A Boosting Algorithm with Subset Selection of Training Patterns

Tomoharu Nakashima, Gaku Nakai, and Hisao Ishibuchi
Department of Industrial Engineering, Osaka Prefecture University
Gakuen-cho 1-1, Sakai, Osaka 599-8531, JAPAN
Phone: +81-72-254-9351 FAX: +81-72-254-9915
E-mail: {nakashi, gnakai, hisaoi}@ie.osakafu-u.ac.jp

Abstract - **This paper proposes a boosting algorithm of fuzzy rule-based systems for pattern classification problems. In the proposed algorithm, several fuzzy rule-based classification systems are incrementally constructed from a small number of training patterns. A subset of training patterns for constructing a fuzzy rule-based classification system is chosen according to weights associated to them. The weight for a training pattern is high when it is correctly classified many times. On the other hand, a low weight is assigned to those training patterns that are misclassified many times. Training patterns with a low weight are included in a subset of training patterns for constructing a single fuzzy rule-based classification system. We select the same number of training patterns from each class so that the bias in the number of training patterns among different classes is minimized. In computer simulations, we examine the performance of the boosting algorithm for the fuzzy rule-based classification systems on several real-world pattern classification problems.**

I. INTRODUCTION

Fuzzy rule-based systems have been applied mainly to control problems [1, 2]. Recently fuzzy rule-based systems have also been applied to pattern classification problems. There are many approaches to the automatic generation of fuzzy if-then rules from numerical data for pattern classification problems. Genetic algorithms have also been used for generating fuzzy if-then rules for pattern classification problems [3, 4, 5].

It is generally said that classification performance can be improved by combining several classification systems. For example, Ueda and Nakano [6] analytically showed that the generalization error of averaged output from multiple function approximators is less than that of any single function approximator. For pattern classification problems, various ensemble methods have been proposed [7, 8, 9, 10, 11, 12]. For example, Battiti and Colla [6] examined voting schemes such as a perfect unison and a majority rule for combining multiple neural networks classifiers. Cho [13] and Cho and Kim [14, 15] used fuzzy integrals for aggregating outputs from multiple neural networks. Ishibuchi et al.[16] examined the performance of two levels of voting in fuzzy rule-based classification systems such as a voting by multiple fuzzy if-then rules and a voting by multiple fuzzy rule-based classification systems. The performance of the fuzzy rule-based classification systems is evaluated from various aspects in [17-19].

A boosting algorithm is one of the major techniques for ensemble learning systems in various research fields such as prediction, control, and pattern classification. In a typical boosting algorithm, several simple learning systems are generated using different subsets of training patterns. A subset for constructing a learning system is selected from all the training patterns depending on the history of the performance of the previously generated learning systems. Each learning system is assigned a credit value, which is used for generating a final output. Freund and Schapire [20] proposed AdaBoost algorithm that performs well on both test patterns and training patterns. The AdaBoost successfully dealt with training patterns: Each training pattern has a weight that represents importance in the construction of a single learning system. They showed that the ensemble learning system works well while the performance of a single learning system itself is not good.

This paper proposes a boosting algorithm of fuzzy rule-based systems for pattern classification problems. In the proposed algorithm, several fuzzy rule-based classification systems are incrementally constructed from a small number of training patterns. A subset of training patterns for constructing a fuzzy rule-based classification system is chosen according to the weight associated to them. A weight for a training pattern is high when it is correctly classified many times. On the other hand, a low weight is assigned to a training pattern that is misclassified many times. Those training patterns that have low weights are selected as a subset of training patterns for constructing a single fuzzy rule-based classification system.

Generally, the number of given training patterns for one class is different from those of other classes. In this situation, it is expected that the generated fuzzy rule-based classification systems tend to be biased toward to major classes with a large number of training patterns. To tackle with this problem, we select the same number of training patterns from each class so that the bias in the number of training patterns among different classes is minimized. In computer simulations, we examine the effectiveness of the pattern selection on the performance of the fuzzy rule-based classification systems for several real-world pattern classification problems.

