Extending Expert Systems with Ontological Models

Thomas Smoker

Tim French, Wei Liu
School of Computer Science and Software Engineering

Melinda Hodkiewicz
School of Mechanical Engineering

Abstract

This paper explores the practical application of extending the capabilities of an expert system with an ontology. This is achieved and benchmarked using an ontological system as an improvement to an existing expert system. Using a specific subset of rules an ontological system was built and run. The value of this system was determined using development effort, maintenance effort, robustness and performance. Overall, development effort was minimal and maintenance for the exsting expert system was improved due to built in consistency checking functionality in ontologies. The ontology is robust in situations where business knowledge is not common and subject to change. The system was of a similar speed and as accurate as the expert system. Further, each subsequent ontology that was developed on different components was developed in less time than the original, including a combined ontology. From these results it is both viable and feasible to utilise knowledge representation and reasoning by ontological extension on an expert system.

1. Introduction

Given some information and context we can infer a result; this can be thought of as making a decision about the information. If inferences are steps in reasoning, automated inference is these steps happening automatically. As such, automated reasoning can calculate inferences much faster than humans can, leading to the automation of decision making. How effective this automated reasoning is comes down to how correct the given information is, in regard to the data and the relationships within. Expert systems are one method of using automated inference using a set of rules on which to base the inference. To properly capture and make decisions with this level of granularity in the data, basic expert system functionality needs to be extended, as expert systems do not have the capability to capture the relationships and rules that connect the data. Ontologies, in this context, are representations of high level information that is machine readable (Corcho, Fernandez-Lopez & Gomez-Perez 2003). An ontology-based expert system uses semantic reasoning to connect dots between decision points that may not be currently programmed by rules in the expert system. This extension supports automated inference, generalisation and consistency checking over the data. This project aims to extend an expert-based knowledge system with an ontology, test it on development effort, maintenance effort and robustness, and identify best practices for replication.

2. Methodology

2.1 Overview

This project set out to develop an ontology that extended the capability of an expert system. The ontology is used to enter and store entities, relationships and rules within as a knowledge store, and this ontology can then be queried with logic to extract a result; for example, "which compnent causes Alarm247?". For the methodology, it has been classed into three sections: data gathering, ontology development and reasoning. These are broken down further in Figure 2.1.

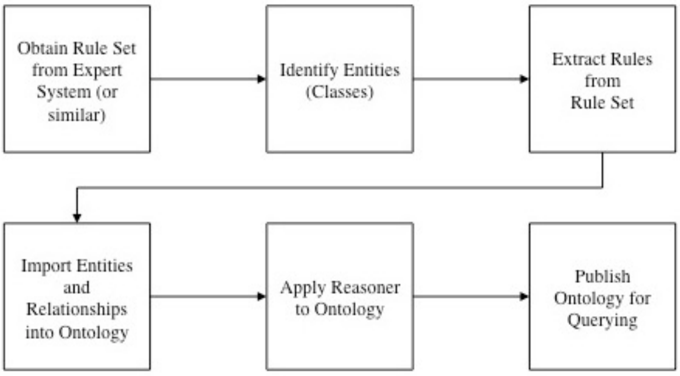

Figure 2.1: Ontology Development Process Diagram

2.2 Preperation and Data Gathering

To effectively develop knowledge representation there are a few prerequisites. Firstly, the specific context of the problem must be large enough for the organisation to be willing to invest in research and development (R&D), but not so large as to be beyond control in regard to scale and speed. Secondly, there should be distinct relationships between elements of the system so reasoning can be effective. Finally, there should be an existing system or solution for comparison, to determine whether the knowledge representation and reasoning is improving the current state of classification.

After deciding on the size and complexity of the data to be represented, next is to gather the actual data, or a representative subset. Contextualisation is crucial when building an ontology, so collaboration with Subject Matter Experts(SME's) or similarly experienced individuals around data set and format is a necessary step. Once working with the data intended to be converted into an ontology, it cannot be stressed enough to understand it on an least a moderate level. Knowledge elicitation is a necessary skill when developing a system that requires input from many parts of an organisation. Identifying key users and contributors is essential as their knowledge and use cases will shape how the system is designed and maintained. The goal is to create an effective human-machine interface to incrementally digitise mundane tasks, and inform decision makers with large and accurate amounts of data from a knowledge store.From an initial data dump it is most likely that it will need to be sanitised and simplified, and finally unified in regard to consistency. Using context and understanding developed in the data gathering stage, a simplified diagram, and therefore reference, of the information of can be developed.

2.3 Ontological Development

There are many tools available to build an ontology. Currently, the most common for of ontology would be built using academic research tools, but given the fundamentals (entities, relationships and rules) they can be built in most languages. The ontology building tool being used in our problem is called Protégé (Noy et al. 2003), an academic ontology builder that includes a convenient GUI from Stanford University. If data gathering and processing has been done correctly this will align with the current state of the rule set.

