

Learning classification taxonomies from a classification knowledge based system

Hendra Suryanto and Paul Compton

Abstract. Knowledge-based systems (KBS) are not necessarily based on well-defined ontologies. In particular it is possible to build KBS for classification problems, where there is little constraint on how classes are organised and a class is expressed by the expert as a free text conclusion to a rule. This paper investigates how relations between such 'classes' may be discovered from existing knowledge bases, then investigates how to construct a model of these classes (an ontology) based on user-selected patterns in the class relations. We have applied our approach to KBS built with Ripple Down Rules (RDR) [1] RDR is a knowledge acquisition and knowledge maintenance methodology, which allows KBS to be built very rapidly and simply, but does not require a strong ontology. Our experimental results are based on a large real-world medical RDR KBS. The motivation for our work is to allow an ontology in a KBS to 'emerge' during development, rather than requiring the ontology to be established prior to the development of the KBS.

1. Introduction

Most knowledge acquisition methodologies first build a model of domain knowledge, before using this to build a particular problem solver. e.g KADS and CommonKADS [2], Protege2000 [3]. Although this approach facilitates re-use it does not overcome the knowledge acquisition and maintenance bottleneck and these problems are present both in the development of the ontology and consequent problem solver.

The RDR approach starts knowledge acquisition (KA) to build the problem solver immediately without any modelling apart from a simple attribute-value data representation [4]. Even the attribute-value representation can be developed while KA is in progress. The focus of the approach is to make the addition of each incrementally added piece of knowledge as simple and as reliable as possible. Although this approach facilitates KA and maintenance [5], it does not facilitate re-use, because of the lack of an ontology.

In this learning problem, we are dealing with rules rather than raw cases; the relevant attributes have already been

identified and extracted. Our aim is to discover the appropriate ontology given that the relevant attributes (and values) are already well-identified. A second aspect of the problem is that in a real-world system, attributes are multi-valued rather than boolean. In rules, conditions can then subsume each other, be disjoint, etc. For example age >10 subsumes age >50, whereas age >40 and age <10 are clearly disjoint. Our method needs to not only combine information about classes from across the knowledge base, but to address, the way in which conditions based on multi-valued attributes interrelate. Figure. 1 shows some rules for the class Satisfactory lipid profile previous raised LDL noted. In the first rule there is a condition $\text{Max}(\text{LDL}) > 3.4$ and in the second rule there is a condition $\text{Max}(\text{LDL})$ is HIGH, where HIGH is a range between 2 real number.

```
Satisfactory lipid profile previous raised
LDL noted <--
(
  (LDL <= 3.4) AND Triglyceride is NORMAL)
AND (Max(LDL) > 3.4)

OR

((LDL is NORMAL) AND (Triglyceride is
NORMAL) AND Max(LDL) is HIGH)
```

Figure. 1. an example of a class which is a disjunction of two rules.

Finally, the problem is compounded, by the way in which the expert adds conclusions. When the Multiple Classification RDR (MCRDR) KB makes an error the task of the expert is to specify the correct conclusion and identify the attributes and values that justify this conclusion. In adding the conclusion, the expert can select from a list of pre-existing conclusions organised into broad categories, but can also simply type in a new conclusion. In medical pathology result interpretation, the evaluation domain here, the conclusions added by the pathologist may provide advice to the referring clinician: on patient diagnosis, management, how treatment is progressing, whether the tests ordered were appropriate, what tests might still be necessary or any combination of the above. It is quite clear to both the expert and the receiver of the advice what information is being provided in the free text interpretation, but these interpretations are a long way from well the defined classes of a formal ontology. A task analysis would assess this domain as a classification problem, but this does not imply well defined classes. Hence the problem is not only that disjuncts for a class

(separate rule paths) may be scattered across the KB, but that the same class may be represented by different text strings. Such text strings may cover a combination of different classes. Some examples are given in Table 1.