II. FUZZY RULE-BASED CLASSIFICATION SYSTEM

A. Pattern Classification Problem

Let us assume that our pattern classification problem is an n-dimensional problem with C classes. We also assume that we have m given training patterns $\mathbf{x}_p = (x_{p1}, x_{p2}, \ldots, x_{pn})$ $p = 1, 2, \ldots, m$. Without loss of generality, each attribute of the given training patterns is normalized into a unit interval $[0,1]$. That is, the pattern space is an n-dimensional unit hypercube $[0,1]^n$ in our pattern classification problem.

In this paper, we use fuzzy if-then rules of the following type in our fuzzy rule-based classification system:

$$\text{Rule } R_j: \text{If } x_1 \text{ is } A_{j1} \text{ and} \ldots \text{and } x_n \text{ is } A_{jn} \text{ then Class } C_j \text{ with } CF_j, \quad j = 1, 2, \ldots, N, \tag{1}$$

where R_j is a label of the j-th fuzzy if-then rule, A_{j1}, ..., A_{jn} are antecedent fuzzy sets on the unit interval $[0,1]$, C_j is the consequent class (i.e., one of the given C classes), CF_j is the grade of certainty of the fuzzy if-then rule R_j, and N is the total number of fuzzy if-then rules. As antecedent fuzzy sets, we use triangular fuzzy sets as in Fig. 1 where an attribute is divided into various numbers of fuzzy sets. In this paper, we divide each attribute into several fuzzy sets so that the sum of the memberships is always one.

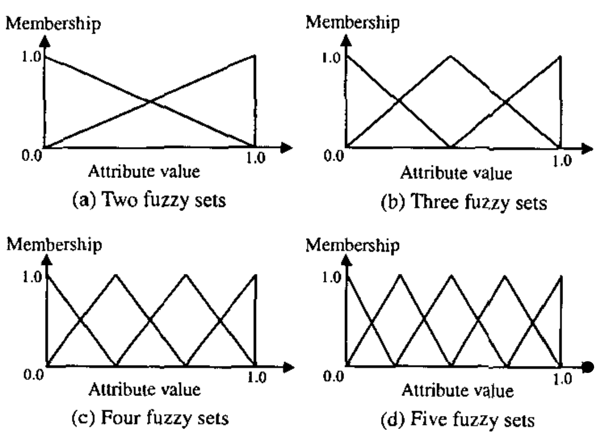

Fig. 1 Antecedent fuzzy sets.

B. Generating Fuzzy If-Then Rules

In our fuzzy rule-based classification systems, we specify the consequent class and the grade of certainty of each fuzzy if-then rule from the given training patterns [16-19]. In [18], it is shown that the use of the grade of certainty in fuzzy if-then rules allows us to generate comprehensible fuzzy rule-based classification systems with high classification performance.

The consequent class C_j and the grade of certainty CF_j of fuzzy if-then rule R_j are determined in the following manner [16-19]:

[Generation Procedure of Fuzzy If-Then Rule]

Step 1: Calculate $\beta_{\text{Class}\,h}(R_j)$ for Class h ($h = 1, 2, \ldots, C$) as

$$\beta_{\text{Class}\,h}(R_j) = \sum_{\mathbf{x} \in \text{Class}\,h} \mu_{j1}(x_1) \cdot \ldots \cdot \mu_{jn}(x_n), \quad h = 1, 2, \ldots, C. \tag{2}$$

Step 2: Find Class $\hat{h}$ that has the maximum value of $\beta_{\text{Class}\,h}(R_j)$:

$$\beta_{\text{Class}\,\hat{h}}(R_j) = \max\{\beta_{\text{Class}\,1}(R_j), \ldots, \beta_{\text{Class}\,C}(R_j)\}. \tag{3}$$

If two or more classes take the maximum value, the consequent class C_j of the rule R_j cannot be determined uniquely. In this case, specify $C_j = \phi$. If a single class takes the maximum value, let C_j be Class $\hat{h}$.

Step 3: If a single class takes the maximum value of $\beta_{\text{Class}\,h}(R_j)$, the grade of certainty CF_j is determined as

$$CF_j = \frac{\beta_{\text{Class}\,\hat{h}}(R_j) - \bar{\beta}}{\sum_{h=1}^{C} \beta_{\text{Class}\,h}(R_j)}, \tag{4}$$

where

$$\bar{\beta} = \sum_{h \neq \hat{h}} \beta_{\text{Class}\,h}(R_j)/(C-1). \tag{5}$$

The number of fuzzy if-then rules in a fuzzy rule-based classification system (i.e., N) is dependent on how each attribute is partitioned into fuzzy sets. For example, when we divide each attribute into three fuzzy sets for a ten-dimensional pattern classification problem, the total number of fuzzy if-then rules is $3^{10} = 59049$.

