expert systems – confrontation – ontology alignment

Samuel GESCHE[*]
Guy CAPLAT[**]
Sylvie CALABRETTO[***]

MANAGING AN EXPERT SYSTEM AMONG OTHERS

Expert systems exist for several decades now and have been provided with many tools, for almost any step of their development. However, if all these methods allow for an effective building of such systems, because knowledge itself is still to be acquired, they cannot act on it. Only experts of the domain are thus able to go beyond the process of designing an expert system and to actually evaluating it using other expert systems. We present here a possibility, using the ontology alignment technologies, to confront several expert systems based on heavy ontologies. Given the problem raised by using more than an order of logics at once, we propose a human-driven approach and we describe our model for an environment dedicated to the confrontation of experts systems.

1. INTRODUCTION

Sometimes there is a need for the confrontation of several models. Differences of opinions are often at the base of this need, but other causes can be found as well. For example, let us suppose that there are two systems, based on two different models, performing the same task. Each of these systems is better than the other in some part of the process, and less good in other part. It raises, intuitively, the wish for a means to get the best of each. One of the solutions is to confront the underlying models and to find what the differences between them are –and which differences can lead to improving any of them.

The world of expert systems has much evolved in the last decades. They have been provided with many tools, for almost any step of their development. From methods of design to generic libraries and design patterns, all is made to grant the design and development process for knowledge-based systems with a good level of reliability. In the beginning of the art, systems were created in extenso from elicitated knowledge and observed practices of an expert. Nowadays, ontologies and methods such as CommonKADS [1] allow exploiting generic knowledge in knowledge modelling. Expert systems are thus a place where knowledge specific to an expert coexist along with capitalized and shared community knowledge.

---

[*] LIRIS, Insa de Lyon, 69621 Villeurbanne Cedex, France – samuel.gesche@insa-lyon.fr
[**] Insa de Lyon, 69621 Villeurbanne Cedex, France – guy.caplat@insa-lyon.fr
[***] LIRIS, Insa de Lyon, 69621 Villeurbanne Cedex, France – sylvie.calabretto@insa-lyon.fr

However, if all these methods allow for an effective building of such systems, because knowledge itself is still to be acquired, they cannot act on it. In other words, the principle of an expert system is to rely on the knowledge of an expert –and only the expert, or another expert on the same field, is able to judge which knowledge has to be used. While this does not raise so much a problem when considering the expert system in itself, when put on the market with other systems informed by other experts, it raises at least the issue of positioning, and with methods that have a good granularity, it can even lead to improving a system that has weak points. This is the reason why we think that performing the confrontation of several systems can be beneficial.

We focus on expert systems built on the model of heavy ontologies, like those specified in the CommonKADS method. And more specifically, we focus on forward and monotonic reasoning engines. Heavy ontologies include an ontology containing the model of the domain (the axioms), and a set of constraints on this ontology, including the rules to be applied by the inference engine.

We begin this paper by presenting two general approaches to expert systems confrontation, along with their respective weaknesses. Then, after having presented the concept of heavy ontology and the way it can be implemented in expert systems, we talk about ontology alignment and its application to expert systems confrontation. Finally, we describe a model for an environment dedicated to this confrontation.

2. EXPERT SYSTEMS COMPARISON

When it comes to comparing two experts systems, two methods can be chosen:

- Either the results of these systems are compared a posteriori, and with a sufficient set of benchmarks one hope to find inconsistencies or differences between the systems;
- Or the systems themselves are tested a priori by another expert system of higher order, in order to find differences in the rules or other data.

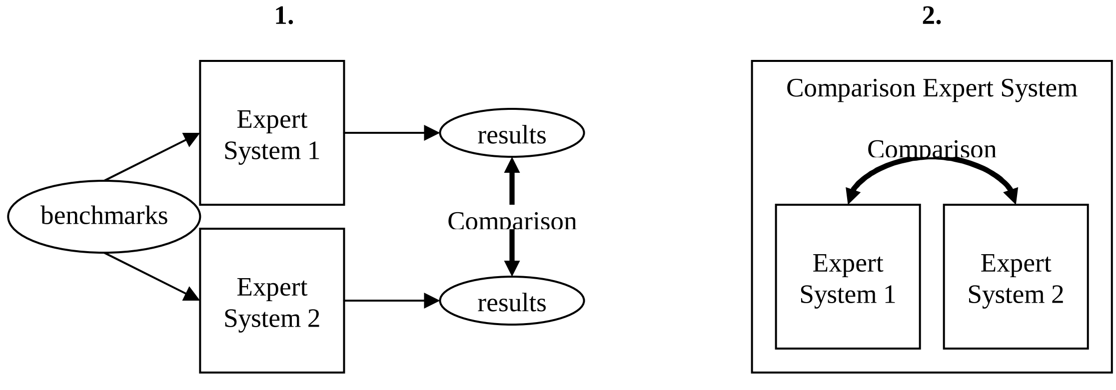

Fig. 1. Two approaches to expert systems confrontation

The main issue with the first approach is that it cannot be proven that the benchmarks are sufficient for uncovering all inconsistencies between the systems. The second approach guarantees exhaustiveness, but at the cost of having to create another system, which itself can benefit from a test including comparison with other systems of this type. And, because successive systems have increasing logical order, it is impossible to design a system for generic comparison. The machine cannot work on more than one order without facing ambiguity issues, as demonstrated in Russell’s type theory [4].