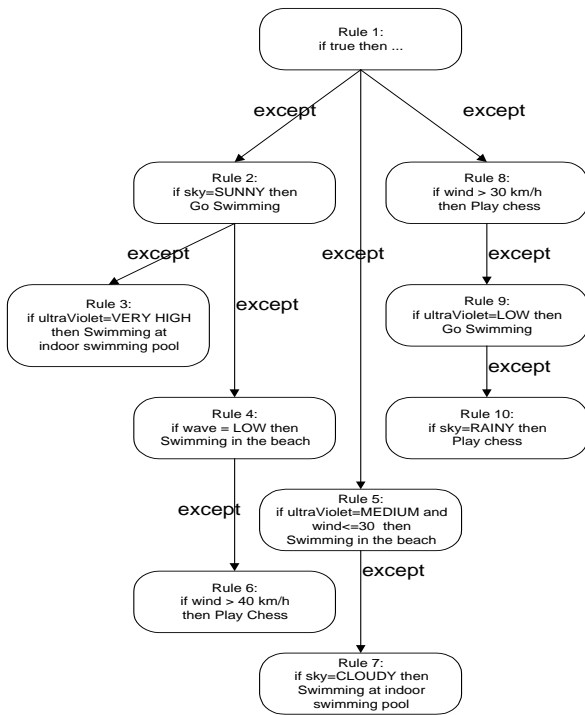


Figure 2. Example of MCRDR tree.

The question arises of whether it would be better to start with a more well developed ontology, as suggested by most KA researchers. However in practice RDR systems allow experts to very rapidly and easily build large knowledge bases [5] and recent commercial RDR systems have confirmed these advantages outside the research environment. (Pacific Knowledge Systems (PKS), personal communication) The aim of the present work then is to preserve the ease and speed of development provided by RDR systems, but overcome their lack of an initial strong ontology, by discovering the ontologies implicit in these incrementally developed systems. This may give us the best of both worlds.

RDR exception structure provides a compact representation of knowledge [6-13]. Gaines ([14]) has also generalized RDR to Exception Directed Acyclic Graphs (EDAGs). He argues EDAGs are more compact and readable than MCRDR because of a graph rather than a tree structure.

Initial RDR development was concerned with classification tasks, first single and later multiple classification. RDR has since been extended to configuration [15], heuristic search [16], document retrieval [17] and a more general RDR system for construction tasks has been proposed [18].

Figure 2 shows a simple example of the exception structure used in MCRDR. All pathways are evaluated. Evaluation on any pathway stops when a leaf node is reached or no child rule is satisfied. A conclusion is provided by the last satisfied rule in each refinement pathway. When an expert identifies that an erroneous conclusion has been given, they enter a new rule, which the system adds as a refinement.

The expert is assisted in providing an appropriate rule by the system requiring that cases that correctly satisfy the parent rule should not satisfy the child rule.

In this study we have used 5 different pathology knowledge bases provided by PKS ranging in size from 25 to 320 rules. We also have associated sets of case data ranging in number from 453 to 2218 cases. The largest PKS knowledge base is over 7000 rules, but this is the subject of other related studies.

2. Ontology learning overview

Firstly we discover class relations between rules. We consider three basic relations: subsumption/intersection, mutual-exclusivity and similarity. Secondly we specify some compound relations which appear interesting using these three basic relations. We then extract the instances of these compound relations or patterns and assemble them into a class model.

We define that class-A subsumes class-B with subsumption value 1.0, if we always have class-A when we have class-B, but not the other way around. We decide that class-A subsumes class-B from the conditions used in the rules for class-A and class-B. This syntactical subsumption needs to be evaluated by the expert - whether class-A semantically subsumes class-B as well. If the subsumption value is less than 1.0 we use the term intersection rather than subsumption. If class-A subsumes class-B with a 0 value, it means we do not have any information about the subsumption/intersection relation for class-A and class-B.

The mutual exclusivity measure of class-A and class-B is 1.0 if class-A and class-B never occur together. In RDR we evaluate this measure by checking whether there is a condition in a rule for class-A mutually exclusive with any condition in a rule for class-B (for example $age > 50$ and $age < 30$). In RDR knowledge bases parent rules are always mutually exclusive with child rule.

Class-A similar to class-B with measure 1.0 if both of the classes have exactly the same conditions. We use the term same rather than similar if the similarity is 1.0.