C. Fuzzy Reasoning

By the rule generation procedure in Section II.B, we can generate N fuzzy if-then rules in (1). After both the consequent class C_j and the grade of certainty CF_j are determined for all the N fuzzy if-then rules, a new pattern $\mathbf{x}$ is classified by the following procedure [16-19]:

[Fuzzy reasoning procedure for classification]

Step 1: Calculate $\alpha_{\text{Class}\,h}(\mathbf{x})$ for Class h, $h = 1,2,\ldots,C$ as

$$\alpha_{\text{Class}\,h}(\mathbf{x}) = \max\{\boldsymbol{\mu}_j(\mathbf{x}) \cdot CF_j \mid C_j = \text{Class}\,h\},\quad h = 1,2,\ldots,C, \tag{6}$$

where

$$\boldsymbol{\mu}_j(\mathbf{x}) = \mu_{j1}(x_1) \cdot \mu_{j2}(x_2) \cdot \ldots \cdot \mu_{jn}(x_n). \tag{7}$$

Step 2: Find Class h^* that has the maximum value of $\alpha_{\text{Class}\,h}(\mathbf{x})$:

$$\alpha_{\text{Class}\,h^*}(\mathbf{x}) = \max\{\alpha_{\text{Class}\,1}(\mathbf{x}),\ldots,\alpha_{\text{Class}\,C}(\mathbf{x})\}. \tag{8}$$

If two or more classes take the maximum value, then the classification of $\mathbf{x}$ is rejected (i.e., $\mathbf{x}$ is left as an unclassifiable pattern), otherwise assign $\mathbf{x}$ to Class h^*.

III. BOOSTING ALGORITHM

In this section, we describe our boosting technique for fuzzy rule-based classification systems described in Section II.

Let d_p, $p = 1,\ldots,m$ be the weight value of a training pattern. A subset of training patterns for generating a single fuzzy rule-based classification system is chosen according to the weight d_p. The key point of our boosting algorithm focuses on how to determine the weight value d_p. In our boosting technique, we first check how many times a training pattern is correctly classified during the course of the boosting process. We assign a large weight value to those training patterns that have been correctly classified by the previously generated fuzzy rule-based systems.

On the other hand, a small weight value is assigned to those training patterns that have not been correctly classified many times. We also calculate the credit of a single fuzzy rule-based classification system. The credit value is used for determining the final output (i.e., the class) of input patterns. Another key point is that we select the same number of training patterns for each class. The aim of this kind of subset selection is to avoid the extreme bias in the number of training patterns from different classes. The proposed boosting algorithm is described as follows:

[Boosting algorithm]

Step 1: Set $d_p = 0$, $c_p = 0$, $p = 1,\ldots,m$, and $t = 1$.

Step 2: Construct a subset U_t such that u training patterns with the smallest values of d_p are included in U_t. If there are multiple training patterns that have the same value of u, training patterns are chosen randomly.

Step 3: Generate a fuzzy rule-based classification system S_t using a subset U_t.

Step 4: Classify all the given training patterns $\mathbf{x}_p$, $p = 1,\ldots,m$, using S_t and update c_p as follows:

$$c_p^{\text{new}} = \begin{cases} c_p^{\text{old}} + 1, & \text{if } \mathbf{x}_p \text{ is correctly classified,} \\ c_p^{\text{old}}, & \text{otherwise,} \end{cases} \quad p = 1,\ldots,m. \tag{9}$$

