Combining Formal Concept Analysis and Ripple Down Rules to Support the Reuse of Knowledge

Debbie Richards and Paul Compton
Department of Artificial Intelligence
School of Computer Science and Engineering
University of New South Wales
Sydney, Australia
Email: {debbier, compton}@cse.unsw.edu.au

Abstract: *Ripple down rules have addressed two of the major limitations of first generation Expert Systems (ES), the maintenance and knowledge acquisition (KA) bottleneck problems. This is achieved through acquiring knowledge directly from an expert, the use of an exception structure for knowledge representation and the storing of the cornerstone case associated with each rule. Just as RDR has offered a paradigm shift in the way these problems were solved, it is expected that RDR can offer a new approach to the issue of knowledge reuse.*

Due the poor acceptance of ES by end-users, our focus is more on reusing knowledge in different modes, such as explanation, critiquing or 'what-if' within the same domain rather than the more conventional approach of reusing problem-solving methods or ontologies to solve a similar problem in a somewhat differerent domain. An evaluation of RDR for reuse showed that many modes of use were possible without any change to the knowledge or its structure but that some modes required understanding of the models represented. Since RDR does not require analysis or modeling of the domain for KA, maintenance or finding conclusions we have incorporated ideas from Formal Concept Analysis (FCA) to allow concepts and the relationships between them to be identified and explored. The addition of FCA tools to RDR is described in this paper.

1. The Reuse of Knowledge

The reuse of knowledge should result in potential savings in cost, savings in time and increase in reliability as has been recognised in the reuse of software components [17]. Given the difficulties associated with knowledge acquisition (KA) the benefits of reusing knowledge should be even greater.

The issue of knowledge reuse has been primarily handled by separation of domain and problem solving knowledge [2],[24],[37] the use of reusable ontologies [16] and design of Expert Systems (ES) at the knowledge level [27],[40]. These methods require extensive and complex modeling of the knowledge and have done little to alleviate the KA bottleneck and maintenance problems associated with large knowledge based systems (KBS) [25]. This study has chosen to use a technique, known as ripple down rules (RDR) which has addressed these issues and does not require a priori modeling of the knowledge. Just as RDR has offered a paradigm shift in the way these problems were solved, it is expected that RDR can offer a new approach to the issue of knowledge reuse.

Another difference between this study and much knowledge reuse research is the focus on what knowledge is to be reused and how. The above approaches have been mostly interested in how to incorporate existing knowledge into another system, thereby solving a similar problem in a somewhat different domain. This study pursues a more novel aspect of reuse and considers how knowledge in an existing KBS can be used to handle a different decision situation in the same domain. The motivation for this type of reuse is the poor acceptance and underutilisation of ES by end-users due to the lack of attention to computer and user cooperation issues in KBS [22],[34]. Cooperation includes the human-computer interface (usability) and the mode of interaction (usefulness) [30]. The problem is that different users, and the same user at different times, may have different needs and it will be necessary to adapt the knowledge to fit the appropriate decision situation. The need to perform knowledge modeling by most approaches has resulted in a KA focus on knowledge elicitation [34]. By using simple KA techniques we plan to focus more on user requirements and the different modes of use needed to fulfill them.

Although the type of reuse we are looking at may be different, all types of reuse require understanding of what we have and how to adapt it to fit the new situation. Current reuse research has concluded that reuse is facilitated by the capturing of contextual [16],[28], deeper [2],[38] and more [1],[17],[23] knowledge and the use of different levels of abstraction and different knowledge representations [2],[36].

The way that RDR addresses these reuse issues and the ability to easily use or adapt existing RDR implementations to handle a wide range of different modes of usage using the same knowledge base has been explored [32]. In that study the two main RDR implementations on the personal computer were used to perform KA, consultation, explanation, what-if analysis and critiquing. In some cases changes were made to the human computer interface but the representation and structure of the knowledge were not changed. These results are in stark contrast to most of the work on knowledge reuse, where the focus is heavily upon changing the problem solving method and/or knowledge representation for each task [2],[24],[37]. However, it was also found that not all modes of use can be supported without greater effort. Since different tasks have different knowledge requirements, we wanted to find a formal method for identifying the concepts we were capturing, the relationships between them and to elicite further concepts with minimal additional effort on the part of the expert to allow other future uses of the knowledge.

To this end, this paper reports on work that has been done to add the ideas from Formal Concept Analysis (FCA) [41] to multiple classification MCRDR. RDR and FCA are described in more detail in the next two sections. We then look at how the techniques have been combined and finish with a discussion of the findings and future work.

