Proceedings of the 2007 IEEE Symposium on
Computational Intelligence and Data Mining (CIDM 2007)

On Obtaining Fuzzy Rule Base from Ensemble of Takagi-Sugeno Systems

Marcin Korytkowski, Leszek Rutkowski, *Fellow, IEEE*,
Rafał Scherer, *Member, IEEE*, Grzegorz Drozda

***Abstract*—Takagi-Sugeno fuzzy systems are very common learning systems. The paper is about building classification ensembles from them and merging resulting rule bases. When merged, the rule base is more intelligible and easier to process. The merging is possible thanks to a modification of TS systems. Numerical simulations show that the modified systems perform very well.**

I. INTRODUCTION

FUZZY systems are established tools for solving various problems [5], [14], [17]. They can be divided into Mamdani type, logical and Takagi-Sugeno (TS) fuzzy systems. In TS fuzzy systems rule consequents are functions of system inputs. To further improve fuzzy system performance, they can be combined in larger ensembles, so that the model robustness and accuracy is nearly always improved, comparing to single-model solutions. Combined systems are developed under different names: blending, combining models, bundling, ensemble of classifiers, or committee of experts. Various known classifiers [1], [4], [7], [12] can be combined at the level of features, data subsets, using different classifiers or different combiners [6], [21], [22]. Popular methods are bagging and boosting [3], [11], [13], [16], [18], [19] which are meta-algorithms for learning different classifiers. They assign weights to learning samples according to their performance on earlier classifiers in the ensemble. Thus subsystems are trained with different datasets.

There are many variations of boosting with the most known – AdaBoost, where every learning vector has a weight assigned. According to the AdaBoost algorithm, consecutive classifiers should be learned with the greatest impact from the samples with the highest weight values. These weights have influence on the learning process by changing the learning rate.

In the paper we develop a method for merging Takagi-Sugeno rule bases from several neuro-fuzzy systems constituting an ensemble trained by AdaBoost and the backpropagation algorithm. The method leads to the merged neuro-fuzzy system characterized by interpretability and possibility to reduce its size. The method is the continuation of earlier work of the authors on boosting ensembles of various fuzzy systems [8]–[10].

In the next section we describe the AdaBoost algorithm and in Section III TS fuzzy systems constituting the boosting ensemble. In Section IV we develop a new method allowing merging fuzzy TS rule bases and in Section V we verify the method on a well known data benchmark.

II. ADABOOST ALGORITHM

Boosting concept concerns creating an ensemble of arbitrary classifiers trained on slightly modified datasets. In the paper we focus on one version of boosting learning, i.e. AdaBoost algorithm.

Let us denote the l-th learning vector by $\mathbf{z}^l = [x_1^l, ..., x_n^l, y^l]$, $l = 1...m$ is the number of a vector, n is a size of input vector $\mathbf{x}$, and y is the learning class label. Weights, assigned to learning vectors, have to fulfill the following conditions

$$\text{(i)}\ 0 < d^l < 1 , \tag{1}$$

$$\text{(ii)}\ \sum_{l=1}^{m} d^l = 1 . \tag{2}$$

The weight d^l is the information how classifiers learned in consecutive steps of an algorithm for a given input vector $\mathbf{x}^l$. Vector $\mathbf{d}$ for all input vectors is initialized according to the following equation

$$d_t^l = \frac{1}{m}, \text{ for } t = 0 , \tag{3}$$

where t is the number of a boosting iteration (and a number of a classifier in the ensemble). Of course, there exist a lot of AdaBoost modifications for designing and training non-binary classifiers (e.g. [16]). Let $\{h_t(\mathbf{x}) : t = 1,...,T\}$ denotes a set of hypotheses obtained in consecutive steps t of the algorithm being described. For simplicity we limit our problem to a binary classification (dichotomy), i.e. y={-1,1} or $h_t(\mathbf{x}) = \pm 1$. Similarly to learning vectors weights, we assign a weight c_t for every hypothesis, such that

$$\text{(i).}\ 0 < c_t\ , \tag{4}$$

This work is supported by the Foundation for Polish Science (Professorial Grant 2005-2008), and the Polish State Committee for Scientific Research (Grant Nr T11C 04827 and Grant T11A Nr 01427).