A class in MCRDR is the set of disjunct rule paths giving the same conclusion. A rule path consists of all conditions from all predecessor rules plus conditions of this particular node's rule. For example, (see figure 2) :

rule path 6 :

```
class Play Chess ← wind > 40,
wave=LOW, sky=SUNNY.
```

The central idea of the technique is to group all rules for each class and compute a quantitative measurement (from 0 to 1) for each relation (subsumption, mutual-exclusivity, similarity) between every pair of classes. We use this quantitative measure as an informal confidence measure as to whether these relations exist. The algorithm will be discussed in detail below, but when applied to the example in figure 2 it gives: class Go Swimming subsumes class Swimming in the beach with degree of confidence 0.83; class Play_Chess and class Go_Swimming are mutually

exclusive with degree of confidence 0.17; class Go Swimming class is similar to class Play Chess have degree of similarity 0.50. This quantitative measure enables us to group different examples of the class and provides information on whether across these examples, a class tends to subsume another class (for example, class Go Swimming subsumed class Swimming in the beach with degree of confidence 0.83). Using this quantitative measure we can say class Go Swimming tends to subsume or almost subsumes class Swimming in the beach, rather than simply saying class Go Swimming subsumes class Swimming in the beach or class Go Swimming does not subsume class Swimming in the beach.

These measures become interesting when applied to real examples such as: class [Euthyroid levels] subsumes class [Levels consistent with adequate replacement] with degree of confidence 0.866. (Medically, 'adequate' thyroid hormone replacement brings thyroid hormone levels to approximately normal levels.) Boolean values are obviously inappropriate for subsumption /intersection, mutual-exclusivity, and similarity relations in real domains. We found that in the Iron knowledge base, there were only 16 subsumption relations with degree of confidence 1.0; 8 mutually-exclusive relations with degree of confidence 1.0 and no similarity relations with degree of confidence 1.0.

We refine the subsumption/intersection measure by not only considering the conditions in the rule path but also the ratio of number of cases handled by parent and child rules in a rule-path. This allows us to deal with the situation where a rule is a gross over-generalisation and the child rule is added as correction to deal with most of the cases the parent rule would fire on. We consider that there is little value to the subsumption relation in this circumstance. For example (see figure 2) rulepath-2 has two cases, rulepath-3 has two cases, rulepath-4 has 3 cases and rulepath-6 has 3 cases. We therefore calculate the quality of rulepath-2 is $2/(2+2+3+3) = 0.2$, quality of rulepath-4 is $3/(3+3) = 0.5$, quality of rulepath-3 is $2/2 = 1.0$ and quality of rulepath-6 is $3/3 = 1.0$.

We do not need a rule quality measure if we use flat rule system rather than the refinements of a rule path, for example a flat rule for rule_2 is:

class Go Swimming \leftarrow sky=SUNNY, not(ultraviolet= VERY HIGH), not(wave=LOW).

Flat rules for rule_3 and rule_6 are the same as their rule paths since the rule path extends to the leaves and so no negation of child conditions is required. Rule_1 has three children with one, one and two conditions, and therefore we get (1 x 1 x 2) flat rules. By converting this RDR knowledge base to flat rules we get an equivalent knowledge base, but this is generally not feasible with real-world knowledge bases. In the five knowledge bases considered here, some rules have many children with several conditions. There is one rule with 10 children and every child has 5 conditions which converts to (5^{10}) flat rules.

One of the advantages of learning from rules is that we can assume that irrelevant attributes have already been discarded. This is significant as in our application domain there are hundreds of attributes. Gaines [19] argues that a rule in a knowledge base is worth many cases for learning. We adopt

the same viewpoint and note that although there is research on combining KA and machine learning and using background knowledge in machine learning, there seems little research so far in learning from a KBS rather than from cases [20], [14].

The immediate precursor of this work [20] applied formal concept analysis to ontology discovery in knowledge bases. This provided a useful way to explore concepts in a knowledge base, but because of the complexity of the conceptual lattice it was more useful to consider sub-sections of the lattice, selected by the user or by a simple nearest neighbour algorithm [21]. The critical difference from the work here, is that in formal concept analysis the difference between different concepts is emphasised. Here we attempt to combine all the concepts that represent a class and consider relations between classes rather than between concepts.