3. EXPERT SYSTEMS AND ONTOLOGIES

Basically, experts systems involve four concepts –facts, domain model, rules and strategies. Facts are the data given as input and returned as output. The whole purpose of the expert system is to use given facts to infere others. Domain model and rules are used by the inference engine to deduce new facts from those that have been previously given or deduced (that is, in a forward reasoning process). Strategies act as higher-order rules, and are used to choose which rule have to be used. Strategies can be either hardcoded or, using successive layers of their inference engine, strategies can be given as rules of higher order. However, there is necessarily a part of strategy that is hardcoded. Figure 2 presents a general model of an expert system:

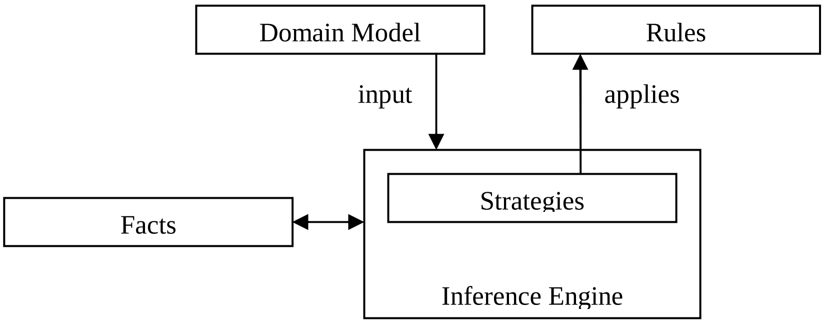

Fig. 2. Model of an expert system

A heavy ontology is composed, as shown in figure 3, of a lightweight ontology (concepts, relations between them and instances) and constraints (of any type) posed upon it (or upon other constraints). A heavy ontology is thus composed of three parts : hierarchies (of concepts and relations types), instances and constraints.

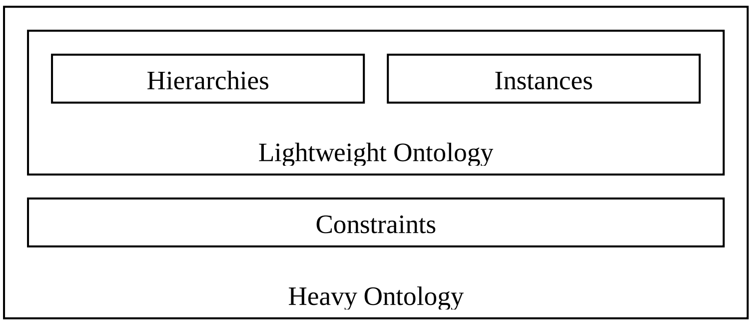

Fig. 3. Model of a heavy ontology

There is a similarity between both models. Indeed, expert system facts cans be considered as instances of ontology concepts. The domain model can thus be expressed in the ontology hierarchies. And rules are constraints on domain model and facts. They can be expressed as constraints on hierarchies and instances. Higher-order rules can be expressed as constraints on other constraints. Expressing the expert system data as a heavy ontology leads to the model presented in figure 4:

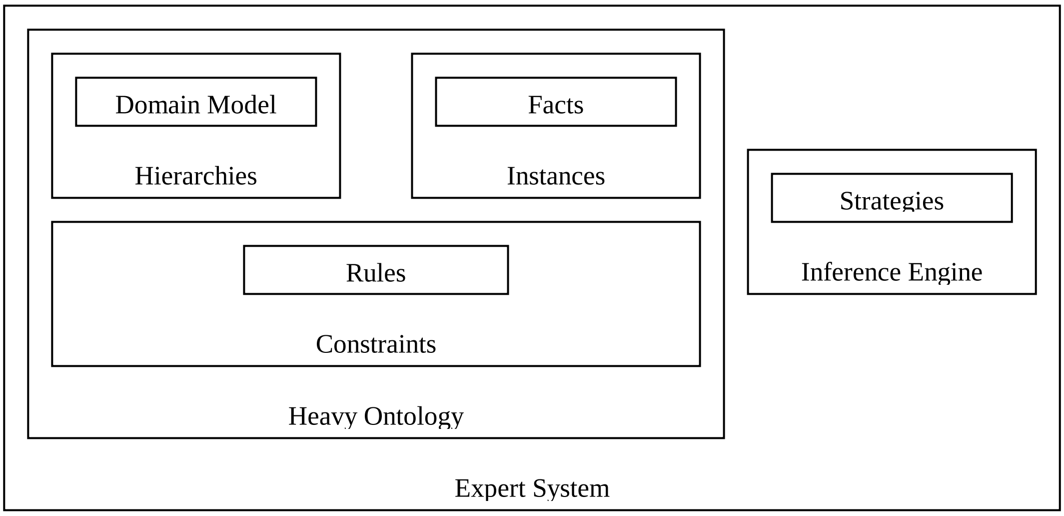

Fig. 4. Model of an expert system based upon a heavy ontology

4. ONTOLOGY ALIGNMENT

Ontology alignment [5] aims at matching different ontologies, mainly for the purpose of interoperability. Indeed, ontologies have become a standard way of describing domain knowledge in all sorts of web and other knowledge-based applications, and their number has strongly increased last years. The main problem, like in expert systems, is that when two ontologies are built to express the same domain knowledge, they are prone to be highly different (see figure 5 for an example).

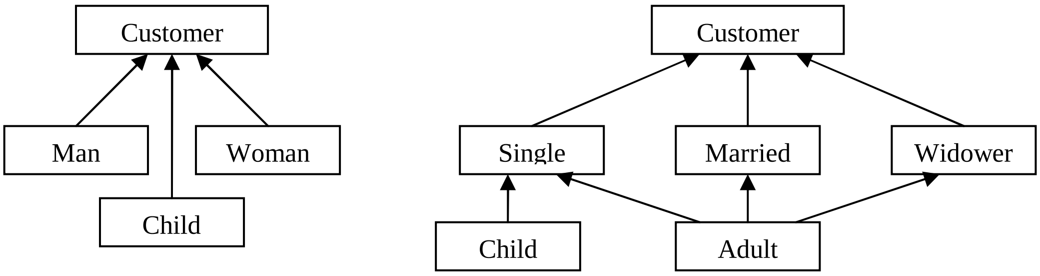

Fig. 5. A part of two different ontologies on the same domain

Web services are the application that are the most prone to be based upon ontologies and to need strong interoperability. In this context, methods have been developed for matching ontologies.

Ontology alignment works best on the lightweight part of ontologies: concepts and relation types hierarchies are aimed first, followed by instances. Depending on the differences between ontologies, several approaches are possible [2], based on the matching of terms, structures and instances, using raw data matching (especially string matching), or taking advantage of available linguistic data such as lexicons and thesauri.

When matching expert systems, there are several parts that may have to be confronted:

- The ontologies and the languages in which they are written;
- The rules and the languages in which they are written;

- The hardcoded strategies and the inference engines.

The last point requires algorithm evaluation and the availability of source code and will not be discussed here.

4.1. FACTS AND DOMAIN MODELS

Since facts are expressed in instances of an ontology, and domain models structured in hierarchies of this ontology, it is possible to match these data from two experts systems using ontology matching technologies and algorithms. Moreover, ontologies have nowadays standard languages (mostly RDF and OWL), so the respective expert systems ontologies are likely to be expressed in the same language.

The result of this matching allows to know how the expert systems approach the domain. If the approaches do not match, then further matching, at the level of rules, would be irrelevant. On the contrary, the most they match, the most the matching of higher-order rules will be beneficial.

4.2. RULES AND STRATEGIES

If experts systems have similar domains models and use similar facts, the next step is to match rules and strategies. Indeed, it is here that all the expert knowledge is contained, and matching at this level has a much greater impact in the terms that we presented in introduction.

However, although ontology alignment contains some algorithm that work on a “semantic” level (that is, on formal logics), it is much more difficult to match rules than facts or domain models.

The first thing is that rules of each expert system have to be expressed in the same language. This is often not the case, even in ontology-based systems. If languages are different, rules have to be translated into a pivot language.

There are three possible approaches to rules confrontation:

- Using the a posteriori approach of section 2., and taking advantage of the matching between facts and domain models, one can build optimised benchmarks to enhance the performance of the comparison.
- Matching the rules like the previous models is another solution, but the expression language must allow the use of advances ontology alignment algorithms.
- Arguing that local matching between rules can be sufficient for the purpose of improvement, it is possible to use less advances methods of matching –like these used on domain models– and let a human expert drive the more semantic part of the process.

5. ENVIRONMENT

The third approach to rule confrontation, as stated in the previous section, needs the steering of a human user. We now present further this approach.

There are advantages and disadvantages when using a human user rather than a fully automated process. The main disadvantage is that interactive environments are more complex to design, while the main advantage is that the user is able to perform tasks that cannot be easily formalized.