Step 5: Calculate the credit γ_t of the fuzzy rule-based classification system S_t as follows:

$$\gamma_t = \sqrt{1 - \varepsilon_t}, \tag{10}$$

$$\varepsilon_t = \frac{1}{m} \sum_{p=1}^{m} Z(\mathbf{x}_p) \cdot d_p, \tag{11}$$

$$Z(\mathbf{x}_p) = \begin{cases} 1, & \text{if } \mathbf{x}_p \text{ is not correctly classified,} \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Step 6: Update d_p for the training pattern $\mathbf{x}_p$ as follows:

$$d_p = \frac{c_p}{t}, \quad p = 1,\ldots,m. \tag{13}$$

Step 7: Evaluate the performance of the ensemble fuzzy rule-based classification systems over all the given training data $\mathbf{x}_p$, $p = 1,\ldots,m$. The final output (i.e., the class) of an input pattern $\mathbf{x}_p$ is determined as the most recommended class C_{final} by the fuzzy rule-based classification systems as follows:

$$C_{\text{final}} = \arg\max_{h=1}^{C} \sum_{\substack{k=1 \\ C_k = h}}^{t} \gamma_k, \tag{14}$$

where C_k, $k=1,\ldots,t$ is the classification result of the input pattern by the k-th fuzzy rule-based classification system (See (8)).

Step 8: If at least one of the following termination conditions are satisfied, stop the algorithm:

(a) All the given training patterns $\mathbf{x}_p$, $p=1,\ldots,m$ are correctly classified by the ensemble fuzzy rule-based classification systems.

(b) A prespecified maximal number of fuzzy rule-based classification systems have been generated. That is, $t=T$ where T is the prespecified maximal number of fuzzy rule-based classification systems.

If these above conditions are not satisfied, let $t=t+1$, and go to Step 1.

V. COMPUTER SIMULATIONS

A. Test Problems

In order to examine the performance of the proposed boosting method, we apply it to two real-world pattern classification problems: Iris data set and appendicitis data set [21]. The iris data set is a four-dimensional three-class problem with 50 training patterns from each class. There are 106 training patterns in the appendicitis data set where there are six attributes and two classes. Weiss and Kulikowski [21] has examined the performance of various classification methods for these two data sets.

B. Fuzzy Classification System in the Experiments

As Freund and Schapire [20] pointed out, a weak learner that has at least 50% classification rate is eligible for a member of ensemble classification system in the application of boosting algorithms. In this paper, we use a simple version of fuzzy rule-based classification systems as a member of the ensemble classification system. In the computer simulations, each attribute is divided into only three fuzzy sets (see Fig. 1(b) in the computer simulation on the iris data set). Furthermore, we restricted the number of antecedent fuzzy sets in fuzzy if-then rules up to two in the application to both the appendicitis data set and the iris data set. Thus, the number of generated fuzzy if-then rules in an individual fuzzy rule-based classification is $1 + {}_4C_1 \cdot 3 + {}_4C_2 \cdot 3^2 = 67$ in the case of the iris data set and $1 + {}_6C_1 \cdot 3 + {}_6C_2 \cdot 3^2 = 154$ in the case of the appendicitis data set.

We discretized each attribute into three fuzzy subsets using an information entropy measure [17]. This method first generate candidate thresholds for discretizing an attribute on the center of the neighboring training patterns. Then the information entropy measure E for each candidate threshold is examined as follows:

$$\begin{aligned} E &= \frac{|S_1|}{|S|}\sum_{c=1}^{C}\frac{|S_{1c}|}{|S_1|}\log\frac{|S_1|}{|S_{1c}|} + \frac{|S_2|}{|S|}\sum_{c=1}^{C}\frac{|S_{2c}|}{|S_2|}\log\frac{|S_2|}{|S_{2c}|} \\ &= \frac{|S_3|}{|S|}\sum_{c=1}^{C}\frac{|S_{3c}|}{|S_3|}\log\frac{|S_3|}{|S_{3c}|} \end{aligned}, \tag{15}$$

where S_1, S_2, and S_3 are subsets of S divided by the threshold ($S_1 \cup S_2 \cup S_3 = S$), S_{1c}, S_{2c}, and S_{3c} are the sets of training patterns that belong to class c within the subset S_1, S_2, and S_3, respectively. Finally, the threshold with the minimum information entropy measure is determined as the best partitioning one. These procedures are iterated until all the thresholds are determined.

C. Performance Evaluation

We evaluate the performance of the proposed boosting method for the fuzzy rule-based classification systems that are described in the last section. In the computer simulations, we specified the value of u as follows: For the iris data set, we specified it as $u=5$. That is, we select five training patterns from each class and the total number of training patterns in a subset for constructing a single fuzzy rule-based classification system is 15 for the iris data set. On the other hand, we specified the value of u as $u=15$ for the appendicitis data set. The total number of the selected training patterns for a subset is 30. We constructed a single fuzzy rule-based classification system 14 times (i.e., $T=14$) for the iris data set and 50 times (i.e., $T=50$) for the appendicitis data set.

