demner-Fushman and Jimmy Lin. Answering clinical questions with knowledge-based and statistical techniques. *Computational Linguistics*, 2007.
12. JW Ely, JA Osheroff, MH Ebell, GR Bergus, BT Levy, ML Chambliss, and ER Evans. Analysis of questions asked by family doctors regarding patient care. *British Medical Journal*, 319:358–361, 1999.
13. James Hendler. Agents and the semantic web. *IEEE Intelligent Systems*, 16:30–37, 2001.
14. Pierre Jacquemart and Pierre Zweigenbaum. Knowledge and reasoning for medical question-answering. In *Proceedings of the 2009 Workshop on Knowledge and Reasoning for Answering Questions*, 2009.
15. Nicholas R. Jennings, Katia Sycara, and Michael Wooldridge. A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1:7–38, January 1998.
16. Igor Jurisica, John Mylopoulos, and Eric Yu. Using ontologies for knowledge management: An information systems perspective. In *In Proceedings of the 62nd Annual Meeting of the American Society for Information Science (ASISÅ 99), Oct. 31 - Nov*, pages 482–496, 1999.
17. Pavandeep Kataria and Radmila Juric. Sharing e-health information through ontological layering. In *Proceedings of the 2010 43rd Hawaii International Conference on System Sciences*, 2010.
18. Haishan Liu. Towards semantic data mining. In *9th International Semantic Web Conference (ISWC2010)*, November 2010.
19. Alexander Maedche, Boris Motik, Nuno Silva, and Raphael Volz. Mafra - a mapping framework for distributed ontologies. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web*, EKAW '02, pages 235–250, London, UK, 2002. Springer-Verlag.
20. Alexander Maedche and Steffen Staab. Ontology learning for the semantic web. *IEEE Intelligent Systems*, 16:72–79, 2001.
21. Eneida A. Mendonca, James J. Cimino, Stephen B. Johnson, and Yoon-Ho Seol. Accessing heterogeneous sources of evidence to answer clinical questions. *Journal of Biomedical Informatics*, 34:85–98, 2001.
22. Notation 3 (n3): A readable RDF syntax. http://www.w3.org/DesignIssues/Notation3.
23. JR Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers, 1993.
24. Erhard Rahm and Philip A. Bernstein. A survey of approaches to automatic schema matching. *The VLDB Journal*, 10:334–350, 2001. 10.1007/s007780100057.

25. Sarunas Raudys and Alvydas Pumputis. Group interests of agents functioning in changing environments. In *Multi-Agent Systems and Applications IV*, volume 3690 of *Lecture Notes in Computer Science*, pages 559–563. Springer Berlin / Heidelberg, 2005.
26. Å arunas Raudys. Survival of intelligent agents in changing environments. In *Artificial Intelligence and Soft Computing - ICAISC 2004*, volume 3070 of *Lecture Notes in Computer Science*, pages 109–117. Springer Berlin / Heidelberg, 2004.
27. Jos De Roo. Euler proof mechanism. Website.
28. Kent Spackman. Snomed-ct overview nehta presentation. http://www.nehta.gov.au/component/docman/doc_details/589-snomed-ct-overview-nehta-presentation-august-2008.
29. Andrew B. Williams. Learning to share meaning in a multi-agent system. *Autonomous Agents and Multi-Agent Systems*, 8:165–193, March 2004.
30. D.H. Wolpert and W.G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1997.
31. D.H. Wolpert and W.G. Macready. Coevolutionary free lunches. *IEEE Transactions on Evolutionary Computation*, 9(6):721 – 735, 2005.
32. Youyong Zou, Tim Finin, Li Ding, Harry Chen, and Rong Pan. Using Semantic web technology in Multi-Agent systems: a case study in the TAGA Trading agent environment. In *Proceeding of the 5th International Conference on Electronic Commerce*, pages 95–101, September 2003.