Marcin Korytkowski is with the Department of Computer Engineering at Częstochowa University of Technology, al. Armii Krajowej 36, 42-200 Częstochowa, Poland and with Olsztyn Academy of Computer Science and Management, ul. Artyleryjska 3c, 10-165 Olsztyn, Poland, (e-mail: marcink@kik.pcz.czest.pl).

Leszek Rutkowski and Rafał Scherer are with the Department of Computer Engineering at Częstochowa University of Technology, al. Armii Krajowej 36, 42-200 Częstochowa, Poland and with the Department of Artificial Intelligence at Academy of Humanities and Economics, ul. Rewolucji 1905 nr 64, 90-222 Łódź, Poland (e-mail: lrutko@kik.pcz.czest.pl, rafal@ieee.org).

Grzegorz Drozda is with Faculty of Mathematics and Computer Science, University of Warmia and Mazury in Olsztyn, ul. Zolnierska 14, 10-561 Olsztyn, Poland (e-mail: gdrozd@matman.uwm.edu.pl).

$$\text{(ii). } \sum_t c_t = 1 . \tag{5}$$

Now in the AdaBoost algorithm we repeat steps 1-4 for $t = 1, \ldots, T$:

1. Create hypothesis h_t and train it with a data set with respect to a distribution d_t for input vectors.
2. Compute the classification error ε_t of a trained classifier h_t according to the formula
$$\varepsilon_t = \sum_{l=1}^{m} d_t^l (z^l) I(h_t(\mathbf{x}^l) \neq y^l) , \tag{6}$$
where I is the indicator function
$$I(a \neq b) = \begin{cases} 1 & \text{if} \quad a \neq b \\ 0 & \text{if} \quad a = b \end{cases} . \tag{7}$$
If $\varepsilon_t = 0$ or $\varepsilon_t \geq 0.5$, finish the algorithm.
3. Compute the value
$$\alpha_t = 0.5 \ln \frac{1 - \varepsilon_t}{\varepsilon_t} . \tag{8}$$
4. Modify weights for learning vectors according to the formula
$$d_{t+1}(\mathbf{z}^l) = \frac{d_t(\mathbf{z}^l) \exp\{-\alpha_t \mathbf{I}(h_t(\mathbf{x}_l) = y^l)\}}{N_t} , \tag{9}$$
where N_t is a constant such that
$$\sum_{l=1}^{m} d_{t+1}(\mathbf{z}^l) = 1 . \tag{10}$$

To compute the overall output of the ensemble of classifiers trained by AdaBoost algorithm the following formula is used

$$f(\mathbf{x}) = \sum_{t=1}^{T} c_t h_t(\mathbf{x}) , \tag{11}$$

where

$$c_t = \frac{\alpha_t}{\sum_{t=1}^{T} |\alpha_t|} \tag{12}$$

is classifier importance for a given training set. The above is a meta-learning algorithm and do not determine the way of learning for classifiers in the ensemble.

III. Takagi-Sugeno Fuzzy Systems Used in the Ensemble

We use Takagi-Sugeno fuzzy systems with linear consequents as classifiers in the ensemble. In general case fuzzy rules in TS systems take the following form

$$\begin{aligned} &R^k : \text{IF } x_1 \text{ is } A_1 \text{ AND} \ldots \text{AND } x_n \text{ is } A_n \\ &\text{THEN } y = f(x_1, x_2, \ldots, x_n) \end{aligned} , \tag{13}$$

where $k = 1, \ldots, K$ is the rule number. The inputs are denoted $x_i, i = 1, \ldots, n$. The function used in the paper is linear

$$\begin{aligned} &R^k : \text{IF } x_1 \text{ is } A_1 \text{ AND} \ldots \text{AND } x_n \text{ is } A_n) \\ &\text{THEN } y = \ c_0 + c_1 x_1 +, \ldots, c_n x_n \end{aligned} . \tag{14}$$