3. The class relations model

The class relations model shows the relations subsumption, mutual-exclusivity and similarity between classes and the degree of confidence that the particular relation exists [22]. We note that the measures we derive are strictly heuristic. Other superior and perhaps more well founded measures may be possible. The results here represent simply a first attempt at carrying out this type of analysis. The second point to note is that these relations have to deal with non-boolean as well as boolean data.

Let X be a class in the MCRDR framework. $\{X_0...X_m\}$ is a set of rules which have class X as the conclusion. $\{X_{i0} ... X_{in}\}$ is a set of conditions for rule X_i where $i = 0...n$, n is the number of distinct conditions in the rule path; m is number of rule paths for class X. In the MCRDR framework the class is given as a disjunction of rule paths [7].

$$\text{Then } \text{class } X = \bigvee_{i=0}^m \left(\bigwedge_{j=0}^n X_{ij} \right)$$

That is, X_{ij} stand for an individual condition in a rule path for the class X. If X is a class and Y is also a class, we could define a similarity measure as follows:

Sim (X_{ij}, Y_{ij}) = 0 if X_{ij}, Y_{ij} are different

Sim (X_{ij}, Y_{ij}) = 1 if X_{ij}, Y_{ij} are same

If α is set of distinct attributes in rule path X_i , β is set of distinct attributes in rule path Y_i , $n = |\alpha \cup \beta|$, (X_{ij}, Y_{ij}) are pair of same attributes, but they could be different conditions, (e.g. Age>50, Age=60) then we can define:

$$\text{Similar}(X_i, Y_i) = \frac{\sum_{j=0}^n \text{Sim}(X_{ij}, Y_{ij})}{|\alpha \cup \beta|} * Q(X_i) * Q(Y_i)$$

where $Q(X_i) = \text{number of cases for } X_i / (\text{number of cases for descendant } X_i + \text{number of cases for } X_i)$. Function Q measures the quality of a rule in RDR.

If the quality of a rule is close to 100%, it means that nearly all cases reaching this rule are processed by its child rules; and the child rules are rare exceptions. On the other hand if the quality of a rule is 10%, it means 90% of cases that satisfy that rule are passed to its children. That is the rule is too general and can be regarded as not being a particularly good rule and so it should not be given as strong consideration in developing the relations in the system.

For example

$$\text{Similar}(\text{rulepath-2}, \text{rulepath-6}) \\ = 1/3 * Q(\text{rulepath-2}) * Q(\text{rulepath-3}),$$

$$\text{Similar}(\text{rulepath-8}, \text{rulepath-9}) \\ = 1/2 * Q(\text{rulepath-8}) * Q(\text{rulepath-9}),$$

$$\text{Similar}(\text{rulepath-9}, \text{rulepath-10}) \\ = 2/3 * Q(\text{rulepath-9}) * Q(\text{rulepath-10}).$$

Function $\text{Similar}()$ measures a similarity between 2 nodes (each node contains a rule path).

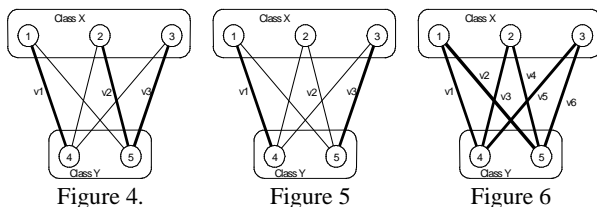


Figure 4 suggests how we can find a similarity measure between 2 classes. It shows that Class X is the disjunction of nodes 1,2 and 3 and Class Y is the disjunction of nodes 4 and 5. We propose that $\text{ClassSimilarity}(X,Y) = (v_1 + v_2 + v_3) / 3$, where we choose the v such that all nodes are covered by at least one edge and the sum of v (eg. $v_1 + v_2 + v_3$) is maximal. Note that v stands for the $\text{Similar}()$ function. In later similar diagrams v stands for the $\text{Subsume}()$ and $\text{MutualEx}()$ measures. Eg. $\text{ClassSimilarity}(\text{Go swimming}, \text{Play chess}) = ((1/3)+(1/2)+(2/3))/3=0.5$. We assume here the quality of all rule paths are 1.0 to simplify the example.