To develop the ontology and its reasoning capability, classes and object properties must be manually entered. Tiered class hierarchies can be created using built in tools, and using indentation to signify subclasses (this format will tier continuously through subsequent subclasses). A consistent naming convention should be used, for readability as well as expressiveness for the reasoner and querying. The result will be all entities entered into the system, relationships between entities set and rules programmed in to control the applicable inferences.

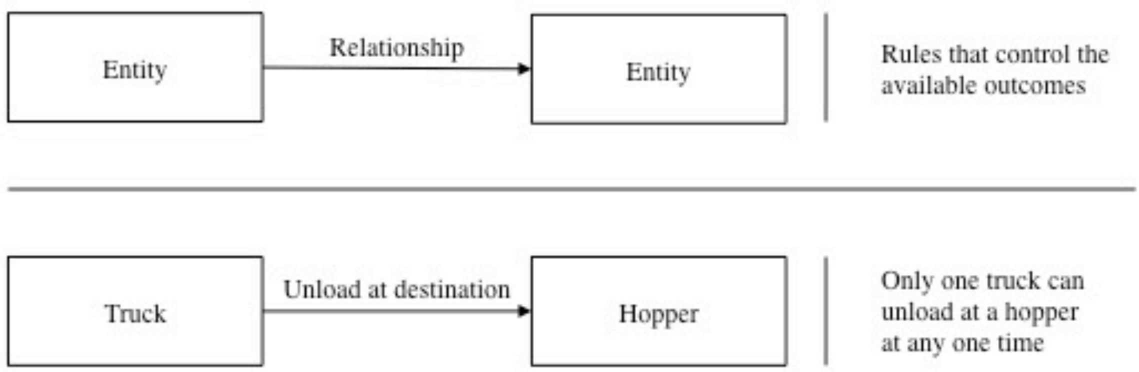

Figure 2.2 Ontology Reference and Example using Trucks and Hoppers

2.4 Reasoning, Validation and Evaluation

To query an ontology a reasoner must be used. The purpose of a reasoner is to derive information from a knowledge base, allowing inference to be made over the knowledge store. Metrics used were development effort, maintenance effort, robustness and performance. Development effort concerns not just the effort to develop an ontology from a given rule set, but also the effort for subsequent ontologies that can be developed within the same domain. Further, a combined ontology from multiple previously developed ontologies (or a combined rule set) will be assessed on the difficulty of the process of combination and performance.

Maintenance effort will be how the ontology will respond to incorrect or inconsistent entries, and what happens when new rules are added or the existing rule set needs to be changed. Robustness is how the system responds to many instances of alarms, and how dependent it is on individuals or teams rather than being a single, independent source of truth. Performance will be query speed of each system given identical queries, and the accuracy of result (which should be the same, given the ontology can only be as accurate as the expert system it is based on).

3. Results

The results of this project are based on a resource case study of specific equipment, as well as several more implementations of similar rule sets from equipment within the same business unit and a combined ontology from three of these rule sets. The metrics used to evaluate the ontologies were discussed in Section 2.4.

Development effort looks at how long it would take to replicate an ontology given an existing expert system and the relative effort of adding to and combining ontologies after an initial ontology is developed. Maintenance effort is the work required to maintain an ontology, both in real time development and for future alterations. Robustness discusses how ontologies can adapt to change within a business, and adapt to larger and more specific rule sets. Finally, performance looks at the speed and accuracy of the ontology when compared with an expert system given a similar desired result.

3.1 Development Effort

The completed ontology and expert system were compared on the amount of time and effort it takes to develop and to maintain/use. The ontology was built using Protégé and developed in OWL, and this development includes the entire process of data gathering, ontological development and querying. The bulk of development is in elicitation, modelling, system design etc. Given the existence of an expert system as a source of truth, the ontology took a further 21 hours to develop 6 entities, 8 object relationships and 190 instances of events.

Further development effort of two similar rule sets took an estimated further 4.5 hours per rule set, with the same amount of entities and object relationships, and 168 and 257 instances of events respectively. The effort for each subsequent rule set was substantially less, due to expedition of the knowledge elicitation process and reuse of ontology con- straints. A final, combined rule set was also developed, which totalled 515 instances of events. Time is estimated at 10 hours, with the same amount of entities and object relationships (this is assuming all previous rule sets are already developed).

Rule Set	Event Instances	Hours (Estimated)
Rule Set 1	190	21
Rule Set 2	168	4
Rule Set 3	257	5
Combination	515	10

Table 3.1 Time taken to Develop Ontologies given 6 Entities and 8 Object Relationships

3.2 Maintenance Effort

Maintenance was highlighted as a goal of the organisation in regard to both consistency checking current rules and inputting new rules. This is particu- larly important for growing systems as the problem only worsens with scale. If time and resources are to be invested into a rule-based decision engine, the ability to correctly and easily maintain and check the health of the system is one of the most important facts to consider.