We use antecedent fuzzy sets which are represented by Gaussian functions

$$\mu_A(x_i) = \exp\left(-\left(\frac{x_i - \bar{x}_i^k}{\sigma_i^k} \right)^2 \right), \tag{15}$$

where σ_i^k and $\bar{x}_i^k$ are respectively width and center of the function for the fuzzy set in the k-th rule and i-th input. After presenting input signal $\bar{\mathbf{x}} = [\bar{x}_1, \ldots, \bar{x}_n]$, we compute the output of a TS system

$$\bar{y} = \frac{\sum_{k=1}^{K} y^k \, \underset{i=1}{\overset{n}{T}} \left(\mu_{A_i^k}(\bar{x}_i) \right)}{\sum_{k=1}^{K} \underset{i=1}{\overset{n}{T}} \left(\mu_{A_i^k}(\bar{x}_i) \right)} . \tag{16}$$

In the experiments we use T-norm operation realized by product operation (Fig. 1). In case of an ensemble of classifiers the output of a single classifier is

$$h_t = \frac{\sum_{k=1}^{K} \bar{y}^k \tau^k}{\sum_{k=1}^{K} \tau^k} , \tag{17}$$

where τ^k is an activation level of the k-th rule ($k = 1, \ldots, K$) and is defined by

$$\tau^k = \prod_{i=1}^{n} \mu_i^k (x_i) , \tag{18}$$

where $\mu_i^k(x_i)$ is the antecedent membership function of the Gaussian type and $\bar{y}^k$ is a consequent singleton fuzzy set.

We train the structures by the backpropagation algorithm considering boosting learning sample weights d_t^l.

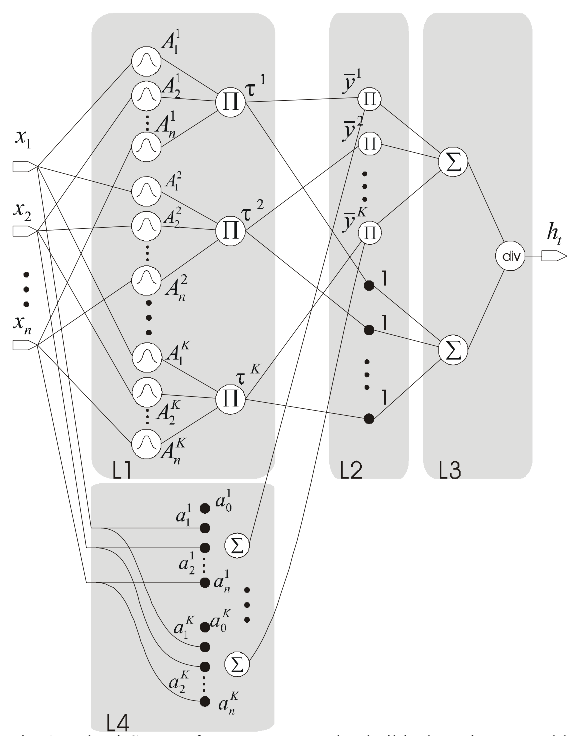

Fig. 1. Takagi-Sugeno fuzzy system used to build a boosting ensemble.

IV. TRANSFORMING TAKAGI-SUGENO ENSEMBLE INTO ONE SYSTEM

In boosting algorithm overall output is combined from all classifiers in the ensemble by (11). In case of fuzzy classifiers (15) the overall output becomes

$$f(\overline{\mathbf{x}}) = \sum_{t=1}^{T} \left(c_t \frac{\sum_{k=1}^{K} y^k \, \overset{n}{\underset{i=1}{T}} \left(\mu_{A_i^k}(\overline{x}_i) \right)}{\sum_{k=1}^{K} \overset{n}{\underset{i=1}{T}} \left(\mu_{A_i^k}(\overline{x}_i) \right)} \right). \tag{19}$$

Index t in (19) is omitted for clarity. Our goal is to make one fuzzy system from all systems constituting the ensemble. It is impossible to bring (19) to a common denominator, thus we can not treat fuzzy rule bases from consecutive systems ($t = 1,...,T$) as one rule base.