We can define a subsumption measure as follows with X_{ij} and Y_{ij} standing for individual conditions in rule paths for the relevant classes as above.

$\text{Sub}(X_{ij}, Y_{ij}) = 0$ if X_{ij} does not subsume Y_{ij}
 $\text{Sub}(X_{ij}, Y_{ij}) = 1$ if X_{ij} subsumes or same Y_{ij} (for example $A>5$ subsumes $A>10$)

If α is set of distinct attributes in rule X_i , β is set of distinct attributes in rule Y_i , then we can define:

$$\text{Subsume}(X_i, Y_i) = \frac{\sum_{j=0}^n \text{Sub}(X_{ij}, Y_{ij})}{|\alpha \cup \beta|} * Q(X_i) * Q(Y_i)$$

where $Q(X_i) = \text{number of cases for } X_i / (\text{number of cases for descendant } X_i + \text{number of cases for } X_i)$

Function $\text{Subsume}()$ measures a degree of confidence that the first rule path subsumes the second rule path. For example $\text{subsume}(\text{rulepath-2}, \text{rulepath-4}) = 2/2 * Q(\text{rulepath-2}) * Q(\text{rulepath-4})$.

Conditions in rulepath-2 are $\text{sky}=\text{SUNNY}$; conditions in rulepath-4 are $\text{sky}=\text{SUNNY}$, $\text{wave}=\text{LOW}$; and $|\alpha \cup \beta| = 2$, that is $\alpha \cup \beta = \{\text{sky}, \text{wave}\}$.

Since rulepath-2 does not have the attribute wave , we consider rulepath-2 is more general than rulepath-4 with the attribute wave . Therefore there are 2 conditions in rulepath-2 which are same as or more general than rulepath-4 . Similarly,

$$\text{subsume}(\text{rulepath-2}, \text{rulepath-5}) = 2/3 * Q(\text{rulepath-2}) * Q(\text{rulepath-5})$$

Figure 5 illustrates how we find a subsumption measure between 2 classes. It shows Class X as a disjunction of nodes 1,2 and 3 and Class Y as a disjunction of nodes 4,5 (each node contains a rule path).

We compute $\text{ClassSubsume}(X,Y) = (v_1 + v_3) / 2$. We choose the v such that all nodes of class Y are covered by at least one edge and the sum of v (eg. $v_1 + v_3$) is maximal. E.g. $\text{classSubsume}(\text{Go swimming}, \text{Swimming in the beach}) = (1+0.667)/2$. We assume the quality of all rule paths is 1.0 to simplify this example.

We can define a mutually exclusive measure as follows with X_{ij} and Y_{ij} standing for individual conditions in rule paths for the relevant classes as above.

$\text{Mut}(X_{ij}, Y_{ij}) = 0$ if X_{ij} and Y_{ij} are not mutually exclusive.
 $\text{Mut}(X_{ij}, Y_{ij}) = 1$ if X_{ij} and Y_{ij} are mutually exclusive (for example $A>5$ and $A<2$)
 $\text{MutualEx}(X_i, Y_i) = 1$, if at least one of $\text{Mut}(X_{ij}, Y_{ij})=1$

$\text{MutualEx}(X_i, Y_i) = 1$, if rule X_i and rule Y_i are parent and child.

