RM and RDM, a Preliminary Evaluation of Two Prudent RDR Techniques

Omaru Maruatona, Peter Vamplew, and Richard Dazeley

Internet Commerce Security Laboratory
University of Ballarat Ballarat, Australia
o.maruatona@icsl.com.au

Abstract. Rated Multiple Classification Ripple Down Rules (RM) and Ripple Down Models (RDM) are two of the successful prudent RDR approaches published. To date, there has not been a published, dedicated comparison of the two. This paper presents a systematic preliminary evaluation and analysis of the two techniques. The tests and results reported in this paper are the first phase of direct evaluations of RM and RDM against each other.

Keywords: Prudence Analysis, RDR, MCRDR, KB brittleness, RM, RDM.

1 Introduction

Traditional knowledge based systems (KBS) have been often criticized for ignoring Knowledge Acquisition (KA) and maintenance innovations [1], [2]. Consequently, Ripple Down Rules (RDR) was introduced as an incremental KA technique whereby KA and maintenance are essentially integrated and usually not requiring the additional services of a knowledge engineer. RDR has since been used in commercial applications including in the Pathology Interpretative Expert Reporting System (PIERS) system, which has been described as user maintained and not requiring knowledge engineering expertise [3]. Due to RDR's inability to provide more than a single classification, Multiple Classification RDR (MCRDR) was introduced with the ability to generate multiple classifications [4]. A further advancement in RDR technologies was the idea of Prudence Analysis (PA). Prudence was introduced to address KBS brittleness, which occurs when a KBS does not realise when its knowledge is inadequate for a particular case [5]. A prudent KBS is one with a mechanism to issue warnings or alerts whenever a current case is beyond the system's expertise. This paper reports on a methodical comparison of two PA techniques: RM and RDM. These two methods had been independently evaluated before but have never been directly compared. Another contribution of this paper is the introduction of a Multiple Classification version of RDM.

2 Rated MCRDR (RM)

RM is a hybrid approach combining MCRDR with an Artificial Neural Network (ANN) [6]. RM is based on [7]'s premise that if captured, a pattern of fired MCRDR

D. Richards and B.H. Kang (Eds.): PKAW 2012, LNAI 7457, pp. 188–194, 2012.

rules can provide an additional context about a given domain. A grouping of this pattern can be given a value representing its contribution to a particular task [7]. RM has a MCRDR output simplifying mechanism which indexes MCRDR conclusions into a set of binary inputs for the ANN. These inputs are assigned a 0 or 1 value depending on whether the particular rule was fired for the current case. The following diagram illustrates the basic composition of RM.

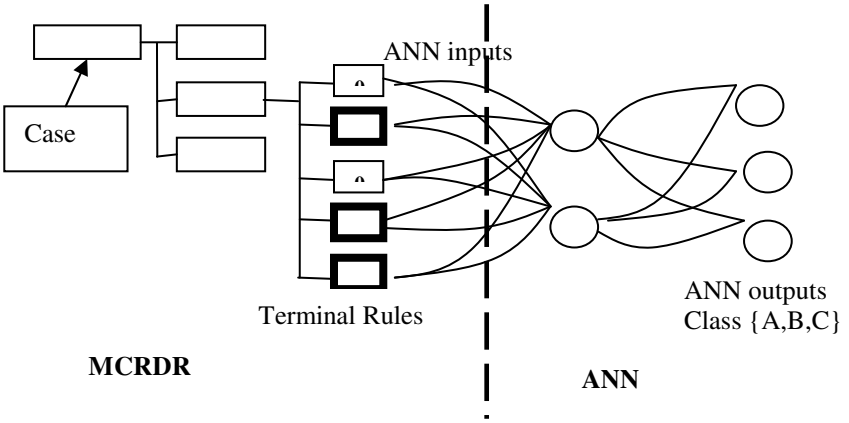

Fig. 1. A basic RM schematic. The bolded MCRDR outputs represent 1 and 0 for the non bolded outputs.

The indexed binary set is fed into a standard 3 layer perceptron ANN such that each firing terminating rule will produce a 1 input for the ANN, and a 0 input for non firing terminal rules. For example, in the RM diagram, the terminating rules are indexed into a binary word 01011 which is the input for the ANN. The ANN uses two main learning approaches. If there are no new rules added to MCRDR, a standard back-propagation algorithm with a sigmoid thresholding function is engaged. If a new rule is added in MCRDR, an additional input is created for the ANN. This may be problematic to the ANN in terms of erasing the previously learned information. To counteract this threat, new shortcut connections are introduced from the newly created input to each output node. The shortcut weights are calculated using the single step initialization formula [6] (see equation 1 below).

$$w = \left(\log\left(\frac{fnet + \partial + 0.5}{0.5 - (fnet + \partial)}\right)\right) - (((\sum A) + (\sum B))/m) \tag{1}$$

where A and B are the weighted sums at the hidden and output nodes respectively, $\mathbf{z}$ is the step distance modifier in the range of 0 to 1. It is the rate of adjustment of for the new features and determines how quickly the shortcut weights adjust to the correct output. m is the number of newly added inputs and ∂ is the sum of differences between the network calculated outputs and the target outputs (or error sum value) at an

output neuron. As the MCRDR produces different classifications, the ANN learns the patterns of the fired rules for each classification. A warning is then given whenever the MCRDR and the ANN produce different classifications.

3 Ripple Down Models (RDM)

RDM, like RM has two main components, the RDR part and a complementary outlier detection mechanism. As in RM, RDM first engages an RDR engine and passes the output to the complementary outlier detection component. In RDM, the RDR output passed to the outlier detector is a model (hence the acronym RDM) [8]. A model is made up of situated profiles. Each situated profile consists of a number of profiles corresponding to the number of attributes in a case. RDM has two outlier detection functions: the Outlier Estimation with Backward Adaptation (OEBA) for continuous attributes and the Outlier Detection for Categorical Attributes (OECA) for discrete attributes [9].

For OEBA, profiles of each attribute in a case are grouped as a Situated Profile and organised according to the conclusions generated by RDR. For example, an OEBA Situated Profile may contain minimum and maximum values for each attribute for the corresponding RDR classification. For each classification produced by RDR, a Model comprising the Situated Profile(s) is returned to the outlier detection component. Ideally, OEBA should flag an anomaly for incorrect classifications by RDR. If an outlier was flagged incorrectly, then Backward Adaptability adjusts the appropriate profiles' minimum and maximum values. In OECA, each profile keeps a set of an attribute's values, a corresponding *M* value and a New Value Ratio (NVR. The NVR is the ratio of the current attribute's *M* value and the *M* value for the last updated value in the profile [8]. An anomaly is flagged when the NVR of a case is greater than a set threshold.

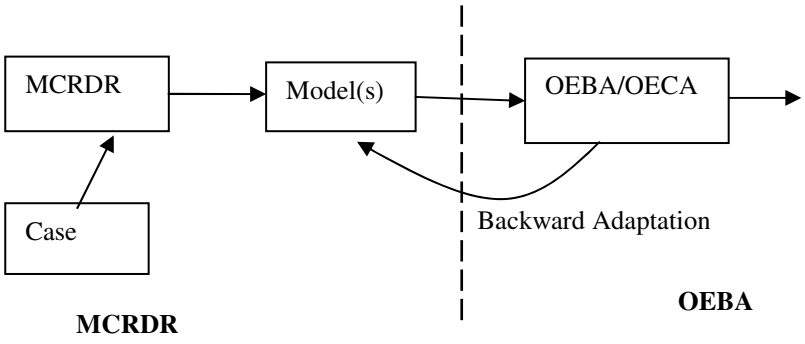

Fig. 2. RDM schematic

Originally, RDM was designed with models passed from a single class RDR engine [8]. This research developed a multiple classification version of RDM where models were passed from a MCRDR rule base. The primary difference between the two versions is that the MCRDR alternative has the ability to generate multiple models from a single case if need be. As in RM, the prudence of RDM is in how well the warning system works. Figure 2 shows the general architecture of RDM.

4 Evaluation Methodology

4.1 Simulated Expert

Evaluating KA methods is an important but difficult task mainly because it is hard and expensive to get a readily available expert for controlled tests [10]. A common solution to this problem has been the use of simulated experts. Simulated experts have been used extensively in testing RDR methodologies [7], [6], [10]. For this research, the simulated expert uses a ruleset file (for each dataset) generated from the See5 tool. For each dataset, only cases that could be matched to a rule (or condition) were used such that the resultant simulated expert was faultless and missed no cases.

4.2 Test Data

Three simple UCI datasets were used for these tests mainly because developing perfect simulated experts for such data is less time consuming. The tests reported in this paper are a preliminary part of a wider research project. Table 1 describes the three datasets used. The last column of the table shows the ratio of each dataset's rules to the total number of cases.

Table 1. Description of datasets

Name	Type	Instances	Rules in SE	Rules Ratio
Iris Plants	Numerical	146	5	3%
Car Evaluation	Categorical	288	15	5%
Physical Action	Numerical	250	60	24%

4.3 Evaluation Metrics

The comparison of the two PA systems, RM and RDM was based on two metrics: Balanced accuracy and prudence. Balanced accuracy is based on the system's True Positives (TP), True Negatives (TN), False Negatives (FN) and False Positives (FP). The TP in this case is when a warning was issued correctly. TN is when a warning was not issued correctly. FN includes instances when a warning was not issued but should have. When a warning was issued incorrectly, then this is a FP [8]. Balanced accuracy combines these metrics to prevent a scenario where a system can warn on every case and still get an accuracy of 100%. The following formula is used for balanced accuracy:

$$bA = TP(\alpha/T)/\beta + TN(\beta/T)/\alpha \quad (2)$$

where α = TN + FP, β= TP+ FN and T= TN + TP + FN + FP.

The prudence of a system will be determined by the effectiveness of its warning mechanism. This will be the rate of correct warnings minus the rate of incorrect warnings. For example, given a dataset of 40 cases in which 10 are TP's and 30 are TN's, a system with 8 TP's and 3 FP's will have a prudence measure of (8/10)% - (3/40)% which is 72.5%. Formula 3 is used to calculate the prudence measure.

$$p = (TPs/TP)\% - (FPs/T)\% \quad (3)$$

where *TPs* is the number of correct warnings issued by the system, *TP* is the total number of warnings that should have been issued. In this study, the total number of warnings expected is the number of rules in the simulated expert. *FPs* is the incorrect warnings issued by the system and *T* is the total number of cases in a dataset.

Incorrect warnings and the proportion of the TP in the data also affect the overall prudence measure of a system. In cases where a dataset has a large proportion of TP's, it may be better to lower the warning threshold so that the system issues more warnings. As the dataset grows and fewer rules are added to the system, there might be a need to raise the warning threshold effectively increasing the system's prudence. This is because as the system acquires knowledge and sees fewer new cases, the frequency of warnings is likely to decline.

5 Results and Analysis

Table 2 displays the two PA systems' corresponding TP, TN, FP and FN metrics on the three datasets. For the Physical Action dataset, RM was tested with two different **ƶ** values, 0.01 and 0.9. In the other datasets, RM's **ƶ** value was set at 0.5.Table 2 shows the two systems' balance accuracy (calculated using formula 2) and prudence measures computed from formula 3.

Table 2. RM and RDM's confusion metrics, Balanced Accuracy (BA) and Prudence (P)

Dataset	System	TP	TN	FP	FN	BA (%)	Pr (%)
Iris	RM	4	142	0	0	100	100
	RDM	4	135	7	0	99.86	95
Car Evaluation	RM	115	48	73	52	52	40
	RDM	160	102	23	3	89	42
Physical Action	RM (= 0.01)	146	8	89	7	42	30
	RM (= 0.9)	119	47	50	34	60	56
	RDM	160	29	40	21	54	44