We modify the TS systems in the ensemble by changing denominators in (19) to fulfill the following normalizing constraint

$$\forall t, \sum_{k=1}^{K} \overset{n}{\underset{i=1}{T}} \left(\mu_{A_i^k}(\overline{x}_i) \right) = 1 . \tag{20}$$

To achieve this, we propose to add special normalizing layer L1A, see Figure 2. Thus we can remove the lower sum in the layer L3, and (19) is reduced to

$$f(\overline{\mathbf{x}}) = \sum_{t=1}^{T} \left(c_t \sum_{k=1}^{K} \overline{y}_t^k \cdot \overset{n}{\underset{i=1}{T}} \left(\mu_{A_i^k}(\overline{x}_i) \right) \right). \tag{21}$$

Such a system can be trained by the backpropagation algorithm in a boosting ensemble of classifiers. Then all fuzzy rule bases can be joined together to obtain one comprehensive rule base. Such rule base can be post processed, for example can be reduced to be more intelligible and transparent.

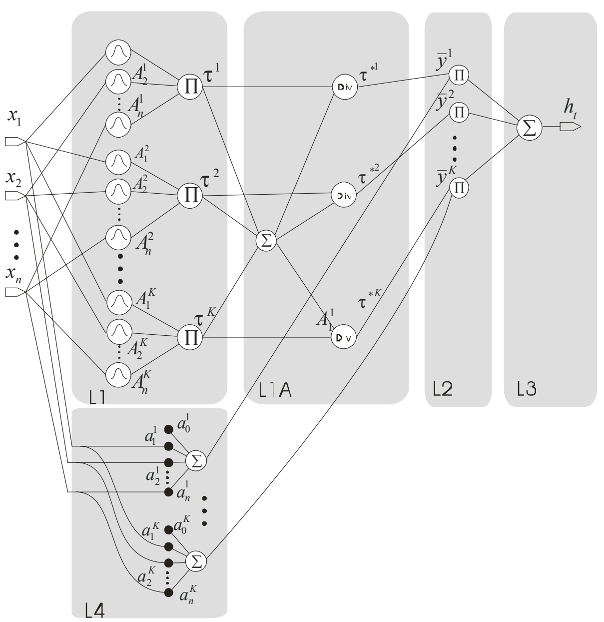

Fig. 2. Single Takagi-Sugeno fuzzy system with normalized rule firing strengths.

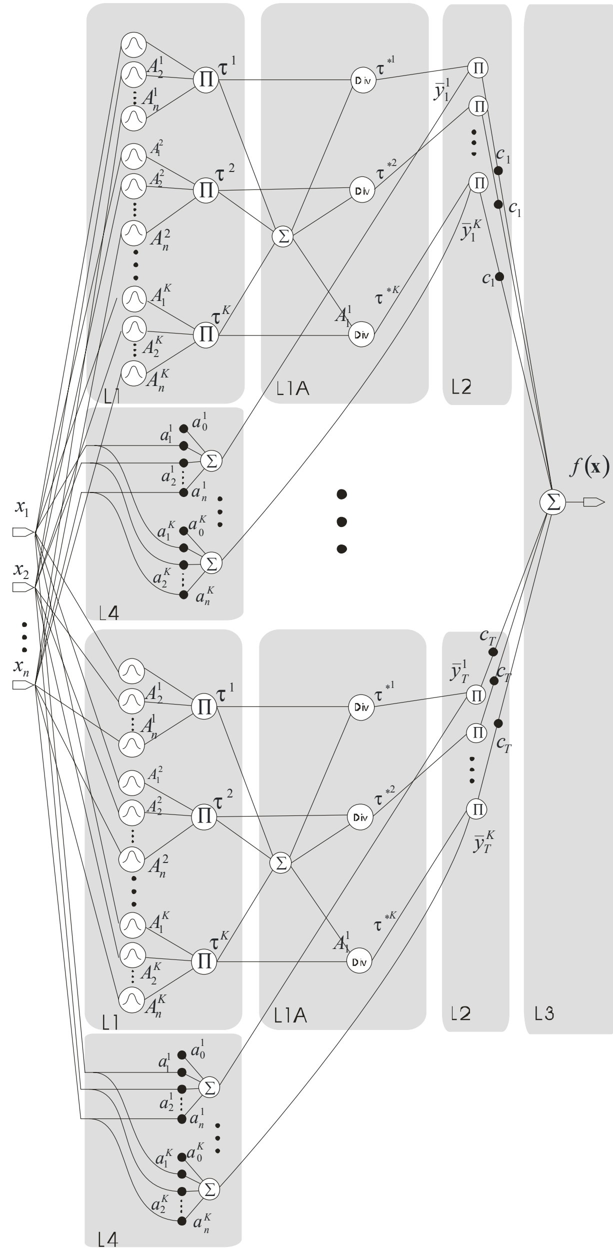

Fig. 3. Modified Takagi-Sugeno fuzzy systems combined into one system after boosting learning. The modification concerns normalizing rule activation levels. Index t denoting classifier number is omitted in layers L1, L1A, L4.

In the new resulting fuzzy system with one rule base, fuzzy rules have weights c_t given by (12). Fuzzy systems with rule weights are common in the literature (e.g. [15]), but the system presented in the paper originates from boosting learning . This system has several advantages:

(i). it is possible to interpret the resulting rule base,

(ii). the merged system can be fine-tuned, regarding boosting as a method for initial choosing of parameters,

(iii). the resulting rule base can be further simplified and reduced.

V. SIMULATIONS

The new method was tested on an ensemble of Takagi-Sugeno systems T-norm and Cartesian product operations realized by product operation with normalized rule activation levels (see Section IV). We used the Wisconsin Breast Cancer Database [2], which consists of 699 instances of binary classes (benign or malignant type of cancer). Classification is based on 9 features (clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, mitoses). From the data set, 205 instances were taken into testing data and 16 instances with missing features were removed. The three first systems in the ensemble have 3 fuzzy rules and the last two systems have 6 fuzzy rules. The classification accuracy was 99.6%. Detailed errors and subsystem parameters are described in Table I.

TABLE I
WISCONSIN BREAST CANCER DATABASE RESULTS FOR THE PROPOSED METHOD

Classifier number	Number of fuzzy rules	MSE learning error	Boosting error, Eq. (6)
1	3	0.29	0.45
2	5	0.13	0.17
3	7	0.21	0.19
4	7	0.06	0.19

VI. CONCLUSION

Classification accuracy is nearly always improved after combining many systems. One of the most popular methods of multiple classification is boosting. In every method of fuzzy ensemble classification we obtain several rule bases. They can not be joined in one rule base unless we normalize them using the method proposed in the paper. We constitute an ensemble of linear Takagi-Sugeno fuzzy systems modified to make possible merging rule bases. The modification relies on changing rule activation so as they sum to unity. Thanks to this it is possible to merge all rules as they have the same importance. Having one rule-base is very convenient in terms of interpretability and possibility to reduce its size. Numerical experiments on a well known benchmark showed the ensembles of modified TS fuzzy systems are able to learn very well.

REFERENCES

[1] C.M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Inc., New York, NY, 1995.

[2] C.L.Blake, C.J. Merz, *UCI Repository of machine learning databases*, [www.ics.uci.edu/ ~mlearn/MLRepository.html], Irvine, University of California, Department of Information and Computer Science, 1998.

[3] L. Breiman. *Bias, variance, and arcing classifiers. Technical Report 460*, Statistics Department, University of California, July 1997.

[4] R.O. Duda, P.E. Hart, D.G. Stork, *Pattern Classification* (2nd Edition), Wiley 2000.

[5] R. J.-S. Jang, C.-T. Sun, E. Mizutani, *Neuro-Fuzzy and Soft Computing, A Computational Approach to Learning and Machine Intelligence*, Prentice Hall, Upper Saddle River 1997.

[6] D.A. Jiménez, N. Walsh, "Dynamically Weighted Ensemble Neural Networks for Classification", *Proceedings of the 1998 International Joint Conference on Neural Networks (IJCNN)*, Anchorage, 1998.

[7] M. Korytkowski, M. Gabryel, A. Gawęda, "Recursive Probabilistic Neural Networks", *Lecture Notes in Computer Science,* Vol. 3070, Springer-Verlag Heidelberg 2004, pp. 626-631.

[8] M. Korytkowski, R. Nowicki, L. Rutkowski, R. Scherer, "Combining Logical-Type Neuro-fuzzy Systems", *Lecture Notes in Artificial Intelligence*, Vol. 4029, 2006, pp. 244-253.

[9] M. Korytkowski, R. Nowicki, L. Rutkowski, R. Scherer, "Merging Ensemble of Neuro-fuzzy Systems", in *Proc.* 2006 IEEE International Conference on Fuzzy Systems, IEEE World Congress on Computational Intelligence, Vancouver, BC, Canada, 2006.

[10] M. Korytkowski, L. Rutkowski, R. Scherer, "On Combining Backpropagation with Boosting", in *Proc. 2006 International Joint Conference on Neural Networks, IEEE World Congress on Computational Intelligence*, Vancouver, BC, Canada, 2006.

[11] L.I. Kuncheva, *Combining Pattern Classifiers, Methods and Algorithms*, John Wiley & Sons 2004.

[12] L. I. Kuncheva, *Fuzzy Classifier Design*, Physica Verlag, Heidelberg, New York, 2000.

[13] R. Meir and G. Rätsch, "An introduction to boosting and leveraging", in *Advanced Lectures on Machine Learning, LNAI 2600,* edited by S. Mendelson and A. Smola, Springer, 2003, pp. 119-184.

[14] D. Nauck, F. Klawon, R. Kruse, *Foundations of Neuro-Fuzzy Systems*, Chichester, U.K., John Wiley, 1997.

[15] H. Ischibuchi, T. Nakashima, "Effect of Rule Weights in Fuzzy Rule-Based Classification Systems", *IEEE Transactions on Fuzzy Systems*, vol. 9, no. 4, pp. 506-515, 2001.

[16] G. Rätsch, T. Onoda, and K.-R. Müller, *Soft margins for AdaBoost. Technical Report NC-TR-1998-021*, Department of Computer Science, Royal Holloway, University of London, Egham, UK, 1998.

[17] L. Rutkowski, *Flexible Neuro-Fuzzy Systems*, Kluwer Academic Publishers, 2004.

[18] R.E. Schapire. "A brief introduction to boosting". In *Proc. the Sixteenth International Joint Conference on Artificial Intelligence*, 1999.

[19] R.E. Schapire, "Theoretical views of boosting", In *Proc. Computational Learning Theory: Fourth European Conference, EuroCOLT'99*, pp. 1-10, 1999.

[20] R. Scherer, L. Rutkowski, "Neuro-Fuzzy Relational Classifiers", in *Proc. 7th International Conference Artificial Intelligence and Soft Computing - ICAISC 2004:*, Zakopane, Poland, June 7-11, 2004, Springer-Verlag Heidelberg , LNAI 3070, pp. 376 – 380, 2004.

[21] L. Xu, A. Krzyzak, Ching Y. Suen, "Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition", *IEEE Trans. on Systems, Man and Cybernetics*, vol. 22, No. 3, May/June 1992, pp. 418-435.

[22] Z.-H. Zhou, I. Wu, W. Tang, "Ensembling Neural Networks: Many Could be Better Than All", *Artificial Intelligence*, vol. 137, no. 1-2, pp. 239-263, 2002.