$\text{MutualEx}(X_i, Y_i) = 0$, otherwise

Function $\text{MutualEx}()$ measures a degree of confidence that the first rule subsumes the second rule. Since the quality of the rule does not affect mutual exclusivity as much as it affects similarity and subsumption/intersection, we do not apply the quality measure to mutual exclusivity. For example

$$\text{MutualEx}(\text{rulepath-2}, \text{rulepath-10}) = 1.0,$$

Figure 6 suggests how we can find a mutual exclusivity measure between 2 classes. It shows that Class X is a disjunction of nodes 1,2 and 3 and Class Y is a disjunction of nodes 4 and 5. We compute $\text{ClassMutualEx}(X,Y) = (v_1 + v_2 + v_3 + v_4 + v_5 + v_6) / 6$. X and Y are mutually exclusive if and only if all nodes of X and Y are mutually exclusive with respect to each other (see Figure 6). Therefore $\text{ClassMutualEx}(\text{Go swimming}, \text{Play chess}) = 1/6$, since Go Swimming has 2 rulepaths, Play chess has 3 rulepaths and all $\text{MutualEx}()$ between those rulepaths are each 0.0, except for $\text{MutualEx}(\text{rulepath-2}, \text{rulepath-10}) = 1.0$.

4. Experimental results

Hormone knowledge bases system similarity-value > 0.66	
Class description	Class description
[5] interpretation [Satisfactory prolactin level.]	[12] Male interpretation [Satisfactory prolactin level.]
[14] interpretation [Consistent with premature ovarian failure.]	[28] interpretation [Consistent with premature ovarian failure. Suggest repeat FSH and oestradiol in 2-3 months to confirm.]

Hormone knowledge bases system subsumption-value = 1	
Class description	Class description
[6] interpretation [Elevated prolactin persists. Suggest TSH]	[33] interpretation [Elevated prolactin persists. Primary hypothyroidism has been excluded. IV sampling through an in-dwelling cannula can help exclude stress-related elevations of prolactin. Pituitary imaging may be required.]
	[36] interpretation [Elevated prolactin persists. Await TSH.]
[7] Interpretation [Raised prolactin in females is commonly due to medication, stress or lactation. Suggest TSH to exclude hypothyroidism and repeat prolactin after 30 minutes rest.]	[23] Male interpretation [Raised prolactin in men is commonly due to stress, medications and occasionally hypothyroidism. Suggest TSH and repeat prolactin after 30 minutes rest.]
	[30] interpretation [Raised prolactin in females is commonly due to medications, stress or lactation. Hypothyroidism has been excluded. Suggest review medications and repeat prolactin after 30 minutes rest.]

Hormone knowledge bases system mutual-exclusivity-value=1	
Class description	Class description
[4] interpretation [Consistent with perimenopause]	[9] interpretation [No evidence of perimenopause, unless patient is on oestrogen therapy.]
[5] interpretation [Satisfactory prolactin level.]	[6] interpretation [Elevated prolactin persists. Suggest TSH]

Table 1. some examples of class relations.

4.1 Class relations model

The results from the endocrinology knowledge are shown in the table 1. The results shown in each table are the class pairs with the highest similarity, subsumption, or mutual exclusivity measures. Only results with high values for these relations are shown

4.2 Extracting patterns from the class relation graph

Since there are many classes (from 25 to 100), it is impossible to consider all possible pairs of relations between the classes.

We therefore extract specific patterns which seem likely to be components of a meaningful taxonomy. For example:

Subsume(A,B)==1.0,
Subsume(A,C)==1.0,
MutualExclusive(B,C)>0.5

for all sets of three classes A,B,C from a knowledge bases

We can then combine such elements. E.g we may join Triangle(A,B,C) and Triangle(D,E,F) if A=D and MutualExclusive({B,C}, {E,F}, > 0.5). If MutualExclusive(B, C, == 1.0), we could say {B,C} are exhaustive subclass partitions of A [23]. We applied this technique to the thyroid knowledge base and obtained the result in figure 7.

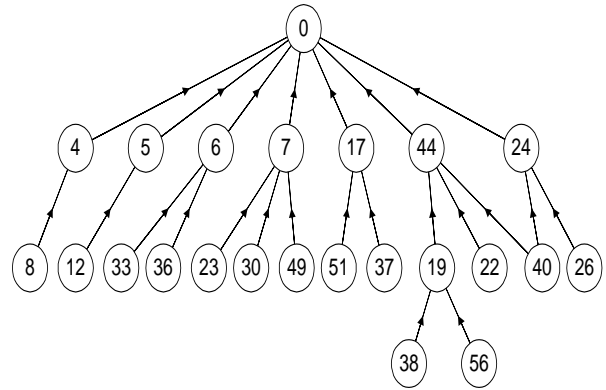


Figure 7. Partial taxonomy of the thyroid knowledge base

5. Discussion

It is beyond the scope of this paper (and the authors) to provide a detailed medical analysis of the ontologies produced by these techniques; however, it is worth noting some lay observations.

