RECURSIVE-RULE EXTRACTION ALGORITHM WITH J48GRAFT AND APPLICATIONS TO GENERATING CREDIT SCORES

Yoichi Hayashi[1], Yuki Tanaka[1], Tomohiro Takagi[1], Takamichi Saito[1], Hideaki Iiduka[1]
Hiroaki Kikuchi[2], Guido Bologna[3], Sushmita Mitra[4]

[1]*Department of Computer Science, Meiji University*
Kawasaki 214-8571, Japan

[2]*Department of Frontier Media Science, Meiji University*
Nakano-ku, Tokyo 164-8525, Japan

[3]*Department of Information Technology, University of Applied Sciences of Western Switzerland*
Rue de la prairie 4, 1204 Geneva, Switzerland

[4]*Sushmita Mitra*
Machine Intelligence Unit, Indian Statistical Institute
203 B.T. Road, Kolkata 700 108, India

Abstract

The purpose of this study was to generate more concise rule extraction from the Recursive-Rule Extraction (Re-RX) algorithm by replacing the C4.5 program currently employed in Re-RX with the J48graft algorithm. Experiments were subsequently conducted to determine rules for six different two-class mixed datasets having discrete and continuous attributes and to compare the resulting accuracy, comprehensibility and conciseness. When working with the CARD1, CARD2, CARD3, German, Bene1 and Bene2 datasets, Re-RX with J48graft provided more concise rules than the original Re-RX algorithm. The use of Re-RX with J48graft resulted in 43.2%, 37% and 21% reductions in rules in the case of the German, Bene1 and Bene2 datasets compared to Re-RX. Furthermore, the Re-RX with J48graft showed 8.87% better accuracy than the Re-RX algorithm for the German dataset. These results confirm that the application of Re-RX in conjunction with J48graft has the capacity to facilitate migration from existing data systems toward new concise analytic systems and Big Data.

Keywords: Rule Extraction, Credit Scoring, Re-RX algorithm, J48graft

1 Introduction

Recently, both Big Data and business analytics have increased in importance, and so companies using established data warehouse architectures have to integrate these methods with existing systems based on traditional business intelligence. There has been little discussion about the means of accomplishing this migration, which requires an information architecture that ideally includes all the advantages of existing data warehousing approaches while facilitating the integration of new technologies.

Automated credit score calculations are evidently superior to hand-calculated risk assessments. Advantages include increased objectivity and relia-

bility, as well as reduced costs and labor during the assessment of new credit applications [1]. Nonetheless, the credit evaluation of a human with sufficient expertise can still be superior to an automated assessment. For this reason, and because the finance industry relies on the appropriate prediction of lending risks, research aimed at improving the validity of computerized credit assessments is ongoing [2].

As noted, automated lending risk evaluations are currently imperfect and, in fact, the failure of credit scoring algorithms to identify loan recipients who will eventually default on their loans results in sizable losses on an ongoing basis [3].

Data show that the loss of vast amounts of capital could be prevented by even small increases in the performance of computerized credit assessments. As a result, numerous methods have been considered for automated loan decision making processes, including linear discriminant analysis (LDA), logistic regression (LR) and multiple discriminant analysis (MDA).

A number of alternative approaches to credit score computation have also been developed over the last several decades, including decision trees (DTs) [4], support vector machines (SVMs) [5], neural networks (NNs) [2, 6], classifier [7] and various genetic algorithms (GA) [8]. However, these methods have not been widely adopted since they are more complex and require greater resources while offering less interpretability, although they have been shown to produce significantly better results compared to those obtained from LR, LDA and MDA.

One promising approach to improved credit scoring is rule extraction, whereby a set of simple and comprehensible rules are found to explain the behavior of NNs [6] and SVMs [5, 9], and three methods for the extraction of rules by an NN for the purpose of evaluating credit risks were assessed by Baesens et al. [6]. In addition, several methods have been proposed for the extraction of information from back-propagation neural networks (BPNNs) [10, 11, 12].

The practical application of classification processes typically uses both discrete and continuous data inputs, or so-called mixed data [13], and so rule extraction by Neurorule [14] and similar algorithms requires that the continuous attributes first be discretized. During discretization, the input space is divided into hyper-rectangular regions, each of which corresponds to data samples belonging to a specific class, and is associated with an extracted rule condition [13].