Simulation results are shown in Fig. 2 to Fig. 5. These figures show the classification rates on training data for Fig. 2 and Fig. 3 and on test data for Fig. 4 and Fig. 5. The performance on test data in Fig. 4 and Fig. 5 is measured by using a leaving-one-out method. In the leaving-one-out method, one given pattern is used as a test pattern and the fuzzy rule-based classification systems are generated from the other given patterns. The performance of the generated fuzzy rule-based classification systems is evaluated by classifying the test pattern. This procedure is iterated until all the given patterns are used as a test pattern. Each of these figures also shows the classification performance of the individual fuzzy rule-based classification system and the ensemble classification system. We can see that the classification rate of the ensemble classification system is increasing as the number of fuzzy rule-based classification systems increases. Its performance is better than any individual fuzzy rule-based classification system. This observation holds for the performance on both training data and test data.

We have performed the same computer simulations with

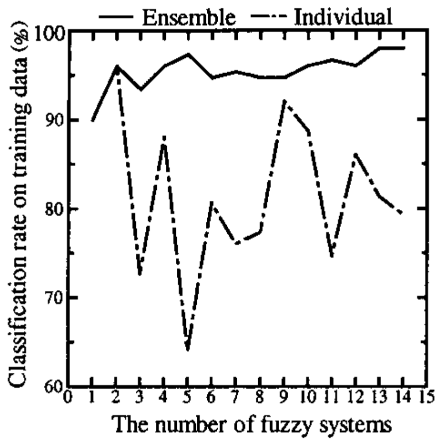

Fig. 2 Performance on training data (Iris data set).

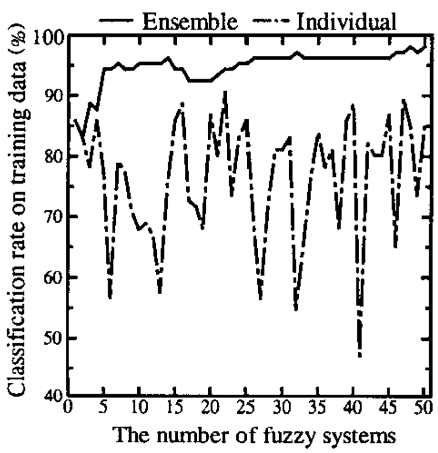

Fig. 3 Performance on training data (Appendicitis data set).

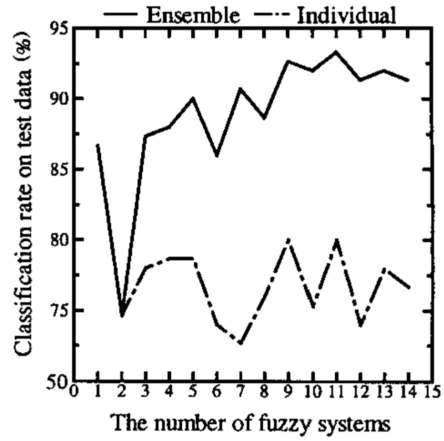

Fig. 4 Performance on test data (Iris data set).

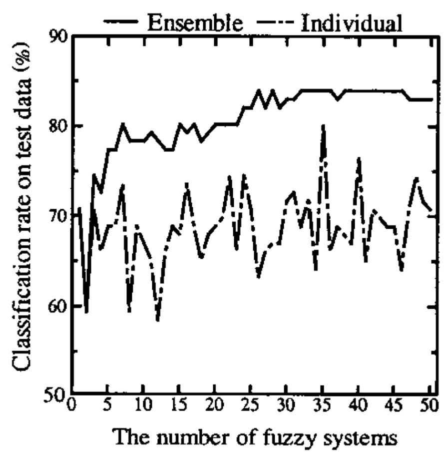

Fig. 5 Performance on test data (Appendicitis data set).

different values of u. That is, different numbers of training patterns are selected from each class.