Maintenance of whatever rule based system is chosen can be effectively separated into two partitions of concern: the ability to check the validity of and effectively reason over the current rules (and how they interlink) and how to add new rules into the system without altering the currently correct system. In the context of this problem, these have equal importance as the rule set and complexity will likely grow considerably with time, as well as initial rule sets being seen as a "Proof of Concept" to be ideally applied to further assets and processes.

3.3 Robustness

As ontologies are effectively both a conceptualisation tool and a data interface (Ceravolo, Damiani and Leida 2008), this can create confusion and therefore effect the robustness of the ontology. Robustness in this context is the reliability of the system, and how this reliability stays consistent in a changing environment. Comparative metrics used are abstraction, business change and ontological process.

In the case of alarm inference, more alarms added can scale the expert system to the point where it is unusable. Ontologies are able to use abstraction entities such as Alarm to mitigate this, and therefore be able to handle the scale resource companies do have to deal with. This is especially applicable with change, and the developed ontology responded well to such a scenario. Another issue would arise if an individual or team within the organisation leave and take valuable information with them, causing there to be a lack of context around the expert system. An ontology would have this context built in to its function so alleviate this issue.

Finally, the expert system in this project effectively captures the alarm information, but does not understand it. The ontology developed can capture and understand the information entered. From here, it can compare it with the current knowledge store, rank the event based on rules and provide a recommendation.

3.4 Performance

Performance was based on speed and accuracy on the queries from the case study. There questions and associated queries were developed with users of the expert system, and were chosen to represent the domain space. Speed is a useful metric for comparing two systems in regard to run time and time to result. Three systems were tested from time of query to end result and averaged over 10 instances: the existing expert system, the first ontology developed as an original case study and the combined ontology discussed in Section 3. The results from this can be seen in Table 3.2. In regard to accuracy, the ontological system was developed with the existing expert system as a basis, and as such can only be as accurate as the expert system currently. This resulted in the two systems being as accurate as each other.

System	Type	Average Time (Seconds)
Existing Expert System	Decision Engine	0.272
Rule Set 1 Extension	Query Processor	0.347
Combined Rule Sets Extension	Query Processor	1.216

Table 3.2 Comparison of Speed between Expert System, Initial Extension and Combined Extension

4. Conclusions and Future Work

Development took 20 hours to develop 6 entities, 8 object relationships and 190 instances of events. Further, subsequent rule sets took less time to develop, and a combined ontology from previously developed rule sets was functional and similar in performanceto smaller developed ontologies and the existing expert system. Maintenance was largely improved by the ontology due to its built in consistency checking functionality. The ontology is robust in situations where business knowledge is isolated and subject to change. In regard to usability, an ontological system built in Protégé using OWL can be queried by a subject matter expert or engineer with minimal training.

Future work still needs to be done in the areas of scale, existing infrastrcuture and ontological tools. With scale, resource companies have large amounts of data to deal with, so for an ontological system to eventually be company-wide it may be too slow to deliver impactful results. To mitigate this, further research should be done in regard to the data structure within the ontology and search function, to determine if these can be improved. Further, the effect of cloud services such as Microsoft Azure and Amazon Web Services can be considered from a cost benefit perspective.

Given the results of this project, there is reason to believe that developing an ontology from no existing infrastructure rather than appropriating an expert system that is already in use could be more effective, as data is controlled from the beginning and consistency will be easier to maintain. However, this may not be practical as most businesses have existing systems than can be extended rather than no functionality at all. Further work should be done in a resource context to determine which approach is more effective, as the state of the art suggests the functionality discussed will only continue to improve, and therefore be less repeatable and testable. Finally, the tool (Protégé) and the language (OWL) were recommended based on a previous study (Sinclair 2016), but given the nuances of businesses data and differences in data sets the recommended tools may no longer be optimal. Further work should be done to match tools with the business problem and capabilities.

5. References

Corcho, O, Fernandez-Lopez, M & Gomez-Perez, A., 2003. Methodologies, tools and languages for building ontologies. Where is their meeting point?, Data & Knowledge Engineering, 46 (1) pp. 41-64.

Noy, N.F., Crubézy, M., Fergerson, R.W., Knublauch, H., Tu, S.W., Vendetti, J. and Musen, M.A., 2003. Protege-2000: an open-source ontology-development and knowledge-acquisition environment. In AMIA Annu Symp Proc (Vol. 953, p. 953).

Ceravolo, P., Damiani, E. and Leida, M., 2008. Ontology Robustness in Evolution. In On the Move to Meaningful Internet Systems: OTM 2008 Workshops (pp. 1010-1017). Springer Berlin/Heidelberg.

Sinclair, K., 2016. Developing a Manufacturing Ontology Language Framework. 2016 CEED UWA Seminar. http://ceed.wa.edu.au/wp-content/uploads/2017/02/3.-Sinclair-Manufacturing-Ontology-Language-Framework.pdf