However, some NN rule extraction algorithms, such as GLARE [15] and OSRE [16], do not need to discretize the attributes of continuous input data [10,11], and instead use linear combinations of the appropriate input attributes, incorporating both continuous and discrete attributes, to generate the extracted rules.

Pronounced improvements in the generalization capabilities of artificial neural networks (ANNs)-based learning systems have been demonstrated via the application of ANN ensembles, based on combining the predictions of numerous trained ANNs in a voting process [17].

More recently, improved generalization has been exhibited by BPNN ensembles [18], although an intuitive understanding of the decision-making processes of such ensembles is impossible due to their lack of transparency, significantly restricting the applications of this method [19].

Real-world classification problems usually involve both discrete and continuous input attributes, all of which must be discretized. However, this process may reduce the accuracy of the networks, and hence the accuracy of the extracted rules, because discretization divides the input space into hyper-rectangular regions.

Setiono et al. [13] has proposed a Recursive-Rule Extraction (Re-RX) algorithm for rule extraction from an NN trained to solve classification problems having mixed discrete and continuous input data attributes. This algorithm is similar to other existing rule extraction algorithms in some ways. Setiono et al. [20] has also proposed a preprocessing method for the Re-RX algorithm and a credit card screening application [21]. Utilizing Re-RX, Hayashi et al. [22] presented a very modern approach to the analysis of consumer heterogeneity in the context of eating-out behavior in Taiwan.

Bologna [23] has proposed the Discretized Interpretable Multi-Layer Perceptron (DIMLP) model with rules generated from NN ensembles. This is a special NN model that generates symbolic rules to clarify the knowledge embedded within

connections and activation neurons. Zhou et al. [24] proposed the Rule Extraction from Network Ensemble (REFNE) approach to extract symbolic rules from trained NN ensembles performing classification tasks.

The novel algorithm Re-RX noted above provides a hierarchical, recursive consideration of discrete variables prior to the analysis of continuous data. This algorithm is able to generate classification rules from NNs that have been trained on the basis of discrete and continuous attributes.

Due to its recursive nature, the Re-RX algorithm tends to generate more rules than other rule extraction algorithms, sometimes generating too many rules for practical applications when working with moderately-sized datasets such as Bene1 and Bene2 [6].

In both multi-objective optimization and economics, so-called Pareto optimality is always an important issue. In the case of credit scoring, there is a trade-off between high accuracy and comprehensibility; extracting rules from credit scoring datasets with an increased degree of accuracy generally results in reduced comprehensibility. Conversely, one can also obtain extracted rules with reduced accuracy and more comprehensibility.

Considering the above Pareto optimality, we wished to increase the conciseness of the extracted rules resulting from the Re-RX algorithm, thus mitigating the tendency of this method to generate more rules than other rule extraction algorithms. In this paper, we newly propose the combination of the Re-RX algorithm with the J48graft program [25] as a means of further enhancing the conciseness of the rules extracted by Re-RX. J48graft is an unique variant of C4.5 [4], and is described in detail in Section 3.2.

Herein, we compare the accuracy, comprehensibilty and conciseness obtained from the Re-RX algorithm [13] with that resulting from the proposed combination of Re-RX with J48graft. We report experiments conducted to generate rules for six types of two-class mixed datasets containing discrete and continuous attributes.

When working with the CARD1, CARD2, CARD3, German, Bene1 and Bene2 datasets, Re-RX with J48graft was found to provide more concise rules than were generated by the Re-RX algorithm. In the case of the German, Bene1 and Bene2 datasets, Re-RX with J48graft allowed 43.2%, 37% and 21% reduction in the number of rules, while a 8.87% accuracy increase was seen when using the German dataset.

2   Recursive-Rule Extraction Algorithm: Re-RX Algorithm

Algorithm Re-RX (S_i, D_i, C_i).

Input: A set of data samples S having discrete attributes D and continuous attributes C.

Output: A set of classification rules.