V. CONCLUSIONS

In this paper, we proposed a boosting method with subset selection of training patterns for fuzzy classification systems. In the boosting algorithm, an individual fuzzy rule-based classification system is incrementally constructed by using a subset of training patterns. Each training pattern is assigned a weight value. A small number of training patterns with the smallest weight values were selected as the member of a subset of training patterns for constructing an individual fuzzy rule-based classification system. One characteristic feature of our boosting algorithm is to select the same number of training patterns from each class. This is because we can avoid the extreme bias in the number of the selected training patterns between different classes.

REFERENCES

[1] M. Sugeno, "An introductory survey of fuzzy control", *Information Science*, vol.30, No.1/2 pp.59-83, 1985.

[2] C.C. Lee, "Fuzzy logic in control systems: fuzzy logic controller Part I and Part II", *IEEE Trans. Syst., Man, Cybern.*, vol.20, pp.404-435, 1990.

[3] H. Ishibuchi and T. Nakashima, "Performance evaluation of fuzzy classifier systems for multi-dimensional pattern classification problems", *IEEE Trans. on Syst., Man, Cybern. Part B*, vol.29, pp.601-618, 1999.

[4] H. Ishibuchi and T. Nakashima, "Improving the perfonnance of fuzzy classifier systems for pattern classification problems with continuous attributes", *IEEE Trans. on Industrial Electronics*, vol.46, no.6, pp.1057-1068

[5] Y. Yuan, H. Zhang, "A genetic algorithms for generating

fuzzy classification rules", *Fuzzy Sets and Systems*, vol.84, no.1, pp.1-19, 1996.

[6] N. Ueda and R. Nakano, "Generalization error of ensemble estimators", *Proc. of Intl. Conf. on Neural Networks*, pp.90-94, 1996.

[7] R. Battiti and A.M. Colla, "Democracy in neural nets: Voting schemes for classification", *Neural Networks*, vol.7, pp.691-707, 1994.

[8] L. Xu, A. Krzyzak, and C.Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition", *IEEE Trans. on Syst., Man, Cybern.*, vol.22, no.3, pp.418-435, 1992.

[9] J.A. Benediktsson and A.M. Colla, "Consensus theoretic classifcation methods", *IEEE Trans. on Syst., Man, Cybern.*, vol.22, no.4, pp.688-704, 1992.

[10] D.H. Wolpert, "Stacked generalization", *Neural Networks*, vol.5, pp.241-259, 1992.

[11] R.A. Jacobs and M.I. Jordan, "Adaptive mixtures of local experts", *Neural Computation*, vol.3, pp.79-87, 1991.

[12] L.K. Hansen and P. Salamon, "Neural network ensembles", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.12, no.10, pp.993-1001, 1990.

[13] S.B. Cho, "Fuzzy aggregation of modular neural networks with ordered weighted averaging operators", *Intl. J. Approx. Reasoning*, vol.13, pp.359-375, 1995.

[14] S.B.Cho and J.H. Kim, "Combining multiple neural networks by fuzzy integral for robust classification", *IEEE Trans. on Syst., Man, Cybern*, vol.25, no.2, pp.380-384, 1995.

[15] S.B. Cho and J.H. Kim, "Multiple network fusion using fuzzy logic", *IEEE Trans. on Neural Networks*, vol.6, no.2, pp.497-501, 1995.

[16] H. Ishibuchi, T. Nakashima and T. Morisawa, "Voting in fuzzy rule-based systems for pattern classification problems", *Fuzzy Sets and Systems*, vol.103, no.2, pp.223-238, 1999.

[17] T. Nakashima and H. Ishibuchi, "Supervised and Unsupervised Fuzzy Discretization of Continuous Attributes for Pattern Classification Problems," *Proc. of Knowledge-Based Intelligent Information Engineering Systems & Allied Technologies*, pp. 32-36, 2001.

[18] H. Ishibuchi and T. Nakashima, "Effect of rule weights in fuzzy rule-based classificaiton systems", *IEEE Trans. on Fuzzy Systems*, vol.9, no.4, pp.506-515, 2001.

[19] H. Ishibuchi, K. Nozaki, and H. Tanaka, "Distributed representation of fuzzy rules and its application to pattern classification", *Fuzzy Sets and Systems*, vol.52, no.1, pp.21-32, 1992.

[20] Y. Freund and R. Schapire, "Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting", vol. 55, no. 1, pp. 119-139, 1997.

[21] S.M. Weiss and C.A. Kulikowski, *Computer systems that learn*, Morgan Kaufmann, San Mateo, 1991.