2. Ripple Down Rules

The original motivation for RDR was to attempt to deal with the situated nature of the knowledge provided by experts, particularly as observed during KBS maintenance [5]. Rules are added in response to an incorrect conclusion. The case that prompted a new rule to be added, known as the cornerstone case, is stored in association with the new rule to provide validated KA by ensuring the expert does not add any rules which would result in any of the cornerstone cases being given a different conclusion from that stored.

In initial studies the focus has been on classification tasks where only a single classification per case is required. We can define a single-classification RDR as a triple <rule,X,N>, where X are the exception rules and N are the if-not rules [35]. When a rule is satisfied the exception rules are evaluated and none of the lower rules are tested. The major success for this approach has been the PEIRS system, a large medical expert system for pathology laboratory report interpretation built by experts without the support of a knowledge engineering [10].

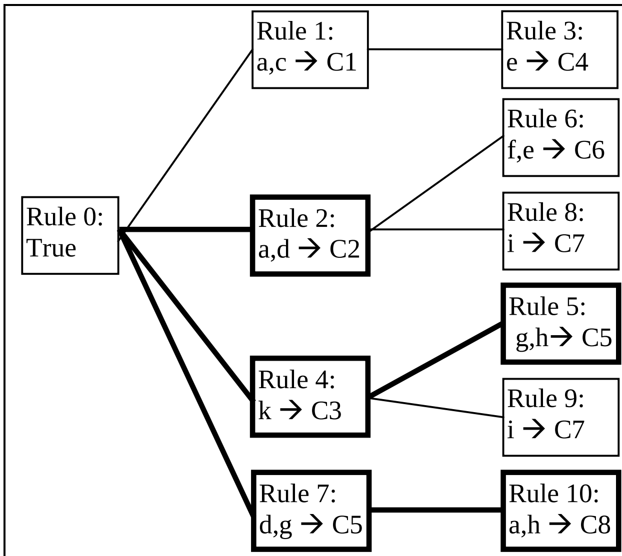

Figure 2:An MCRDR KBS. The highlighted boxes represent rules that are satisfied for the case {a,d,g,h,k}. We can see that there are three conclusions: Class 2(Rule 2), Class 5 (Rule 5) and Class 8 (Rule 10).

More recently we have developed MCRDR to deal with classification tasks where multiple independent classifications are required [19],[20]. MCRDR is defined as the triple <rule,C,S>, where C are the children/exception rules and S are the siblings. All siblings at the first level are evaluated and if true the list of children are evaluated until all children from true parents have been exhausted. The last true rule on each pathway forms the conclusion for the case. Figure 1 shows an example of an MCRDR. The MCRDR system produces somewhat more compact knowledge bases with less repetition than RDR even for single classification domains, probably because more use is made of expertise rather than depending on the KB structure [20]. MCRDR was chosen for this study since the ability to provide multiple conclusions for a given case is more appropriate for many domains and, more importantly, because the problem of how to handle the false “if-not” branches [33] does not exist.

Although RDR research has concentrated on KA and maintenance the work on reflective ES [11] and causal modeling [23] has found RDR to be an extendable representation. The rule pathway provided by the RDR exception structure has been shown [6],[32] to offer a better explanation of how the knowledge has evolved, why a rule has both succeeded and failed and what alternative pathways are possible than conventional rule traces. As mentioned in section one, the concern of this study has been more on reusing knowledge in different modes than on different problem types and since certain modes of use require understanding the models inherent in the KB rules we decided to investigate Formal Concept Analysis.

3. Formal Concept Analysis

Formal Concept Analysis, first developed by Wille [41], is a mathematically based method of finding, ordering and displaying formal concepts as a lattice [44]. The ability to express all relationships between attributes, such as which attributes occur together, and the ability to describe each object in terms of the concepts it contains and the relationship of those concepts to others is a major strength of the lattice structure [4].

	A1	A2	A3	A4	A5	A6	A7	A8
Bird	X	X			X		X	
Reptile					X		X	X
Amphibian					X	X		
Mammal			X	X			X	
Fish					X	X		X

Figure 2: Context of "Vertebrates of the Animal Kingdom". Columns A1-A8 represent has-wings, flies, suckles-young, warm-blooded, cold-blooded, breeds-in-water, breeds-on-land and has-scales.

FCA is "based on the philosophical understanding of a concept as a unit of thought consisting of two parts: the extension and intension (comprehension); the extension covers all objects (entities) belonging to the concept while the intension comprises all attributes (or properties) valid for all those objects" [44, p.493]. The set of objects and their attributes, known as the extension and intension respectively, constitute a formal context which may be used to derive a set of ordered concepts. The following description of FCA follows Wille [41].

A formal context (K) has a set of objects G (for Gegenstande in German) and set of attributes M (for Merkmale in German) which are linked by a binary relation I which indicates that the object g (from the set G) has the attribute m (from the set M) and is defined as: K = (G,M,I). Thus in figure 2 we have the context K of animals with G = {bird, reptile, amphibian, mammal and fish} and M = {has wings, flys, suckles young, warm-blooded, cold-blooded, breeds in water, breeds on land, has scales}. The crosses show where the relation I exists, thus I = {(bird,has wings), (bird,flys), (bird,cold-blooded), (bird,breeds on land), (reptile, cold-blooded),…,(fish, has scales)}.

A formal concept is a pair (X,Y) where X is the extent, the set of objects, and Y is the intent, the set of attributes, for the concept. The derivation operators:

$$X \subseteq G: \ X \mapsto X' := \{m \in M \mid gIm \text{ for all } g \in X\}$$
$$Y \subseteq M: \ Y \mapsto Y' := \{g \in G \mid gIm \text{ for all } m \in Y\}$$

are used to construct all formal concepts of a formal context, by finding the pairs (X′′,X′) and (Y′,Y′′). We can obtain all extents X′ by determining all row-intents {g}′ with $g \in G$ and then finding all their intersections. Alternatively Y′ can be obtained by determining all column-intents {m}′ with $m \in M$ and then finding all their intersection. This is specified as:

$$X' = \bigcap_{g \in X} \{g\}' \qquad\qquad Y' = \bigcap_{m \in Y} \{m\}'$$

Less formally, we take the set of objects, G, to form the initial extent X which also represents our largest concept. We then process each attribute sequentially in the set M, finding the intersections of the extent for that attribute with all previous extents. Once the extents have been found for all attributes, the intents X′ for each extent X may be found by taking the intersection of the intents for each object within the set. Thereby we determine all formal concepts of the context K by finding the pairs (X,X′).

Having found the concepts it is necessary to find the subconcept-superconcept relation between concepts so that they may be ordered and represented as a labelled line diagram. We can use the subsumption relation ≤ on the set of all concepts formed such that $(X_1,Y_1) \le (X_2,Y_2)$ iff $X_1 \subseteq X_2$. For a family (Xi,Yi) of formal concepts of K the greatest subconcept, the join, and the smallest superconcept, the meet, are respectively given by:

$$\bigvee_{i \in I} (X_i,B_i) := ((\bigcup_{i \in I} A_i)'', \bigcap_{i \in I} B_i)$$
$$\bigwedge_{i \in I} (X_i,B_i) := (\bigcap_{i \in I} A_i, (\bigcup_{i \in I} B_i)'')$$

From Lattice Theory, we are able to form a complete lattice, called a concept lattice and denoted B(K), with the ordered concept set. The concept lattice provides "hierarchical conceptual clustering of the objects (via the extents) …. and a representation of all implications between the attributes (via its intents)" (44, p.497). The lattice may be drawn as a line or *Hasse* diagram. The line diagrams in figure 4 are from our implementation, called MCRDR/FCA. The concepts are shown as small circles and the sub/superconcept relations as lines. Each concept has various intents and extents associated with it. In MCRDR\FCA it is possible to display the concept, attribute/s or object/s belonging to each node, or as in Figure 4, all three dimensions can be displayed concurrently. It is also possible to click on an individual node to see the concept number and all of its extents and intents. The labeling provided has been reduced for clarity. All intents of a concept β are reached by ascending paths from β and all extents are reached by descending paths from the concept β.

4. Adding modeling tools to RDR using Formal Concept Analysis

RDR and FCA both see that KA should be a task primarily performed by the expert, since models are imperfect representations of the world and greatly prone to variation between experts and even with one expert over time [12]. Both techniques reduce modeling to the tasks of classifying objects (cases) and identifying the salient features. However, they approach classification from alternative perspectives. FCA is concerned with identifying the similarity between objects, the conjunction of sets of attributes. RDR looks at differences between cases (objects) and is conceptually close to research based on based on Personal Construct Psychology [21] using Repertory Grids [12] and the use of a discernability matrix in Rough Sets [29].

RDR and FCA place a strong emphasis on the importance of knowledge in context, a view supported by much of the knowledge reuse community [16],[28]. Compton and Jansen [5] found that experts do not offer explanations of why they made a decision rather they offer a justification and that justification will depend on the situation. FCA is also "guided by the conviction that human thinking and communication always take place in contexts which determine the specific meaning of the concepts used" [45, p.23].