1. Train and prune [26] a neural network by using the dataset S and all of its D and C attributes.

2. Let D' and C' be the sets of discrete and continuous attributes, respectively, still present in the network, and let S' be the set of data samples correctly classified by the pruned network.

3. If $D' = f$, then generate a hyperplane to split the samples in S' according to the values of the continuous attributes C', and stop. Otherwise, by using only the discrete attributes D', generate the set of classification rules R for dataset S'.

4. For each rule Ri generated:
If support$(Ri) > \delta_1$ and error$(Ri) > \delta_2$, then

- Let S_i be the set of data samples that satisfy the condition of rule R_i and let D_i be the set of discrete attributes that do not appear in rule condition R_i.

- If $D_i = f$, then generate a hyperplane to split the samples in S_i according to the values of their continuous attributes C_i, and stop.

- Otherwise, call Re-RX (S_i, D_i, C_i).

The Re-RX algorithm [13] is designed to generate classification rules from datasets that have both discrete and continuous attributes. The algorithm is recursive in nature and generates hierarchical rules.

The rule conditions for discrete attributes are disjointed from those for continuous attributes. The continuous attributes only appear in the conditions of the rules lowest in the hierarchy. The outline of the algorithm is as follows.

The support of a rule is the percentage of samples that are covered by that rule. The support and the corresponding error rate of each rule are checked in step 4. If the error exceeds threshold δ_1 and the support meets the maximum threshold δ_2, then the subspace of this rule is further subdivided by either recursively calling Re-RX when discrete attributes are still not present in the conditions of the rule or by generating a separating axis hyperplane involving only the continuous attributes of the data.

3 Recursive-Rule Extraction Algorithm with J48graft

Herein, we propose combining the Re-RX algorithm with J48graft to enhance the conciseness of classification rules. The conventional Re-RX algorithm uses C4.5 as a decision tree. In the present work, however, we replaced C4.5 with J48graft [25] as the new decision tree. Sections 3.1 and 3.2, below, describe this new algorithm.

3.1 J4.8

J4.8 [27] is a Java-implemented version of C4.5 [4], representing an advanced version of Quinlan's ID3 algorithm [28]. C4.5 is generally referred to as a statistical classifier since the decision trees generated by this algorithm are used for classification. C4.5 functions in a manner very similar to ID3 except that it uses the gain ratio to determine the best target attribute. The C4.5 algorithm also has some advancements relative to ID3, such as the ability to handle numerical attributes by creating a threshold and splitting the data into two sets: data for which the attribute value is above the set threshold and data whose attribute values are less than or equal to this threshold. In addition, this algorithm has the ability to handle attributes with differing costs. Finally, because C4.5 is able to prune the decision tree after creation of the tree, it can potentially decrease the size of the tree and hence reduce the demand for both memory and computation time.

3.2 J48graft

Although simple methods of selecting appropriate trees have typically been considered advisable, tree grafting differs in that this process works on the assumption that similar objects are very likely in the same class. Tree grafting therefore attempts to generate superior classification models at the expense of producing highly complex trees.

Decision trees can undergo grafting as a postprocess meant largely to reclassify parts of the instance space in which there are no training data or there are only misclassified data, as a means of reducing prediction errors. As such, this method identifies leaf reagions that should be pruned and subsquently generates new leaves with novel classifications via a branching out process, necessarily producing a more complex tree. This process only allows branching that avoids the introduction of classification errors into data that have previously been correctly classified. As a result, rather than intoducing errors, the grafting technique eliminates them.

To provide a more efficient means of evaluating the supporting evidence, the C4.5A algorthim was originally introduced by Webb [29]. This algorithm is associated with grafting from the All-Test-But-One-Partition (ATBOP) regions of a leaf, defined as those regions that result from removing all surrounding decision surfaces. Since only the training dataset from this ATBOP region is taken into account for each leaf, the ATBOP approach reduces computational requirements. Since the introduction of C4.5A, the open source data mining software Weka has incorporated the algorithm, terming it J48graft [25].