The mutually-exclusive classes in Table 1 seem reasonable. However, we also have some cases where very similar conclusions are identified as mutually exclusive. This occurs when experts make up rules that include different values for the same attribute. For example some mutually exclusive conclusions seem to give the same clinical advice but specifically refer to a patient being male or female. This may or may not be of clinical importance, but there is an obvious opportunity to have a further superclass.

The most interesting issues arise with the nature of subsumption. A superclass subclass relation may arise where one rule specifies a value for an attribute and another does not. For example a key factor in comments 6, 7, 23, 30, 33 and 36 is whether or not a TSH measurement should be ordered and whether or not primary hypothyroidism has been excluded. TSH results are important in the diagnosis of primary hypothyroidism. The generic comment suggesting a TSH measurement is given when there is no TSH result available. The comment also suggests the clinical cause of the high prolactin level remains unknown. When there is a TSH result available this provides some evidence to confirm or exclude one of the causes of the high prolactin. These relations appears to us as ontologically reasonable. However, the wording of the actual comments does not readily indicate such relationships. It would be interesting to know how the expert would react and how comments might be worded if this ontological information was available as rules were being developed.

A more general example of this pattern is the comment [0]:“patient has ovulated” which is at the top level of the

taxonomy in Fig 7. This subsumes a whole range of more specific comments related to other attributes. Again it would be interesting to see if this taxonomic information influenced the expert's wording.

5.1. Further work

At this stage we have only conducted a preliminary examination of some of the relations in the knowledge bases. A detailed examination by domain experts will be required to establish the utility of this approach. This examination may suggest that other types of measure than those suggested here may be useful and patterns other than those used in Fig 7 may be of interest for browsing the relations. At this stage we make no claim about the particular heuristic measures we have used, except that broadly, measures of this kind seem useful in discovering and exploring implicit ontologies.

We are also investigating the possibility of learning interesting patterns from a coloured graph of class relations. The frequency of isomorphic sub-graphs may be interesting. If the graph is large, then we could scale up the algorithm by applying data mining technique (e.g the apriori algorithm [24])

Finally, the present technique only considers the conditions in rule paths and proportion of cases handled by rules in determining the relations. It does not consider any other information about the classes themselves. The refinement structure for RDR does not indicate any ontological refinement; however, it does indicate that an expert thought a conclusion was inappropriate and so should be replaced by another. We are looking at the possible relations between conclusions that this would allow which could be related to the idea of relations presented here.

6. Conclusion

The work we have presented is an attempt to develop techniques to discover the ontologies implicit in knowledge bases. We believe it will be of increasing importance to carry out this particular kind of knowledge discovery as larger and larger knowledge bases come into use and we seek to exploit the knowledge in the knowledge bases in different ways. We do not make any particular claim for the techniques we have developed to date, except that they suggest that such ontology discovery is possible. The key idea in the techniques we have developed is that it seems reasonable to use heuristic quantitative measures to group classes and class relations. This then enables possible ontologies to be explored on a reasonable scale

References

1. Compton, P. and R. Jansen., A philosophical basis for knowledge acquisition. *Knowledge Acquisition*, 1990. 2: p. 241-257.
2. Schreiber, G., B. Wielinga, and J. Breuker, *KADS A Principled Approach to Knowledge-Based System Development*. 1993: Academic Press.