In our case, a human user has some useful abilities. He is better at many-logical order sentence disambiguation (an expert system cannot process a sentence where more than an order of logics is involved without facing ambiguity and failing). He is also good at common sense reasoning, and his intuition and analogic reasoning allow him to do quickly some part of the tasks of matching. This allows algorithms to get data that are easier to process.

We currently design [3] an environment, Platon, which makes it possible to carry out the confrontation of structures on document bases (an example is given in figure 6 below). These structures, in this particular case, are built in a pre-existent platform by the the experts themselves, and it is done in a formalism which is adapted to them. Our approach of confrontation aims at helping the expert to position himself among all others, to compare his vision with that of the others and at drawing some new ideas from the process.

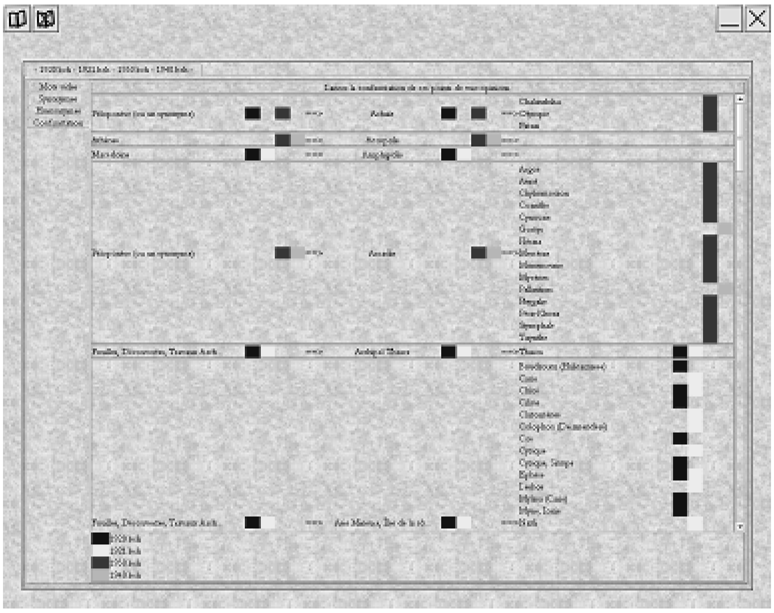

Fig. 6. An example of confrontation with Platon

Such an environment is based on iterative interaction between the user and algorithms, using a heuristic based on a convergence of attempts and errors. Confrontation is a subjective process (as subjective as the knowledge put in the systems) that involves a confronter and models to confront. The confronter has his own cognitive resources, his own strategies and procedures. Whereas an automatic system would formally tell whether two models are contradictory or not, an environment dedicated to confrontation by a human expert allows him to make attempts and links, concluding that these two models are contradictory (or not) for him. It may be another conclusion with another expert.

This means that not only the confrontation of rules, but also the confrontation of facts and domain models has to be made in the same environment. Indeed, it is difficult for a human expert to confront rules that apply to models that he did not confront. Moreover, what can be done automatically can also be done using this interactive approach, since most of the algorithms can be steered, and parametrized, by a user.

There is however a difference between our environment for document bases structures and an environment for expert systems confrontation. Since confrontation here is based on already-formal knowledge (ontologies, bases of rules and so on), the experts we are talking about are not the experts of the domain that provided knowledge for the system, but rather experts at modeling and computer specialists.

6. CONCLUSION

Expert systems nowadays can be expressed using the paradigm of ontologies. We proposed the use of ontology-related technologies, such as ontology alignment, to compare and confront different expert systems. We saw that facts and structures alignment is easily done using these technologies. For rules and strategies, however, we found that ontology alignment is unsufficient and more evolved methods have to be used.

We used a human-driven approach as the base for our proposal. We have described an environment that takes the better part of the human being and the machine, letting the user make the strategic choices and the algorithms process the chosen data with the chosen parameters.

REFERENCES

[1] BREUKER J., VAN DE VELDE W., *CommonKads Library for Expertise Modelling*, IOS Press 1994.

[2] EUZENAT J. et Al., *State of the art of ontology alignment*, KnowledgeWeb deliverable 2.2.3, University of Karlsruhe, pp. 15-43, 2004. Available online.

[3] GESCHE S., CALABRETTO S., CAPLAT G., *Managing Difference of Opinion in Semantic Structures.*, Proceedings of the International Workshop On Semantically Aware Document Processing And Indexing, pp. 79-86, 2007.

[4] RUSSELL B., WHITEHEAD A N., *Principia Mathematica*, vol. I, pp. 39-68, 1910.

[5] SHVAIKO P., EUZENAT J., *a website dedicated to ontology matching*, *http://www.ontologymatching.org*, 2007.