Pruning can be thought of as the reverse process to grafting because it lowers decsion tree complexity while maintaining a suitable degree of prediction accuracy. In contrast, grafting increases the complexity of the tree. Despite this, Webb [30] determined that the parallel use of these two techniques generates good results. This is perhaps because pruning considers solely instances internal to analyzed leaves (local information), while grafting takes into account information external to the leaves (global information). Thus these approaches are complementary and their combined use generally produces a lower prediction error than their separate applications [30].

3.3 Combining Re-RX with J48graft

The pruning approaches employed in J4.8 and the grafting process that occurs in J48graft [25] both complement and contrast with one another. We believe that the performance of the Re-RX algorithm [13] is greatly affected by the decision tree, and so our goal was to use the grafting techniques in J48graft and Re-RX to improve the accuracy and conciseness of extracted rules. To this end, we replaced J4.8 in the Re-RX algorithm with J48graft.

When combined with Re-RX, the J48graft algorithm is frequently employed to form decision trees in a recursive manner while training Multi-Layer Perceptrons (MLPs) using Back-Propagation (BP). This allows for pruning [26] and thus generates more efficient MLPs for rule extraction.

4 Datasets and Experimental Procedure

4.1 CARD Credit Datasets

The UCI Repository [31] provides a dataset of credit card approvals comprising 690 data samples, each of which has six continuous and nine discrete (that is, categorical or nominal) attributes. The discrete attributes are designated using a binary system and so, in total, the dataset consists of 51 input attributes with two outputs.

Prechelt [32] randomly redistributed this dataset to generate three permutations to serve as benchmarks: CARD1, CARD2 and CARD3, each containing 518 and 172 training and test samples, respectively.

4.2 German Credit Dataset

The UCI Repository's German credit dataset [31] holds 1000 samples, each having 20 attributes, including a class attribute that designates individuals with good (about 700 samples) or bad (300 samples) credit. In the present study, this dataset was apportioned in a random manner such that 70% of the samples were allotted to training and the remaining 30% were used for test purposes.

4.3 Bene1 and Bene2 Credit Datasets

This study also employed the Bene1 and Bene2 datasets [6] generated by Benelux-based major financial institutions, which summarize consumer credit application data. In keeping with standard banking practice, customers are flagged as a poor risk in these datasets if they have been in payment arrears longer than 90 days at any point. These voluminous datasets were divided such that approximately two thirds of the samples were applied for training and one third for test purposes.

4.4 Experimental Setup

The training sets within each database were used to train NNs and to extract rules. During this process, one input unit was created in the NN for each continuous attribute contained in the dataset, while either thermometer or dummy variable encoding converted discrete attributes into a binary input string [33]. The datasets used to assess two different rule extraction algorithms are summarized in Table 1. To deal with the class imbalanced datasets, the area under the receiver operating characteristic (ROC) curve (AUC) was used as an appropriate performance evaluator not including class distribution or misclassification costs [1].

Table 1. Characteristics of Datasets Used in Credit-risk Evaluation

	Dataset Size	Input Total	Input Continuous	Input Discrete
CARD1	690	51	6	45
CARD2	690	51	6	45
CARD3	690	51	6	45
German	1000	20	7	13
Bene1	3123	27	13	14
Bene2	7190	28	18	10

5 Experimental Results

We trained six types of two-class mixed datasets [6, 31, 32] using both the Re-RX and Re-RX with J48graft algorithms. The resulting accuracies and number of extracted rules were determined for each test dataset, as well as the average number of antecedents in each extracted rule. Tables 2–4 summarize the results obtained from the Re-RX algo-

Table 2. Performance of training and test datasets for CARD1 and CARD2 by the Re-RX algorithm (averaged over ten runs)

	CARD1 Training (%)	Test	# Rules	Ave. # Ante.	CARD2 Training (%)	Test	# Rules	Ave. # Ante.
Re-RX	-	89.53	9	3.44	-	86.63	9	1.56

Table 3. Performance of training and test datasets for CARD3 and German by the Re-RX algorithm (averaged over ten runs)

	CARD3 Training (%)	Test	# Rules	Ave. # Ante.	German Training (%)	Test	# Rules	Ave. #. Ante.
Re-RX	-	88.95	6	2.33	-	71.66±0.88	24.9	9.29