3. Grosso, W.E., et al. Knowledge Modeling at the Millennium (The Design and Evolution of Protégé-2000). in *Twelfth Workshop on Knowledge Acquisition, Modeling and Management*, Voyager Inn, Banff, Alberta, Canada. 1999.
4. Richards, D. and P. Compton. Knowledge Acquisition First, Modelling Later. in *10th European Knowledge Acquisition Workshop*. 1997
5. Edwards, G., et al., PEIRS: a pathologist maintained expert system for the interpretation of chemical pathology reports. *Pathology*, 1993. 25: p. 27-34.
6. Catlett, J. Ripple Down Rules as a Mediating Representation in Interactive Induction. in *Proceedings of the Second Japanese Knowledge Acquisition for Knowledge Based Systems Workshop*. 1992. Kobe, Japan.
7. Gaines, B.R. and P.J. Compton. Induction of Ripple Down Rules. in *Fifth Australian Conference on Artificial Intelligence*. 1992. Hobart.
8. Kivinen, J., H. Mannila, and E. Ukkonen. Learning Rules with Local Exceptions. in *European Conference on Computational Theory*. 1993.
9. Scheffer, T. Algebraic foundations and improved methods of induction or ripple-down rules. in *2nd Pacific Rim Knowledge Acquisition Workshop*. 1996.
10. Siromoney, A. and R. Siromoney, Variations and Local Exception in Inductive Logic Programming, in *Machine Intelligence - Applied Machine Intelligence*, K. Furukawa, D. Michie, and S. Muggleton, Editors. 1993. p. 213 - 234.
11. Compton, P., P. Preston, and B. Kang. The Use of Simulated Experts in Evaluating Knowledge Acquisition. in *9th AAAI-sponsored Banff Knowledge Acquisition for Knowledge Base System Workshop*. 1995. Canada.
12. Kang, B., P. Compton, and P. Preston. Simulated Expert Evaluation of Multiple Classification Ripple Down Rules. in *11th Banff knowledge acquisition for knowledge-based systems workshop*. 1998. Banff: SRDG Publications, University of Calgary.
13. Suryanto, H., D. Richards, and P. Compton. The automatic compression of Multiple Classification Ripple Down Rule Knowledge Based Systems: Preliminary Experiments. in *Knowledge-based Intelligence Information Engineering Systems*. 1999. Adelaide, South Australia: IEEE.
14. Gaines, B.R., Transforming Rules and Trees into Comprehensible Knowledge Structures. 1995: AAAI/MIT Press.
15. Ramadan, Z., et al. From Multiple Classification RDR to Configuration RDR. in *11th Banff Knowledge Acquisition for Knowledge Base System Workshop*. 1998. Canada.
16. Beydoun, G. and A. Hoffmann. NRDR for the Acquisition of Search Knowledge. in *10th Australian Conference on Artificial Intelligence*. 1997.
17. Kang, B., et al., A help desk system with intelligent interface. *Applied Artificial Intelligence*, 1997. 11((7-8)): 611-631.
18. Compton, P. and D. Richards. Extending Ripple Down Rules. in *Australian Knowledge Acquisition Workshop 99*. 1999. UNSW.
19. Gaines, B. R. (1989). Ounce of Knowledge is Worth a Ton of Data: Quantitative Studies of the Trade-off between Expertise and Data based on Statistically Well-founded Empirical Induction, San Mateo, California: Morgan Kaufmann, 1989.
20. Richards, D. and P. Compton. Uncovering the conceptual models in RDR KBS. in *International Conference on Conceptual Structures ICCS'97*. 1997. Seattle: Springer Verlag.
21. Richards, D., The Reuse of Knowledge in Ripple Down Rule Knowledge Based Systems, in *Artificial Intelligence Department*. 1998, New South Wales: Sydney. p. 335.
22. Suryanto H. and Compton P. Discovery of class relations in exception structured knowledge bases. *The International Conference on Conceptual Structures 2000*.
23. Perez, A.G. Evaluation of Taxonomic Knowledge in Ontologies and Knowledge Bases. in *KAW'99, Twelfth Workshop on Knowledge Acquisition, Modeling and Management*. 1999. Voyager Inn, Banff, Alberta, Canada.
24. Agrawal, R., et al., Fast Discovery of Association Rules. *Advances in Knowledge Discovery and Data Mining*, ed. U.M. Fayyad, G. Piatetsky-shapiro, and P. Smyth. 1996: