Available online at www.sciencedirect.com
ScienceDirect
Knowledge-Based Systems 19 (2006) 356–362
Knowledge-Based SYSTEMS
www.elsevier.com/locate/knosys

Experience with Ripple-Down Rules ☆

P. Compton [a,*], L. Peters [b], G. Edwards [b], T.G. Lavers [b]

[a] *University of New South Wales, Sydney, Australia*
[b] *Pacific Knowledge Systems Pty. Ltd, Sydney, Australia*

Received 28 October 2005; accepted 28 November 2005
Available online 17 February 2006

Abstract

Ripple-Down Rules (RDR) is an approach to building knowledge-based systems (KBS) incrementally, while the KBS is in routine use. Domain experts build rules as a minor extension to their normal duties, and are able to keep refining rules as KBS requirements evolve. Commercial RDR systems are now used routinely in some Chemical Pathology laboratories to provide interpretative comments to assist clinicians make the best use of laboratory reports. This paper presents usage data from one laboratory where, over a 29 month period, over 16,000 rules were added and 6,000,000 cases interpreted. The clearest evidence that this facility is highly valuable to the laboratory is the on-going addition of new knowledge bases and refinement of existing knowledge bases by the chemical pathologists.

Keywords: Knowledge acquisition; Expert systems; Knowledge maintenance; Ripple-Down Rules; Pathology

1. Introduction

The aim of this paper is to provide data on the use of the Ripple-Down Rule (RDR) knowledge acquisition technique. RDR is a general knowledge acquisition technique, but the particular application area considered here is adding clinical comments or interpretations to laboratory reports to assist the referring clinicians. That is, the clinician who orders some chemical pathology blood tests, receives not only the laboratory results but advice from the pathologist on interpretation of the results, further testing that may be required and so on. Many pathology reports have some sort of simple canned comment; the aim of using a knowledge-based systems (KBS) is to provide much more detailed comments providing expert pathologist advice on the clinical management of the specific patient.

The advantage of this area for an expert system or other AI technology is that there is no demand or expectation placed on the clinician receiving the report. The clinician does not have to interact with the system, or change their mode of operation in any way. Since a clinician chooses to order diagnostic tests for a patient presumably they will wish to view the report, including any interpretative comments. Of course, the quality of the comments will be critical in whether the clinician pays attention to them, but the quality of the comments depends purely on the level of expertise of the system, not on issues of integration into the clinical workflow. Buchanan's 1986 report of expert systems in routine use noted that three of the first four medical systems in routine use provided clinical interpretation of the results of diagnostic testing [1].

RDR were initially developed to deal with the maintenance problems of one of these first medical expert systems [2]. They were first tested in medicine in the PEIRS system [3]; however, in these studies there was a single domain expert who was intimately involved in the development and use of the system, so there has always been a question whether the technique would be as useful in other hands.

☆ This paper describes results generated by Labwizard, a software system produced by Pacific Knowledge Systems (PKS). The first author has a small shareholding in PKS, while the other authors are employees and hold shares and/or options in PKS; thus all may benefit from any increased reputation of Labwizard.

* Corresponding author. Tel.: +61 2 93855518; fax: +61 2 93854071.
E-mail address: compton@cse.unsw.edu.au (P. Compton).

0950-7051/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2005.11.022

There has been a range of other evaluation for different problems types, but this has all been in the research context. The following paper describes results based on commercial use of the Labwizard version of RDR developed by Pacific Knowledge Systems (PKS) (www.pks.com.au). Results are presented from one particular laboratory customer of PKS and the data was collected by automatic logging of all activity on the system. The chemical pathologists responsible for developing the KBS have had no involvement in this analysis and we have therefore masked the identity of the laboratory and the various areas of chemical pathology covered.

2. Methods

RDR was developed to deal with the contextual nature of expert knowledge [4,5]. In brief, when a domain expert is asked to explain how they reached a specific conclusion, they provide a justification that their conclusion is correct. Implicitly or explicitly they provide a justification that shows that their conclusion is to be preferred to other conclusions that might be considered in the context.

The two key features of RDR to facilitate adding knowledge in context are:

- First, when a conclusion provided by a KBs is incorrect, a refinement rule is linked to the incorrect rule so that the refinement rule is only ever evaluated in the same context, that is, when the parent rule also has fired. The conclusion of the refinement rule is used rather than the conclusion of the parent rule if both fire.
- Second, the expert only ever adds a rule to deal with a particular case, so that every rule has an associated case called a cornerstone case. If the expert creates a rule that will fire not only on the case in hand, but on other cornerstone cases, they are asked to add conditions to the rule to distinguish the case from the other cornerstone cases or to accept that the refinement rule should apply to one of more cornerstone cases.

There are many different RDR structures, implementing these key features in different ways, e.g. [6,7] and a range of other work linking RDR to machine learning, e.g. [8].

An RDR system is built while the system is in use; it starts with an empty KBS and is built gradually over time as cases are processed. The expert monitors the output and gradually adds rules until the conclusions provided reach the standard required. As the system evolves the conclusion given for a case will be either correct, incorrect or missing. If it is incorrect or missing the expert adds a rule which gives the correct conclusion for the case. This rule is automatically added to the KB as a refinement of the rule that gave the incorrect conclusion, and is only evaluated in this same context, or is added at the end of the previous rules and is only evaluated after the previous rules have been evaluated. This is the simplest of a range of RDR structures. With this structure only a single cornerstone case, the case associated with the parent rule, needs to be considered.

Labwizard uses a variant of the Multiple-Classification RDR structure (MCRDR) [9]. This structure is based on an n-ary tree allowing many rules to fire on a case and potentially giving many conclusions. With MCRDR many cornerstone cases also may fire on a new rule that is added. This is handled by showing any conflicting cases to the expert one by one, for the expert to refine the rule they are developing, or to allow its conclusion to be applied to the cornerstone cases. In practice only two or three cases out of possibly thousands need to be considered before a sufficiently precise rule is arrived at.

A critical feature of Labwizard is that all data available on a patient can be passed to the KBS including up to 3 years of past test results and clinical notes. This allows for highly patient-specific comments to be made. Another critical feature is Labwizard's integration into the laboratory workflow [10]. The laboratory information system (LIS) sends reports ready for output to the Labwizard server along with all other information and past results available on each patient. The server can handle multiple KBs and comments are appended to the report. The report is then validated by a pathologist to see if it should go out to the referring clinician or whether the interpretative comment is incorrect and a new rule should be added.

It is too tedious to keep checking reports where the pathologist knows from experience the system is fully reliable. To deal with this, auto-validation is provided which allows some reports to go straight to the referring clinician. Auto-validation is based on particular combinations of comments that the expert believes can be sent out unchecked, but the expert can also set the system so some percentage of these reports are sent for manual validation as a way of on-going monitoring. The human validator either confirms the comment or changes it and the report is then sent out. Any changed comments are queued for knowledge acquisition. The expert responsible for building rules for a KB reviews the cases where comments have been changed and may or may not decide to add a rule. Labwizard supports a very distributed environment: reports may come from geographically distinct laboratories, may be sent elsewhere for validation, and rules may be built at a yet another location. Although all rules may be added to the one KB, laboratories tend to develop a number of KBs for different sub-domains managed by different experts. The Labwizard server supports multiple concurrent KBs.

3. Results

The data all come from one particular commercial chemical pathology provider. The most significant results are shown in Fig. 1. Over the 29 month period since Labwizard was introduced more and more KBs were put into use and more and more patient reports were processed by the system. The KBs cover different sub-domains of chem-

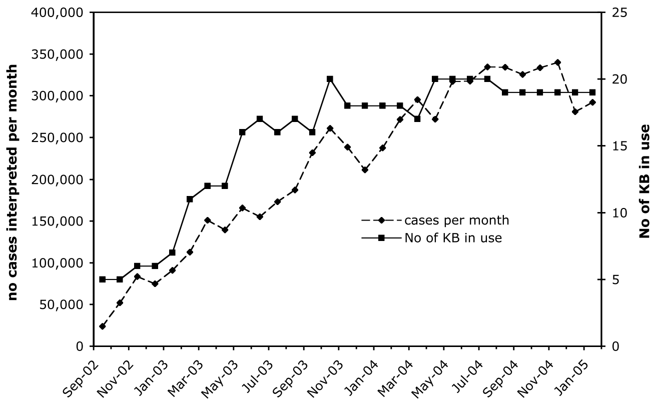

Fig. 1. The total number of patient reports processed and issued per month and the number of knowledge bases in use.

ical pathology. A particular KB is activated if the data contains relevant laboratory results, but all data available on the patient is passed to the KB. The same patient data may be passed to various KBs, e.g., thyroids and lipids.

The PKS payment model is based on the volume of reports processed. Broadly, the more reports that are processed, the more the laboratory pays PKS. This model was used to facilitate the introduction of a new technology where laboratories were unsure of the value of the technology. Fig. 1 shows the ongoing introduction of new KBS despite the increased costs that this produces. This provides very strong evidence that this particular pathology provider considered automated interpretations improved the quality of their clinical reports and their ability to provide a better quality service to referring clinicians. It is beyond the scope of the paper to consider the evidence that referring clinicians find the interpretative comments helpful, but it is clear the laboratory considers that it is achieving increased customer satisfaction.

Fig. 2 shows that the rate at which comments on reports are edited before the report is sent out. Comments are edited at the validation stage and the pathologist doing the validation is free to edit the comments on the report in any way they wish. They edit comments in three ways:

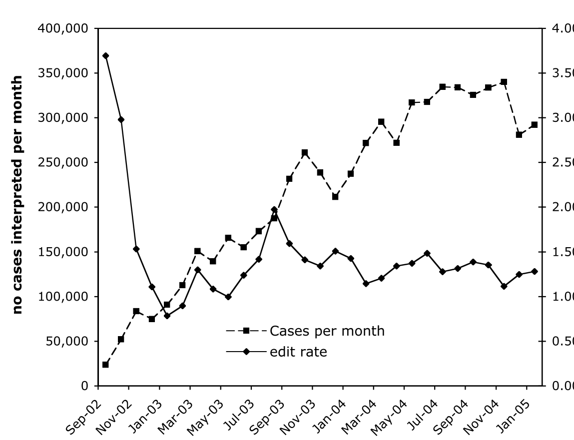

Fig. 2. The number of reports where interpretative comments have been added, changed or deleted.

- The report does not have any comment, so a clinically useful comment is added.
- The report has a comment that is incorrect and which the pathologist changes before the report goes out.
- The pathologist deletes the comment and may refer the report back to the laboratory.

The last case covers the use of Labwizard for internal laboratory quality assurance. Laboratories have realized that it can be very useful to write rules to pick up anomalous sets of results which should be referred back to the laboratory for further testing or other checking. The logs we have used for the data in this paper count all these changes to reports, but do not distinguish between them. Only the first two indicate an error or inadequacy in the KB and may result in a new rule being added; however, a new rule is not necessarily added when the comment on a report is changed. The validating pathologist may wish to add some further highly patient-specific information, perhaps because of a conversation he or she has had with the referring clinician. This case is then kept for knowledge acquisition, but the expert who adds rules (perhaps the same pathologist) may consider it is inappropriate to add a rule to provide such a patient-specific comment.

Fig. 2 shows that the fraction of reports edited rapidly drops to less than 1.5% of reports processed. This figure combines the results from all KB in use, regardless of how recently they have been introduced. If we assume 20 working days per month (although there is work done on weekends), about 220 reports per day have comments edited. Averaged across 20 knowledge bases, this is about 11 reports per day. Fig. 6 shows the same data for individual KB. It may seem surprising that the initial rate at which reports are edited is low. This may be because the experts initially add comments only for the more critical cases and allow the rest to go out without a comment. However, another important factor is the logs we have used provide only monthly totals and this masks an initially higher editing rate. Fig. 6 shows that relatively large numbers of rules tend to be added initially, and past experience with such systems shows a fairly high level of accuracy is rapidly reached. Since the fraction of reports edited drops to less than 1.5% of reports issued, and this includes quality assurance comments, the overall error rate ends up less than 1.5%. Another way of looking at accuracy is the number of rules added. The number of rules added is 0.26% of the total number of cases processed or less in the later stages of development. In conclusion is seems reasonable to surmise that this group of KBs has an overall error rate of less than 1.5%. This compares very favourably with other KBS technology, but with the added advantage that RDR allows for further correction of errors at any stage.

Since a KBS is meant to capture the expertise of the relevant domain expert, we consider an accurate/appropriate/correct comment for a report is one that the domain expert pathologist is happy with. It is the pathologist who provides advice to the referring clinician; is he or she happy that the advice provided through Labwizard is appropriate? Clearly the notion of appropriate advice varies. Included in the results below are two KBs (E & E′) for the same sub-domain of chemical pathology, but developed in different subsidiary laboratories of the parent company. Both have processed roughly similar number of cases but for one about 9000 rules have been constructed with an editing rate for cases of about 5%, while for the other only about 1000 rules have been created with a editing rate of about 1%. We are aware that the pathologist with the larger knowledge base has decided to provide more detailed educative advice than the other pathologist, but the relative clinical value of the different type of reports has not been assessed.

Labwizard's auto-validation facility enables a laboratory to decide which particular interpretations and combinations of interpretations are so reliable that the can be sent out without being validated by a chemical pathologist. The pathologist can choose to send out all interpretations of a particular type, or can choose that a random selection of these should be checked, say 5% or 10%. Reports that are auto-validated tend to be close to 100% auto-validated. Some types of reports are never auto-validated such as comments that refer a report back to the laboratory scientists for quality assurance purposes. Fig. 3 shows the overall auto-validation levels for the laboratory. Within 5 months of the introduction of Labwizard, regardless of new KBS being introduced, over 80% of reports were auto-validated. Considering that 20 sub-domains are involved, about 100 reports per sub-domain per day need to be checked.

Checking interpreted reports may result in new rules being added when cases are found for which the KB's interpretation is missing or incorrect. Fig. 4 shows the rate at which rules are added. It shows the total rules per month as well as the average number of rules per KB per month (note: the average is taken of the individual rule/month data). When all the KB were new over 100 rules were added per KB per month – about 5 per day. The time taken to add a rule is discussed further below, but overall it took 353 h, about 10 man weeks, to add the rules; i.e., an average time of 77 s per rule over 16,000 rules. At the peak development time the laboratory was investing about 20 h per month, about an hour per day, in rule development. Towards the end of the 29 month period this was about 6 h per month, less than 20 min per day.

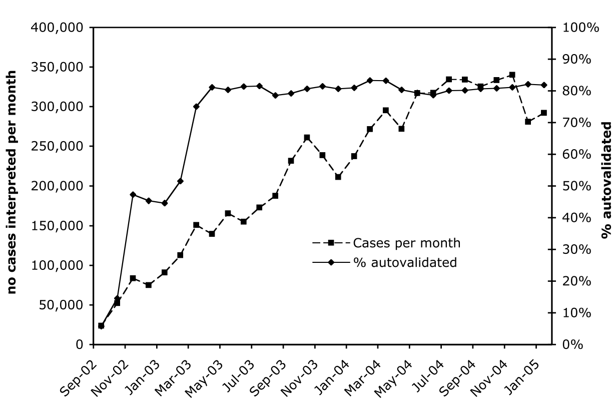

Fig. 3. The percentage of interpreted reports that are sent out without being manually checked.

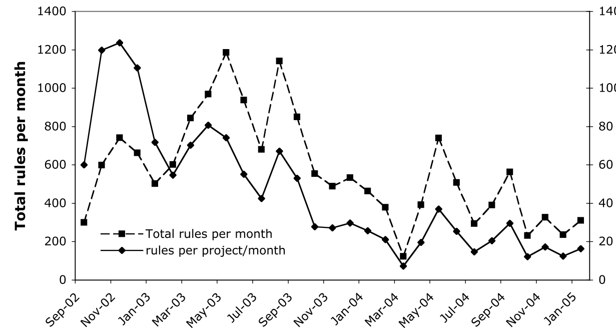

Fig. 4. The rate at which rules are added both across the whole laboratory and per KB.

3.1. Individual KB results

Data on some of the individual KBs are presented to demonstrate the variety that can occur. The domains for these KBs are specified as A, B, C, etc., as it is beyond the scope of this paper to consider how the differences in the KBS relate to particular sub-domains of chemical pathology; however, we note that E and E′ are both for the same sub-domain of chemical pathology but were developed by different experts for different local laboratories within the overall company.

Table 1 shows the expected result: that for all domains auto-validation increases and report editing decreases as the KB develops. The two lowest comment-editing rates also have the highest auto-validation rates. Although both have processed over 1 million reports one has 1061 rules, while the other has 339 rules. We assume the very low comment-editing rates and high auto-validation rates are because these KBs are not used significantly for internal quality assurance, which necessarily increases the apparent comment-editing rate. E′ is particularly interesting in that the number of rules constructed approximates the number of cases edited, suggesting the very high rejection rate is because this particular expert is seeking to develop very specific comments. Despite the relatively high and continuing comment-editing rate, the expert has been willing to markedly increase the auto-validation rate over the development.

Fig. 5 shows the average time taken to add a rule at different stages of development. Data for the graphs was obtained by taking average rule creation time from the logs at approximately 100, 250, 500, 1000 rules, etc. We are only able to provide average data as the logs used provided total time adding rules and the number of rules added since the last log download. The time adding a rule is the total amount of time the expert is logged on to the knowledge acquisition module constructing a rule for a case. It

Table 1
Summary data for 7 sample KBS

	Total cases interpreted	Total rules added	Months in use	Cases per month	Auto-validation rate (%)	Final auto-validation rate (%)	Comment-editing rate (%)	Final comment-editing rate (%)
A	1,490,767	1061	29	51,406	92	97	0.23	0.13
B	1,333,598	1091	18	74,089	72	73	2.20	1.86
C	1,205,566	339	28	43,056	86	97	0.22	0.12
D	419,555	123	24	17,481	86	87	0.64	0.88
E	271,371	1036	21	12,922	85	89	1.21	0.85
E′	187,848	9307	29	6,478	44	75	5.14	4.81
F	46,176	2021	23	2,008	82	92	2.91	1.61

The auto-validation and rejection rate are averaged over the whole period of development. The final auto-validation and comment-editing rates are the average of the last three months of use.

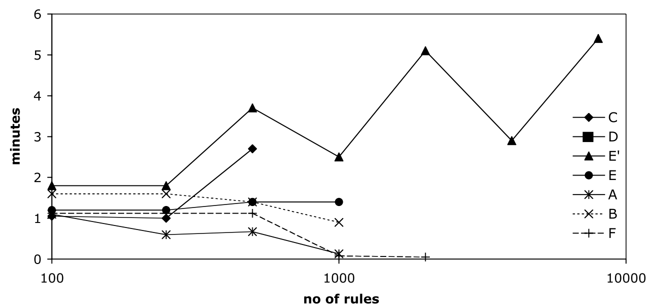

Fig. 5. The average time taken to add a rule at different KB sizes for some sample KBS.

includes time for interruptions such as answering the phone, getting coffee, etc. However, assuming that the rate at which these interruptions occur does not increase with KB size, the average time taken should provide a reasonable indicator of the relationship between knowledge acquisition and KB size.

The time taken to add a rule for some KBs decreases as the KB grows, and for two of the KBs is very small. It should be noted that we are measuring only knowledge engineering time, not the time taken in medical decision-making. The pathologist who validates the report considers whether the comment is appropriate and constructs a new comment or finds an appropriate comment to reuse. The expert who adds a rule only has the task of deciding what features distinguish a case for which this interpretation is appropriate from other cornerstone cases. This is the only knowledge engineering task in RDR and is what has been measured in these studies. This is a very rapid point and click task particularly if the same pathologist who validated the report is the expert adding the rule.

E′ provides the most interesting data because of its much larger size and because the average time to add a rule to E′ increases roughly proportional to the log of KB size. There are four possible reasons why the knowledge acquisition time increases as KB size increases:

- The cases being dealt with are increasingly unusual and may take more time to think about.
- The number of cornerstone cases that the expert has to consider, increases.
- The number of cornerstone cases that have to be processed by the system during knowledge acquisition, increases.
- Because the rule building for E′ takes place at a remote site from the server, with a relatively poor link, downloading increasing numbers of cornerstone cases may slow the process.

Despite this combination of factors, these results are consistent with informal observations of experts where they generally take a minute or two to add a rule. The major claim of RDR that it is very simple and rapid to add a rule, largely independent of KB size, is supported by this data.

4. Conclusions

Relatively few medical systems reach routine clinical use, and of those that do, the reasons are often unclear: is it because of the clinical value of the system or because of the particular values and interests of the organization or individuals involved? In the data that is presented here, the pathology provider using Labwizard is a commercial company whose aim is to generate financial returns for shareholders by providing high quality diagnostic laboratory services. Despite an explicit link between the volume of Labwizard usage and costs, the pathology company has chosen to increase its use of Labwizard by adding new knowledge bases throughout a 29-month period. The conclusion from this is that the pathology provider believes that the very specific clinical advice provided through Labwizard is of considerable value in satisfying its clinician customers and thereby increasing market share.

A central claim for RDR has been that new rules can be added throughout the life of the system, very simply and easily. The data presented show that rules are added throughout the life of the KB, and that the time taken to add a rule is only a few minutes regardless of the size of the KB. The maintenance of a KB is carried out by the relevant chemical pathologist and is at their discretion. The ongoing addition of rules suggests that pathologists see a value and little cost in adding further refinements as required. It has also been argued that the incremental approach of RDR also enables the KB to evolve as requirements in the domain evolve.

Although the time to build an individual rule is small, the question remains of whether the RDR structure is efficient or whether perhaps the expert is required to add repeat knowledge, with the same knowledge added in different contexts. We do not have a definitive answer to this except to note that simulation studies show RDR produce KB similar in size to those developed by machine learning [11]; that attempts to compress early Labwizard KBS resulted in only about 10% compression [12], and above all that there is very little complaint from the pathologists concerned about the knowledge acquisition process. We have studies under way comparing the efficiency of a number of KBS structures. The conclusion from all this is that although RDR structures may introduce some repetition, the cost of this is small compared to the ongoing ease of adding rules. By any measure, an overall total of 353 h to build the KBS described here is a very small investment.

A question that arises from this study is: what is the appropriate level of specificity in interpretative comments? Comments that are too general, or cover too many different cases, rather than the specific situation of the patient will tend to be ignored by the clinician receiving the report.

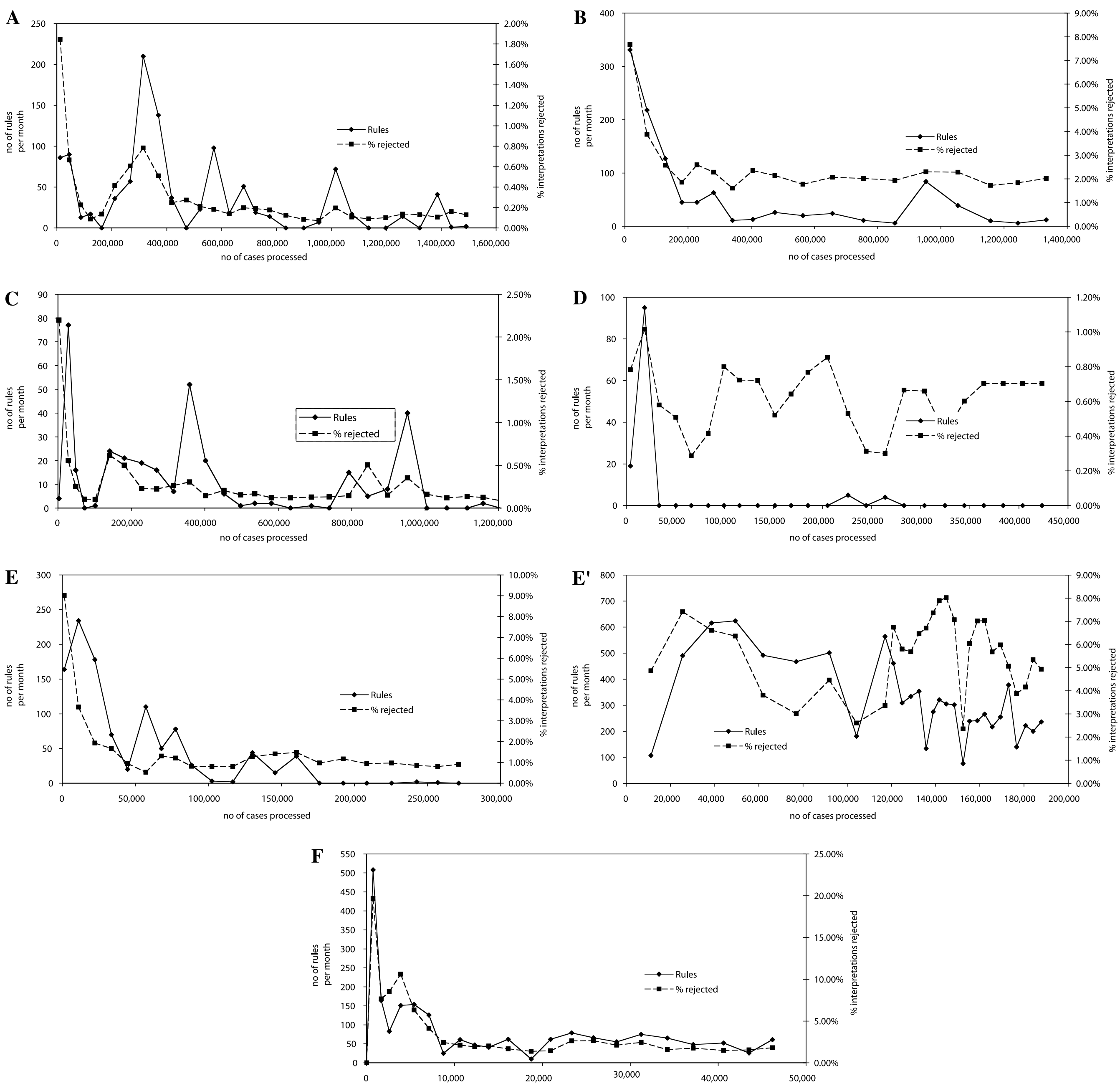

Fig. 6. A selection of knowledge bases, showing the number of rules added per month and the percentage of reports rejected, i.e., where the comments are edited before reports go out, against the number of cases processed.

For a comment to provide useful management advice, it must take into account the patient test history, medication, clinical notes and demographics, not just the clinical guidelines for the current tests. That is, it should answer the specific question the clinician is asking with regard to this patient. Second, the comment needs to be directed to the clinician who requested the diagnostic tests. What sort of information will be helpful to him or her; what are they likely not to know or to miss? Obviously it is superfluous to tell a specialist about their area of specialty so the simple comment "specialist management noted" may be preferred in some cases. The graphs in Fig. 6 show a very rapid decrease in the comment-editing rate in the early stages of development. Probably a good strategy for the expert is to produce rules for fairly general and common comments early on, or to concentrate on a few clearly defined objectives such as "normals". In this way auto-validation can be set high early in the project's development, reducing the manual validation load on the expert. Over time the expert adds more and more refined and patient-specific comments as seems appropriate.

There has been a strong move in medicine towards standardization. In contrast Labwizard is used by laboratories to compete for market share by having higher quality, more helpful, interpretative comments than other laboratories – comments that are aimed specifically at the clinicians likely to use the laboratory and the type of patients they have. The data from E and E′ shown here, suggest that there might be quite significant differences between how expert pathologists prefer to advise other clinicians, even in the same sub-domain. Although pathologists can be trained in a few hours to build rules, it can take some time building rules before they appreciate the best level of granularity for rules and comments. Some pathologists will tend initially to use Labwizard as another way to produce very coarse generic comments, while others will develop rules that are so specific they are unlikely to be used again. It is unclear whether the large size of E′ compared to E is because of the more educative nature of the comments or because the rules are unnecessarily specific. It will be fascinating to compare KB for the same sub-domains by different experts, with perhaps different purposes, and some initial steps have already been taken in developing techniques to do this [13].

The application described here has been in medicine, but this gradual approach to building a system over time can be applied to any area where it is natural and appropriate to monitor the performance of the system. Help desks are an obvious example. Other examples included financial systems such as loan systems or monitoring for fraud. In fact most industrial and commercial applications of KBS deal with a stream of cases. A system can be built and then tested on the stream of cases, or with the RDR approach the stream of cases can be used to build the system. The rate at which changes need to be made in the RDR system at a particular stage of development corresponds to the error rate on unseen test cases in a more conventional evaluation. The difference is that with the RDR system the errors in the test cases can also be fixed and the system further improved.

Acknowledgements

RDR research has had long-term support from the Australian Research Council and PKS development was assisted by a grant from AusIndustry. The authors thank the referees for their helpful comments.

References

[1] B. Buchanan, Expert systems: working systems and the research literature, Expert Systems 3 (1986) 32–51.
[2] P. Compton, K. Horn, R. Quinlan, L. Lazarus, Maintaining an expert system, in: J.R. Quinlan (Ed.), Applications of Expert Systems, Addison-Wesley, Reading, MA, 1989, pp. 366–385.
[3] G. Edwards, P. Compton, R. Malor, A. Srinivasan, L. Lazarus, PEIRS: a pathologist maintained expert system for the interpretation of chemical pathology reports, Pathology 25 (1993) 27–34.
[4] P. Compton, R. Jansen, A philosophical basis for knowledge acquisition, Knowledge Acquisition 2 (1990) 241–257.
[5] D. Richards, P. Compton, Taking up the situated cognition challenge with ripple down rules, International Journal of Human Computer Studies 49 (1998) 895–926.
[6] G. Beydoun, A. Hoffmann, Theoretical basis for hierarchical incremental knowledge acquisition, International Journal of Human Computer Studies 54 (3) (2001) 407–452.
[7] T.M. Cao, P. Compton, A simulation framework for knowledge acquisition evaluation, in: V. Estivill-Castro (Ed.), Twenty-Eighth Australasian Computer Science Conference (ACSC2005), Newcastle, (CRPIT 38 Australian Computer Society 2005) 353–360.
[8] T. Yoshida, H, Motoda, T. Washio, Adaptive ripple down rules method based on minimum description length principle, in: Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002) (IEEE Computer Society, 2002) 530–537.
[9] B. Kang, P. Compton, P. Preston, Multiple classification ripple down rules, in: R. Mizoguchi (Ed.), Proceedings of the Third Japanese Knowledge Acquisition for Knowledge-Based Systems Workshop (JKAW'94) (Japanese Society for Artificial Intelligence, 1994) 197–212.
[10] P. Compton, G. Edwards, L. Lazarus, L. Peters, M. Harries, Knowledge based system, U.S Patent 6,553,361, 2003.
[11] P. Compton, P. Preston, B. Kang, T. Yip, Local patching produces compact knowledge bases, in: L. Steels, G. Schreiber, W. Van de Velde (Eds.), A Future for Knowledge Acquisition: Proceedings of EKAW'94, Springer, Berlin, 1994, pp. 104–117.
[12] H. Suryanto, D. Richards, P. Compton, The automatic compression of multiple classification ripple down rule knowledge base systems: preliminary experiments, in: L. Jain (Ed.), Proceedings of the Third International Conference on Knowledge-Based Intelligent Information Engineering Systems. (IEEE Cat. No. 99TH8410, 1999) pp. 203–206.
[13] H. Suryanto, P. Compton, Discovery of ontologies from knowledge bases, in: Y. Gil, M. Musen, J. Shavlik (Eds.), Proceedings of the First International Conference on Knowledge Capture, ACM, New York, 2001, pp. 171–178.