Table 4. Performance of training and test datasets for Bene1 and Bene2 by the Re-RX algorithm (averaged over ten runs)

	Bene1 Training (%)	Test	# Rules	Ave. # Ante.	Bene2 Training (%)	Test	# Rules	Ave. # Ante.
Re-RX only	-	71.40±0.51	49.9	9.25	-	72.23±0.42	50.20	7.57

Table 5. Performance of training and test datasets for CARD1 and CARD2 by the Re-RX with J48graft (averaged over ten runs)

	CARD1 Training (%)	Test	# Rules	Ave. # Ante.	CARD2 Training (%)	Tes t	# Rules	Ave. # Ante.
Re-RX with J48graft	89.09±0.49	88.98±1.11	**6.3**	3.15	87.72±0.67	87.58±0.18	7.2	3.68

Table 6. Performance of training and test datasets for CARD3 and German by the Re-RX with J48graft (averaged over ten runs)

	CARD3 Training (%)	Test	# Rules	Ave. # Ante.	German Training (%)	Test	# Rules	Ave. # Ante.
Re-RX with J48graft	89.48±3.32	89.39±1.88	5.7	2.73	81.32±1.02	**80.53±0.88**	**14.4**	**5.36**

Table 7. Performance of training and test datasets for Bene1 and Bene2 by the Re-RX with J48graft (averaged over ten runs)

	Bene1 Training (%)	Test	# Rules	Ave. # Ante.	Bene2 Training (%)	Test	# Rules	Ave. # Ante.
Re-RX with J48graft	74.42±0.34	**74.15±0.38**	**31.5**	9.17	75.73±3.00	**75.52±2.62**	**39.9**	7.15

rithm, while Tables 5–7 show results from Re-RX with J48graft. In Tables 2–7, the bold numbers indicate the highest test accuracies or the fewest number of rules of all the rules extracted.

6   Discussion of Experimental Results

In the case of each database, the Re-RX with J48graft generated a more concise set of extracted rules than the Re-RX algorithm.

As confirmed by the number of rules and the average number of antecedents of extracted rules in Tables 6 and 7, the use of J48graft is evidently an effective means of obtaining improved conciseness and comprehensibility when working with larger and/or complex datasets such as German and Bene2.

For German, Bene1 and Bene2 datasets, we independently implemented the Re-RX algorithm in Java from the authors [13] so that we obtained the results for test datasets by averaged over ten runs.

The Re-RX algorithm exhibits 71.66±0.88 accuracy when working with the German dataset and extracts 24.9 rules.

The Re-RX algorithm also showed 71.40±0.51 accuracy with Bene1 dataset and extracts 49.9 rules.

In the same manner, the Re-RX algorithm showed 72.23±0.42 accuracy with Bene2 dataset and extracts 50.2 rules.

In the case of German dataset, Re-RX with J48graft gave the accuracy of 80.53±0.88 and extracted 14.4 rules. Clearly, the Re-RX with J48graft is considerably more accurate and concise compared to that of the Re-RX algorithm.

In the case of Bene1 and Bene2 datasets, Re-RX with J48graft gave slightly better accuracy and considerably more concise rules compared to that of the Re-RX algorithm.

7   Conclusion

This paper proposes that the Recursive-Rule Extraction algorithm can be combined with J48graft. Experiments were conducted to obtain classification rules for six two-class mixed datasets to establish the accuracy, comprehensibility and conciseness of this new approach relative to the Re-RX algorithm.

Working with the German, Bene1 and Bene2 datasets, the proposed Re-RX with J48graft combination clearly showed a more effective set of extracted rules.

Needless to say, if we can accomplish Pareto optimization, we will be able to obtain the best possible rule extraction process. Ideally, we hope to extend the Pareto optimization curve to obtain a wider viable region that provides improvements in both accuracy and comprehensibility.

In future work, we will explore additional decision trees that are more compatible with the Re-RX algorithm and that generate results closer to the Pareto optimal accuracies for various datasets.

References

[1] Marqus, A.I., Garca, V., & Snchez, J.S. On the suitability of resampling techniques for the class imbalance problem in credit scoring. Journal of the Operational Research Society 64, pp. 1060–1070, 2013.

[2] Zhao Z., Xu, S., Kang, B. H., Kabir, M. M. J., & Liu, Y. Investigation and improvement of multi-

layer perceptron neural networks for credit scoring Expert Systems with Applications 42, pp. 3508–3516, 2015

[3] Finlay, S. M. Multiple classifier architectures and their applications to credit risk assessment. European Journal of Operational Research 210, pp. 368–378, 2011.

[4] Quinlan, J.R. Programs for Machine Learning. Morgan Kaufmann Series in Machine Learning, San Mateo, CA, 1993, Morgan Kaufman.

[5] Martens, D., Baesens, B., Van Gestel, T., & Vanthienen, J. Comprehensible credit scoring models using support vector machines. European Journal of Operational Research 183, pp. 1497–1488, 2007.

[6] Baesens, B., et al. Using neural network rule extraction and decision tables for credit-risk evaluation. Management Science 49, No. 3, pp. 312–329, 2004.

[7] Abellan, J., and Mantas, C. Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring. Expert Systems with Applications, pp. 3825–3830, 2014.

[8] Finlay, S. M. Multiple classifier architectures and their applications to credit risk assessment. European Journal of Operational Research 210, pp. 368–378, 2011.

[9] Braket, N., and Bradely, A.P. Rule extraction from support vector machine: a review. Neurocomputing 74, pp. 178–190, 2010.

[10] Setiono, R., and Liu, H. Neurolinear: From neural networks to oblique decision rules. Neurocomputing 17, No. 1, pp. 1–24, 1997.

[11] Setiono, R. and Liu, H. A connectionist approach to generating oblique decision trees. IEEE Trans. Syst., Man, Cybern. B, Cybern. Vol. 29, No. 3, pp. 440–444, Jun. 1999.

[12] Setiono, R., Baesens, B. & Mues, C. A note on knowledge discovery using neural Setiono networks and its application to credit card screening. European Journal of Operational Research 192, pp.326-332, 2009.

[13] Setiono, R., et al. Recursive neural network rule extraction for data with mixed attributes. IEEE Trans. Neural Netw. 19, No. 2, pp. 299–307, 2008.

[14] Setiono, R., and Liu, H. Symbolic representation of neural networks. IEEE Computer 29, No. 3, pp. 71–77, 1996.

[15] Gupta, A, Park, S. and Lam, S.M. Generalized analytic rule extraction for feedforward neural networks, IEEE Trans. Knowledge and Data Engineering, 11, pp.985-991, 1999.

[16] Etchell, T.A. and Lisboa, J.P.G., Orthogonal search-based rule extraction (OSRE) for trained neural-networks: A practical and efficient approach, IEEE Trans. Neural Networks 17, pp.374-384, 2006.

[17] Hansen, L.K., and Salamon, P., Neural network ensembles. IEEE Trans. Patter Analysis and Machine Intelligence 12, pp. 993–1001, 1990.

[18] Igelnik, S., Pao, Y.-H., LeClair, S. R., and Shen, C. Y. The ensemble approach to neural-network learning and generalization. IEEE Trans. Neural Networks 10, pp. 19–30, 1999.

[19] Liao, J.-J., Shih, C.-H., Chen, T.-F., and Hsu, M.-F. An example-based model for two-class imbalanced financial problem. Economic Modelling 37, pp. 175–183, 2014.

[20] Setiono R. et al., Rule extraction from minimal neural networks for credit card screening, Inter. J. of Neural Systems., Vol. 21, No. 4, pp. 265–276, 2011.

[21] Setiono R. et al., A note on knowledge discovery using neural networks and its application to credit card screening, European J. Operational Research, Vol. 192, pp. 326–332, 2009.

[22] Hayashi Y. et al., Understanding consumer heterogeneity: A business intelligence application of neural networks, Knowledge-Based Systems, Vol. 23, No. 8, pp. 856–863, 2010.

[23] Bologna G., Is it worth generating rules from neural network ensemble?, J. of Applied Logic, Vol. 2, pp. 325–348, 2004.

[24] Zhou, Z.-H. Extracting symbolic rules from trained neural network ensembles. AI Communications 16, pp. 3–15, 2003.

[25] http://fiji.sc/javadoc/weka/classifiers/trees/J48graft.html

[26] Setiono R. et al., A penalty-function approach for pruning feedforward neural networks, Neural Comp., Vol. 9, No. 1, pp. 185–204, 1997.

[27] Witten, I.H. and Frank, E., Data Ming: Practical Machine Learning Tools With Java Implementations. San Francisco, CA: Morgan Kaufmann, 1999.

[28] Quinlan J.R., Induction of decision trees, Machine Learning, Vol.1, pp.81-106, 1986.

[29] Webb. G.I., Decision Tree Grafting from the All-Tests-But-One Partition, in Proc. 16th International Joint Conference on Artificial Intelligence (IJCAI), Vol. 2, pp. 702–707, 1999.

[30] Webb, G. I., Decision Tree Grafting, Learining, IJ-CAI'97 Proceedings of the 15th International Conference on Artificial Intelligence, Vol.2, pp. 846-885, 1997.

[31] Frank, A. & Asuncion, A. University of California. Irvine Machine Learning Repository. http://archive.ics.uci.edu/ml/, 2010

[32] Prechelt, L. Proben1 — A set of benchmarks and benchmarking rules for neural network training algorithms, Technical Report 21/94, Fakultt fr Informatik, Universitt Karlsruhe, Germany. Anonymous ftp available from ftp://pub/papers/techreport/1994/1994-21.ps.gz on ftp.ira.uka.de , 1994.

[33] Smith, M. Neural Networks for Statistical Modeling, New York: Van Nostrand Reinhold, 1993.

Prof. **Yoichi Hayashi** received the BE in Management Science and ME degree in Systems Engineering and Dr. Eng. degree in Systems Engineering, all from the Tokyo University of Science, Tokyo, Japan, in 1979, 1981, and 1984, respectively. Then, joined Information Promotion Agency (IPA) as a project reader of DBMS management technology. IPA has been remarkably expanding the scale as a big IT research organization of the Ministry of International Trade and Industry, Japan. In 1986, he joined the Computer and Information Sciences department of Ibaraki University, Japan, as an Assistant Professor and was a Visiting Professor of Computer Science Department of the University of Alabama at Birmingham and the University of Canterbury, New Zealand respectively. Since 1996, he is a full Professor of Computer Science department at Meiji University, Tokyo, Japan. In 2004-2006, he served the chairman and full professor of his department. He is the author of over 230 academic papers published in academic journals and major international conference proceedings in computer science. His current research interests include Big Data analytics, high accuracy data mining, neural network ensemble, rule extraction, hybrid classifier, bioinformatics and medical informatics. He is the Associate Editor of Artificial Intelligence in Medicine and was an Associate Editor of the IEEE Transactions on Fuzzy Systems, the Action Editor of Neural Networks and Journal of Advanced Computational Intelligence and others. He is a senior member of the IEEE since 2000.

Mr. **Yuki Tanaka** is currently a graduate student of Computer Science at Meiji University. His research interests include high accuracy Big Data mining algorithm. He presented some papers on high accuracy rule extraction algorithm at IJCNN, ICANN and ICONIP**.**

Tomohiro Takagi received the Doctor of Engineering degree from the Tokyo Institute of Technology in 1983. He was an EECS research fellow at the University of California Berkeley from 1983-84. From 1987 to 1997 he worked in the central research laboratory and corporate multimedia promotion division at Matsushita Electric Industrial Co., LTD. He was also a deputy director at the Laboratory for International Fuzzy Engineering Research, which was a national project supported by the Ministry of International Trade and Industry, from 1991 to 1993. Since 1998 he has been affiliated with the Department of Computer Science of Meiji University. Professor of Computer Science Course in graduate school of Science and Technology. The International Fuzzy Systems Association fellow. IEEE Computational Intelligence Society Fuzzy Systems Pioneer Award.

Takamichi Saito received the Ph.D. degree from the Tokyo University of Science in 2001. Currently, he is an Associate Professor at Meiji University, Japan.

Hideaki Iiduka received a B.S. degree from Tokyo University of Science, Tokyo, Japan, in 2000, and M.S. and D.S. degrees from Tokyo Institute of Technology, Tokyo, Japan, in 2002 and 2005, respectively. From 2005 to 2007, he was a Research Assistant in the Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Tokyo, Japan. From 2007 to 2008, he was a Research Fellow (PD) of Japan Society for the Promotion of Science. From October 2008 to March 2013, he was an Associate Professor in the Network Design Research Center, Kyushu Institute of Technology, Tokyo, Japan. Since April 2013, he has been an Associate Professor in the Department of Computer Science, School of Science and Technology, Meiji University, Kanagawa, Japan. His research field is optimization theory and its applications to mathematical information science. He is a member of the SIAM and MOS.

Hiroaki Kikuchi, he received B. E., M. E. and Ph.D. degrees from Meiji University in 1988, 1990 and 1994. After he was working in Fujitsu Laboratories Ltd. in 1990, he had worked in Tokai University from 1994 through 2013. He is currently a professor in at Department of Frontier Media Science, School of Interdisciplinary Mathematical Sciences, Meiji University. He was a visiting researcher of the school of computer science, Carnegie Mellon Univer-

sity in 1997. His main research interests are network security, cryptographical protocol, privacy-preserving data mining, and fuzzy logic. He received the Best Paper Award for Young Researcher of Japan Society for Fuzzy Theory and Intelligent Informatics in 1990, the Best Paper Award for Young Researcher of IPSJ National Convention in 1993, the Best Paper Award of Symposium on Cryptography and Information Security in 1996, the IPSJ Research and Development Award in 2003, the Journal of Information Processing (JIP) Outstanding paper Award in 2010, and the IEEE AINA Best Paper Award in 2013. He is a member of the Institute of Electronics, Information and Communication Engineers of Japan (IEICE), the Information Processing Society of Japan (IPSJ), the Japan Society for Fuzzy Theory and Systems (SOFT), IEEE and ACM. He is a director of IPSJ since 2013. He receives IPSJ Fellow.

Guido Bologna is senior lecturer at the University of Applied Sciences of Western Switzerland, and senior scientist with the Computer Vision and Multi-Media Lab of the University of Geneva, Switzerland. He received a Ph.D. in artificial intelligence in 1998 at the University of Geneva for the development of a new neural network model from which symbolic rules are easily extracted. Subsequently, he worked as a researcher at the Queensland University of Technology, Brisbane; at the National University of Singapore and at the Swiss Institute of Bioinformatics. His research interests focus on machine learning, computer vision and multimodal interfaces for blind users. He is currently investigating mobility aids for blind users and for the elderly, as well as rule extraction from neural network ensembles. He has authored or co-authored over 80 full papers in conference proceedings, book chapters and journal articles.

Sushmita Mitra is a full Professor at the Machine Intelligence Unit (MIU), Indian Statistical Institute, Kolkata. She was the Head of MIU during the period 2010 to 2012. From 1992 to 1994 she was in the RWTH, Aachen, Germany as a DAAD Fellow. She was a Visiting Professor in the Computer Science Departments of the University of Alberta, Edmonton, Canada in 2004, 2007; Meiji University, Japan in 1999, 2004, 2005, 2007; and Aalborg University Esbjerg, Denmark in 2002, 2003. Dr. Mitra received the National Talent Search Scholarship (1978-1983) from NCERT, India, the University Gold Medal in 1988, the IEEE TNN Outstanding Paper Awardin 1994 for her pioneering work in neuro-fuzzy computing, and the CIMPA-INRIA-UNESCO Fellowship in 1996.

Dr. Mitra has guest edited special issues of several journals, is an Associate Editor of ``IEEE/ACM Trans. on Computational Biology and Bioinformatics'', ``Information Sciences'', ``Neurocomputing'', and is a Founding Associate Editor of ``Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery (WIRE DMKD)''. She has more than 75 research publications in referred international journals. Dr. Mitra is a Fellow of the IEEE, and Fellow of the Indian National Academy of Engineering and The National Academy of Sciences, India. She is an IEEE CIS Distinguished Lecturer for the period 2014-2016. Her current research interests include data mining, pattern recognition, soft computing, image processing, and Bioinformatics.