FCA differs from RDR in that some consideration of the whole domain is required, as does Repertory Grids [12], and does not consider incremental maintenance. On the other hand, the RDR structure is designed for incremental acquisition and validation. The RDR approach to KA, which simply involves the classification of cases and identification of features is probably less demanding for experts than the development of crosstables.

As discussed in section one, the motivation for adding FCA tools to RDR was because certain modes of use required the ability to understand the relationships and models inherent in the RDR rules. The following discussion refers to an enhanced version of MCRDR for Windows written in Visual C++ and Visual Basic, known as MCRDR/FCA. The screen shown in figure 3 used a 60-rule Blood Gases KBS that had been developed from the cornerstone cases associated with the 2000+ PEIRS rules[1] to evaluate the performance of MCRDR with RDR.

FCA starts with the definition of a formal context. We wanted to use the rules we already had in our KBS to form a context. To enable this, the RDR KBS was converted to a flat structure by sequentially traversing the KB for each rule picking up the conditions from the parent rule until the top node with the default rule was reached. From this flattened KBS the user chose either the whole KB or a more narrow focus of attention from which to derive a formal context. When the whole KB was chosen the rules and rule clauses formed the extents and intents, respectively. Such a global view is only feasible for small, if not very small, KBS. As with any graphical representation, as the number of rules being modeled grew, the line diagram became too cluttered to be comprehensible. This was the case even with the relatively small Blood Gases KBS. Therefore, to limit the concepts to a manageable size that could be viewed in a matrix or a line diagram the user was asked to narrow their focus of attention to a particular rule or conclusion. The decomposition of a concept lattice into smaller parts is a strategy that has previously been found useful [42]. Our approach is similar to that proposed by Ganter [14] where the context is shortened to find subcontexts and subrelations.

The current implementation has 13 various ways a user can generate a context. The simplest of these options are selecting an individual rule or conclusion. If the user selected a conclusion, all rules using the specified conclusion were selected and added as objects to the set G, forming the extents of the context. As each extent was added the clauses of the rules were added to the set M of attributes to form the intents of the context, first checking

[1] It would have been interesting to use the 2000+ PEIRS rules. This was not done at this time because they are not in the MCRDR format used by this study.

to see if any attributes had already been added by previous rules. Where the relation I held, that is object g had attribute m, a cross was marked in the appropriate row and column. If the user chose a particular rule then that rule was added as the first object with the rule clauses as the initial intension. Every clause in each rule in the flattened RDR rule base was searched for a match on the initial set of attributes. If a match was found, that rule was added to the extension and all new attributes (clauses) found in the matching rule were also added to the intension. The result was a formal context K comprised of a set of objects G and attributes M connected by the binary relation I.

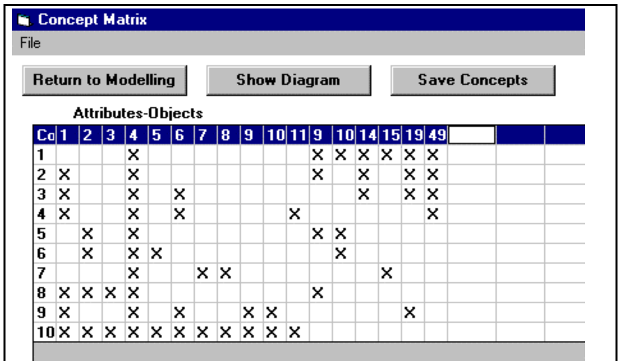

Figure 3: The Concept Matrix Screen from MCRDR/FCA. Here the user is shown the concepts for the context generated from the rules which conclude MC002 for the blood gases domain.

Treating the rule clause, which is actually an attribute-value pair, as a boolean or condition attribute is similar to the technique known as conceptual scaling [15] which has been used to interpret a many-valued context into a (binary) formal context. A many-valued context, such as that represented in an RDR KBS, is a quadruple (G,M,W,I) where I is a ternary relation between the set of objects G, the set of attributes M and the set of attribute values W (merkmalsWerte in german). Essentially, each attribute is treated as a separate formal context with the values as attributes associated with each of the original objects. A scale is chosen, such as a nominal scale (=) or an ordinal scale (≥), to order these attributes. From the many contexts, one for each attribute, the concepts are derived.

The crosstable generated in the above process was then used to construct all formal concepts of the formal context, using the process described in section 2. To allow drawing of the Hasse diagram it was necessary to compute the predecessors and successors of each concept. Predecessors were found by finding the largest subconcept of the intents for each concept. Successors were found by finding the smallest superconcept of the intents. The successor list was used to identify concepts higher in the diagram, the parents, and the predecessor list identified concepts lower in the diagram, the children. As Wille [44] points out, there is not one fixed way of drawing line diagrams and often a number of different layouts should be used because concepts can be viewed and examined in different ways depending on their purpose and meaning. In MCRDR/FCA the nodes may be repositioned anywhere providing the node is not moved higher than any of its parents or lower than any of its children.

In addition to use of the Blood Gases domain, the system-to-date has been evaluated on two other domains. One of these domains, known as LOTUS, concerns the adaptation and management of the Lotus Uliginosis cv Grasslands Maku for pastures in the Australian state of New South Wales. The knowledge is being recorded into four KBS by four independent agricultural advisors as a way of consolidating knowledge about this emerging domain [18]. The other domain was the geology domain being used for the SISYPHUS III experiments. Both domains provided an opportunity to test whether we could use MCRDR/FCA to compare and combine the conceptual models of multiple experts. There is insufficient space here to report our results but we will look briefly at the two main tools we used to assist the omparison of conceptual models: the concept matrix and the line diagram. The discussion is restricted to LOTUS.

From the concept matrices for the LOTUS KBS it was possible to see that all KBS share a number of concepts. There was a visual pattern that could be seen between all KBS and it was possible to identify any rows (concepts) in one KBS that did not match with other KBS. By looking at the matrix the experts are able to see not only what attributes (intents) and conclusions (extents) others consider important but also the relationship between them and how it affects other conclusions. As in Figure 3, we have replaced the labels of the intents and extents with numbers to fit the whole concept on the screen at the same time. However, it is difficult to understand the knowledge being modeled without labeling. To assist the user it is possible to drop down a list of attributes and/or objects with their corresponding numbers or to click on the number to get the corresponding full label.

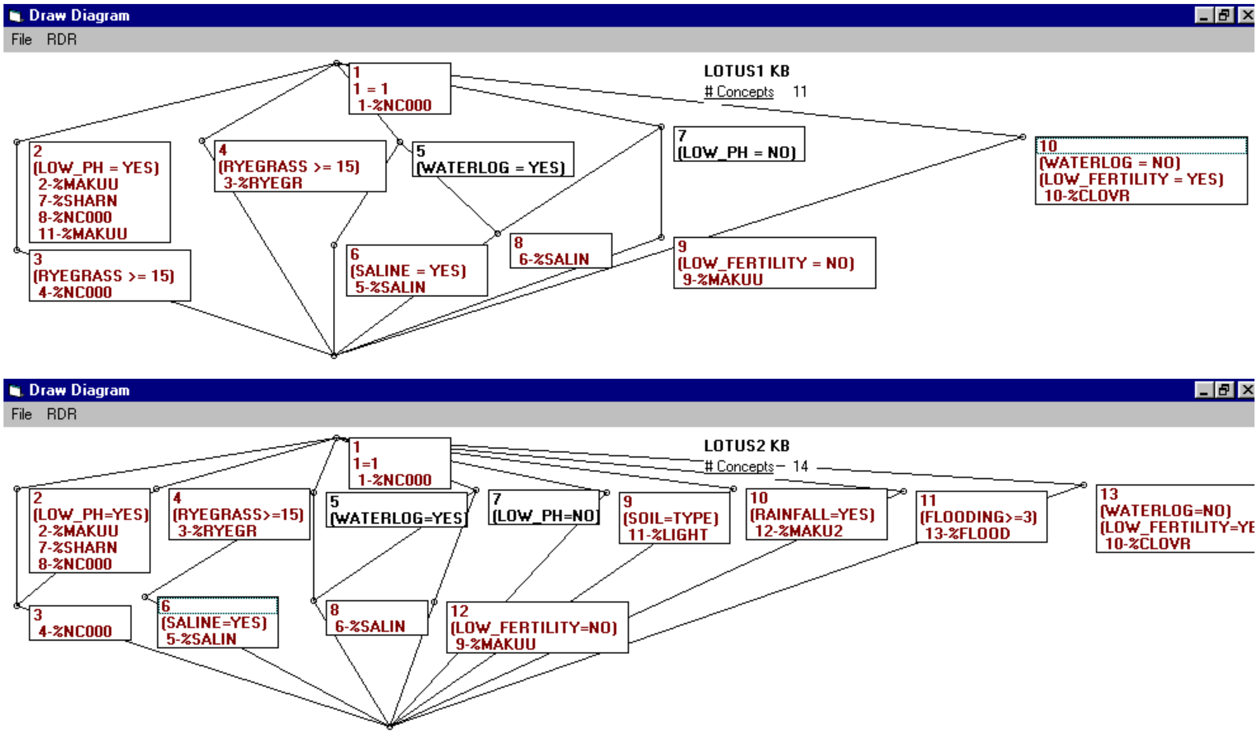

Figure 4: The line diagrams for Lotus1 and Lotus2 KBS.

The line diagrams for two of the LOTUS KBS are shown in Fig. 4. Although the labeling clutters the screen the extra information is important in understanding the diagram presented. The line diagram provides a more hierarchical understanding of the sub and super relationships in the domain. We can see that Lotus1 and 2 have 11 and 14 concepts, respectively and that the three concepts that are different are concepts number 9, 10 and 11 in the Lotus2 KBS. These concepts have introduced new attributes and conclusions (objects) not used by the Lotus1 KBS. The structure of the knowledge in both KBS is very similar with four levels of concepts in both. Even though concepts 2, 3 and 4 in both KBS appear to be slightly different structurally, due to inheritance of attributes on higher paths, both advisors consider that when (LOW_PH=YES) and (RYEGRASS>=15) the conclusion should be %NC000 No Conclusion. The concept matrix, and even more so the line diagram, provide succinct but powerful tools for analysing conceptual models. To facilitate comparison it was important to ensure the attributes shared by all contexts were in the same order before the concepts were determined.

5. Discussion and Future Directions

We can see from the examples presented that the inclusion of FCA tools into RDR supports the derivation of concepts and the relationships between them without the need for prior modeling of those concepts. While these preliminary results appear promising, we are just starting our investigations. As mentioned in section 4, the formulation of concept lattices from many-valued contexts requires their interpretation into a formal context. While the approach we took was straightforward there may be situations were a rule is relevant but has not been included in a context because it does not match on a conclusion or attribute already selected. The use of different conceptual scales [15] may provide a solution. Another method we have considered is using a distance-weighted nearest neighbour algorithm to assign a score to clauses to find the closeness of rules and/or conclusions to one another, which could be used to determine which rules should be added to a context.

Other related research that will be considered includes: the use of rough sets to find relationships in KBS [31]; a comparison of concept lattices to concept maps [13]; the use of attribute exploration for acquisition of formal contexts [43] and review of work which combined the use of repertory grids and FCA [39]. A review of these approaches may reveal if the study of similarities or differences between objects is more useful in a practical sense and what are the relative benefits of each.

The usefulness of FCA to support the reuse of knowledge in RDR KBS is the key issue that remains to be investigated. We want to see if we are able to use the matrix or hierarchy to provide a measure of closeness between concepts so that we could use the system for critiquing and also as a means of assisting the user with KA. A major benefit of providing such a powerful browsing tool would be the ability to perform what-if analysis. The pathways shown in the graphs and the proposed enhancement to allow zooming and selection of specified nodes would enable the user to test out different scenarios by dropping or adding attributes and exploring different pathways. The next step in this study is to evaluate if and how the FCA tools facilitate reuse and what further enhancements are necessary. This will require better definition of the requirements of each mode of use and a benchmark test that evaluates how well MCRDR/FCA has performed.

Acknowledgements

RDR research is supported by grants from the Australian Research Council. The authors are grateful to Zvi Hochman for provision of the LOTUS KBS and related information, to Phil Preston for the MCRDR and the Hierarchy software and to Rudolf Wille for the inspiration of this work and the ConImp software.

Bibliography

1. Buchanan, B.G. and Shortliffe, E.H. (1984) Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project Addison-Wesley Publishing Co. Reading, Machachusetts.

2. Chandrasekaran, B. and Johnson, T. (1993*) Generic Tasks and Task Structures* In David, J.M., Krivine, J.-P. and Simmons, R., editors Second Generation Expert Systems pp: 232-272. Springer, Berlin.

3. Clancey, W.J., *Heuristic Classification* Artificial Intelligence, 1985. 27: p.289-350.

4. Cole, R.J. and Eklund, P.W. (1996) *Applications of Formal Concept Analysis to Information Retrieval using an Hierarchically Structured Thesaurus* Conceptual Structures: Knowledge Representation as Interlingua (Eds. P.Eklund, G. Ellis and G. Mann), Springer.

5. Compton, P. and Jansen, R., (1990*) A Philosophical Basis for Knowledge Acquisition*. Knowledge Acquisition 2:2

6. Compton, P., Edwards, G., Kang, B., Lazarus, L., Malor, R., Menzies, T., Preston, P., Srinivasan, A. and Sammut, C. (1991) *Ripple Down Rules: Possibilities and Limitations* 6th Banff AAAI Knowledge Acquisition for Knowledge Based Systems Workshop, Banff (1991) 6.1 - 6.18.

7. Compton, P., Preston, P. and Kang, B., (1994) *Local Patching Produces Compact Knowledge Bases* A Future in Knowledge Acquisition Eds L. Steels, G. Schreiber and W. Van de Velde, Berlin, Springer Verlag, pp:104-117.

8. Compton, P., Preston, P. and Kang, B. (1995) *The Use of Simulated Experts in Evaluating Knowledge Acquisition*, Proceedings 9th Banff Knowledge Acquisition for Knowledge Based Systems Workshop Banff. Feb 26 - March 3 1995.

9. Eades, P. (1996*) Graph Drawing Methods* Conceptual Structures: Knowledge Representation as Interlingua (Eds. P.Eklund, G. Ellis and G. Mann) pp:40-49, Springer.

10. Edwards, G., Compton, P., Malor, R, Srinivasan, A. and Lazarus, L. (1993) *PEIRS: a Pathologist Maintained Expert System for the Interpretation of Chemical Pathology Reports* Pathology 25: 27-34.

11. Edwards, G., Kang, B., Preston, P. and Compton, P. (1995) *Prudent Expert Systems with Credentials: Managing* the expertise of Decision Support Systems Int. Journal Biomedical Computing 40:125-132.

12. Gaines, B. R. and Shaw, M.L.G. (1989*) Comparing the Conceptual Systems of Experts* The 11th International Joint Conference on Artificial Intelligence :633-638.

13. Gaines, B. R. and Shaw, M.L.G. (1995*) Collaboration through Concept Maps* CSCL'95 Proceedings September 1995.

14. Ganter, B. (1988) *Composition and Decomposition of Data* In Classification and Related Methods of Data Analysis (Ed. H. Bock) pp:561-566, North-Holland, Amsterdam.

15. Ganter, B. and Wille, R. (1989) *Conceptual Scaling* In Applications of Combinatorics and Graph Theory to the Biological Sciences (Ed. F. Roberts) pp:139-167, Springer, New York.

16. Guha, T.V., and Lenat, D.B. (1990*) CYC:A Mid-Term Report* AI Magazine 11(3):32-59.

17. Hemmann, Thomas (1993*) Reusable Frameworks of Problem Solving* In Catherine Peyralbe (ed.) IJACI-93 Workshop on Knowledge Sharing and Information Interchange Chamberry, France, August 29, 1993.

18. Hochman, Z., Compton, P., Blumenthal, M. and Preston, P. (1996) *Ripple-down rules: a potential tool for documenting agricultural knowledge as it emerges*. p313-316. In Proc. 8th Aust. Agronomy Conf., Toowoomba 1996.

19. Kang, B., Compton, P. and Preston, P (1995) *Multiple Classification Ripple Down Rules: Evaluation and Possibilities* Proceedings 9th Banff Knowledge Acquisition for Knowledge Based Systems Workshop Banff. Feb 26 - March 3 1995, Vol 1: 17.1-17.20.

20. Kang, B., (1996) Validating Knowledge Acquisition: Multiple Classification Ripple Down Rules PhD Thesis, School of Computer Science and Engineering, UNSW, Australia.

21. Kelly, G.A, (1955) The Psychology of Personal Constructs New York, Norton.

22. Langlotz, Curtis P. and Shortliffe, Edward E. (1983) *Adapting a Consultation System to Critique User Plans*. International Journal of Man-Machine Studies 19, 479-496.

23. Lee, M. and Compton, P. (1995*) From Heuristic Knowledge to Causal Explanations* Proc. of Eighth Aust. Joint Conf. on Artificial Intelligence AI'95, Ed X. Yao, 13-17 November 1995, Canberra, World Scientific, pp:83-90.

24. McDermott, J. (1988) *Preliminary Steps Toward a Taxonomy of Problem-Solving Methods* Automating Knowledge Acquisition for Expert Systems Marcus, S (ed.) Kluwer Academic Publishers, pp: 225-256.

25. Menzies, T.J. and Compton, P. (1995) *The (Extensive) Implications of Evaluation on the Development of Knowledge-Based Systems* in Proceedings 9th Banff Knowledge Acquisition for Knowledge Based Systems Workshop Banff. Feb 26 - March 3 1995,.

26. Mulholland, M., Preston, P., Sammut, C., Hibbert, B. and Compton, P. (1993) *An Expert System for Ion Chromatography Developed using Machine Learning and Knowledge in Context* Proceedings of the 6th Int. Conf. on Industrial and Engineering Applications of AI and ES Edinburgh.

27. Newell, A. (1982) *The Knowledge Level* Artificial Intelligence 18:87-127.

28. Patil, R. S., Fikes, R. E., Patel-Schneider, P. F., McKay, D., Finin, T., Gruber, T. R. and Neches, R., (1992*) The DARPA Knowledge Sharing Effort: Progress Report* In C. Rich, B. Nebel and Swartout, W., Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference Cambridge, MA, Morgan Kaufman.

29. Pawlak, Zdzislaw (1991) Rough Sets: Theoretical Aspects of Reasoning about Data Kluwer Academic Publishers Dordrecht.

30. Rector, A. L. (1989) *Helping with a Humanly Impossible Task: Integrating KBS into clinical care* In Proceedings of the Second Scandinavian Conference on AI, SFCAI'89.

31. Richards, D ., Gambetta, W. and Compton, P (1996) *Using Rough Set Theory to Verify Production Rules and Support Reuse* Proceedings of the Verification, Validation and Refinement of KBS Workshop, PRIFCAI'96 26-30 August 1996, Cairns, Australia, Griffith University.

32. Richards, D and Compton, P (1996) *Building Knowledge Based Systems that Match the Decision Situation Using Ripple Down Rules*, Intelligent Decision Support '96 9th Sept, 1996, Monash University.

33. Richards, D., Chellen, V. and Compton, P (1996) *The Reuse of Ripple Down Rule Knowledge Bases: Using Machine Learning to Remove Repetition* Proceedings of Pacific Knowledge Acquisition Workshop PKAW'96, October 23-25 1996, Coogee, Australia.

34. Salle, J.M. and Hunter, J., (1990) Computer/User Cooperation Issues for Knowledge-Based Systems: A Review Technical Report Aberdeen University AUCS/TR9003

35. Scheffer, T. (1996) *Algebraic Foundation and Improved Methods of Induction of Ripple Down Rules* Proceedings of Pacific Knowledge Acquisition Workshop PKAW'96, October 23-25 1996, Coogee, Australia.

36. Simmons, R. and Davis, R. (1993) *The Roles of Knowledge and Representation in Problem Solving* In David, J.M., Krivine, J.-P. and Simmons, R., editors Second Generation Expert Systems pp: 273-298. Springer, Berlin.

37. Steels, L. (1993) *The Componential Framework and Its Role in Reusability* In David, J.M., Krivine, J.-P. and Simmons, R., editors Second Generation Expert Systems pp: 273-298. Springer, Berlin.

38. Swartout, W. and Moore, J.(1993*) Explanation in second generation ES* In David, J.M., Krivine, J.-P. and Simmons, R., editors Second Generation Expert Systems pp: 211-231. Springer, Berlin.

39. Spangenberg, N and Wolff, K.E. (1988) *Conceptual Grid Evaluation* In H.H. Bock ed. Classification and Related Methods of Data Analysis Elsevier Science Publishers B.V. North Holland.

40. Van de Velde, W. (1993*) Issues in Knowledge Level Modelling* In David, J.M., Krivine, J.-P. and Simmons, R., editors Second Generation Expert Systems pp: 211-231. Springer, Berlin.

41. Wille, R. (1982) *Restructuring Lattice Theory: An Approach Based on Hierarchies of Concepts* In Ordered Sets (Ed. Rival) pp:445-470, Reidel, Dordrecht, Boston.

42. Wille, R. (1989a) *Lattices in Data Analysis: How to Draw them with a Computer* In Algorithms and Order (Ed. I. Rival) pp:33-58, Kluwer, Dordrecht, Boston.

43. Wille, R. (1989b) *Knowledge Acquisition by Methods of Formal Concept Analysis* In Data Analysis, Learning Symbolic and Numeric Knowledge (Ed. E. Diday) pp:365-380, Nova Science Pub., New York.

44. Wille, R. (1992) *Concept Lattices and Conceptual Knowledge Systems* Computers Math. Applic. (23)6-9:493-515.

45. Wille, R. (1996) *Conceptual Structures of Multicontexts* Conceptual Structures: Knowledge Representation as Interlingua (Eds. P.Eklund, G. Ellis and G. Mann) pp:23-39, Springer.

46. Zdrahal, Z and Motta, E. (1995) *An In-Depth Analysis of Propose and Revise Problem Solving Methods* 9th Knowledge Acquisition for KBS Workshop Banff, Canada, SRDG Publications, Departments of Computer Science, University of Calgary, Calgary, Canada pp:38.1-38.20