5.1 Analysis

The RM method seems to have a slightly better balanced accuracy and prudence over RDM in the Iris dataset. In the Car Evaluation dataset, RDM outperformed RM by a vast margin in terms of BA but was just slightly better in Pr. However, [7] advises that RM's accuracy and prudence is not preset and can be controlled by altering the z value. For the dataset that RM was tested with different z values, it is clear that a high z value (0.9) produces a higher BA and Pr than RDM and a low z value conversely resulted in a BA and Pr much less than RDM's. Based on the results in Table 2, there does not seem to be an obvious correlation between balanced accuracy and prudence. However there seems to be a consistency in that the system with the higher BA also had a higher PA. The ratio of rules to the total number of cases a dataset has does not seem to affect the prudence of either system. The prudence results were expected to be lower for the Physical Action dataset since each rule covers very few cases. The likelihood of a misclassification in such a setting is compounded by the fact that the differences between the rules may be minute. So when a system has proportionally many, almost similar rules, it is likely that some rule may overlap with another, resulting in a lot more misclassifications. For these tests however, this claim was not affirmed.

6 Conclusion

RDM and RM are two PA systems whose accuracy and viability have been demonstrated in different domains [8,11]. These two approaches have not been directly compared previously. This paper presented a preliminary comparison of the two systems using three relatively small datasets. For the smallest and simplest dataset, RM appears to have a higher accuracy and prudence, albeit by a small margin. RDM outperformed RM in the categorical dataset and in the Physical Action dataset, RDM's performance was almost midpoint between RM's optimal setting ($z = 0.9$) and worst setting ($z = 0.01$). The tests conducted for this paper are part of a bigger research project whose aim is to integrate RM and RDM into a single, prudent anomaly detection system. Future tests will use bigger, complex datasets and will use optimal configurations for the two systems.

References

[1] Richards, D.: Two decades of Ripple Down Rules research. The Knowledge Engineering Review 24(2), 159–184 (2009)

[2] Compton, P., Peters, L., Edwards, G., Lavers, T.G.: Experience with Ripple-Down Rules. In: AI 2005, Cambridge (2005)

[3] Edwards, G., Compton, P., Malor, R., Srinivasan, A., Lazarus, L.: PEIRS: a pathologist maintained expert system for the interpretation of chemical pathology reports. Pathology 25(1), 27–34 (1993)

[4] Kang, B., Compton, P., Preston, P.: Multiple Classification Ripple Down Rules: Evaluation and Possibilities. In: 9th Banff Knowledge Acquisition for Knowledge Based Systems Workshop, Banff (1995)
[5] Compton, P., Preston, P., Edwards, G., Kang, B.: Knowledge based systems that have some idea of their limits. CIO 15, 57–63 (1996)
[6] Dazeley, R., Kang, B.: Detecting the Knowledge Boundary with Prudence Analysis. In: Wobcke, W., Zhang, M. (eds.) AI 2008. LNCS (LNAI), vol. 5360, pp. 482–488. Springer, Heidelberg (2008)
[7] Dazeley, R.: To the Knowledge Frontier and Beyond: A Hybrid System for Incremental Contextual-Learning and Prudence Analysis. University of Tasmania, PhD Thesis (2007)
[8] Prayote, A.: Knowledge Based Anomaly Detection, University of New Soth Wales, PhD Thesis (2007)
[9] Prayote, A., Compton, P.: Detecting Anomalies and Intruders. In: Sattar, A., Kang, B.-H. (eds.) AI 2006. LNCS (LNAI), vol. 4304, pp. 1084–1088. Springer, Heidelberg (2006)
[10] Compton, P., Preston, P., Kang, B.: The Use of Simulated Experts in Evaluating Knowledge Acquisition. In: Knowledge Acquisition for Knowldge Based Systems Workshop, Banff (1995)
[11] Dazeley, R., Kang, B.: The Viability of Prudence Analysis. In: The Pacific Rim Knowledge Acquisition Workshop, Hanoi (2008)
[12] Dazeley, R., Kang, B.: Detecting the Knowledge Frontier: An Error Predicting Knowledge Based System. In: Knowledge Acquisition Workshop, Auckland (2004)