Automatic Generation of a Fuzzy Rule Base for Online Handwriting Recognition

Ashutosh Malaviya, Hartmut Surmann and Liliane Peters

German National Research Center for Computer Science (GMD)
Schloß Birlinghoven, 53754 St. Augustin, Germany
Ashutosh.Malaviya@gmd.de, *EUFIT'94, Aachen, vol.2, p. 1060-1065, 1994*

ABSTRACT

An automatic method to generate fuzzy rules and their membership functions to recognize handwritten characters is described. Firstly an initial rule base is created on the basis of a referential data set containing handwriting prototypes. Subsequently the classification behavior of the fuzzy rules is optimized with a genetic algorithm, which is regarded as a typical solution to NP-complete problems. A suitable fitness function which corresponds to the human perception of the linguistic variables is obtained. The proposed rule generation process extends the learning and adaptive capabilities of existing fuzzy rule based recognition system.

Keywords: *fuzzy features, fuzzy rule generation, genetic algorithm.*

1.0 INTRODUCTION

Online character recognition systems have to operate in real-time and have to cope with a multiple number of users. These requirements can only be fulfilled if

i) number of character prototypes is reduced, and simultaneously

ii) classification method is flexible enough to match various characters from multiple users.

The proposed scheme achieve these requirements with the flexible and efficient generation of a fuzzy rule base (FRB). The prototypes are represented by linguistic fuzzy rules and the required flexibility is obtained by the variable width of the membership functions. Real time classification of handwritten characters is achieved by limited number of rules and a fast processing algorithm [9]. In this application the syntactic relations between the extracted features in the form of linguistic rules have been utilized to describe the handwritten characters.

The paper is organized as follows: in the next section an outline of the fuzzy online handwriting recognition system (FOHRES)[6] is given. Section 3 handles with the definition and syntax of the fuzzy rule base. In the two subsequent sections initialization and optimization of the fuzzy rule base (FRB) is explained. The last section presents some of the experimental results.

2.0 Outline of the FOHRES System

The human visual system functions successfully even when patterns possess a certain vagueness, slight mismatch and imprecision[11]. It is able to select those features which identify the pattern while ignoring the rest of the uncertainties. Based on this fact we have applied the theory of fuzzy logic to the recognition process. Our solution for the online handwriting recognition problem estimates the imprecision or the vagueness of acquired handwritten symbols in three processing stages, preprocessing, feature extraction and rule generation/classification [6][8] and subsequently with suitable actions these uncertainties are reduced or eradicated. In

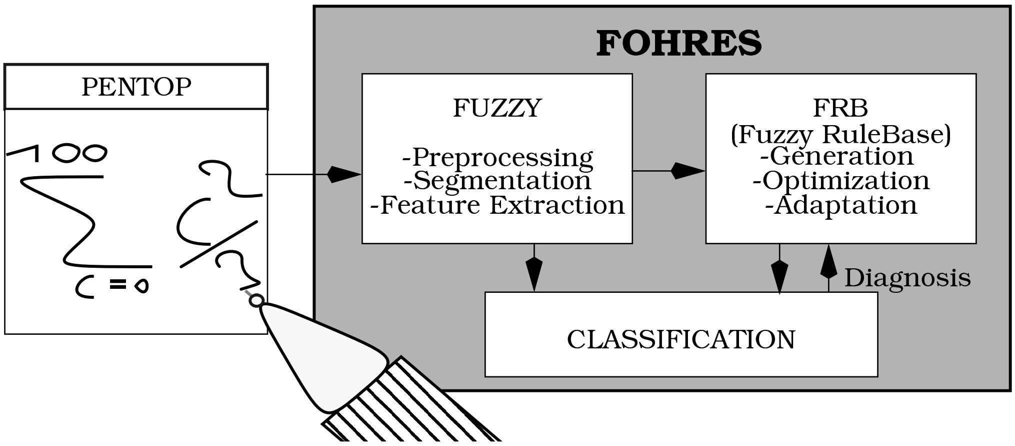

Fig 1. Outline of the FOHRES

the preprocessing stage low level uncertainties e.g. pentop errors, are eliminated. In the next stage these symbols are segmented according to the sudden changes and local minima. Henceforth in the fuzzy feature extraction stage, geometrical and other features (e.g. a *vertical line* at *left*, *thin* or *wide symbol, C-Like or D-Like curves*[6]) of the acquired symbol are determined in the form of fuzzy membership functions [12]. These features are further processed with fuzzy operations (max, α-cut etc.) to reduce the feature set for computational ease in the rule generation/classification stage. Scope of this paper is limited to the final stage i.e. rule generation and classification from the extracted fuzzy features.

In the classification stage vagueness exists in the syntactical interpretation of the extracted features and fuzzy rules. This ambiguity can be termed as syntactic vagueness. However this implies that the acquired fuzzy structural elements have certain correspondence with each other. But this correspondence does not always have unequivocal meanings and secondly it is difficult for a human to generate enormous syntactic definitions manually. To overcome these syntactical ambiguities we have employed the genetic algorithms to generate and optimize fuzzy linguistic rules in the form of a knowledge base or fuzzy rule base (FRB) automatically[10].

The FRB generation is accomplished in two phases. In the first phase a fuzzy rule base (FRB) is initialized with a combination of expert knowledge and some stochastic information. Afterwards this knowledge base is optimized to improve and generalize the initial rules. This optimization is based on the genetic algorithms[1][3][7] and its primary goal is to maximize the classification rate by minimizing the mean fuzzy entropy of the FRB [10]. Because fuzzy entropy characterizes the disorder of the fuzzy rule base. The advantage of the proposed scheme lies in the flexibility and adaptability in incorporating new handwriting styles and newly added symbols. And additionally it simplifies the fuzzy rule base generation process.

3.0 Syntax of the Fuzzy Rule Base (FRB)

It has been proven that FRB are universal approximators[2]. For each $a, b \in \Re$ with $a < b$ let $\mu_{a,b}: \Re \rightarrow \Re$ [0,1] be a membership function such that $\mu_{a,b}(x) \neq 0$ if $x \in [a, b]$. Furthermore let T_1 and T_2 be t-norms and I be a fuzzy implication verifying I(a,0) = 0 if $a \neq 0$, e.g. an R implication or a t-norm implication can be expressed as $I(a,b) = T_3(a,b)$. Let G be a t-conorm.

The 6-tupel $FRB = (T_1, T_2, I, G, S, \mu_{a,b})$ is a family of FRB's with the following properties.

1. A fuzzy rule base is composed by a finite number of rules k of the form:

R_j: *if* x_1 is A_{1j} and ... x_n is A_{nj} *then* y is B_j

where $j = 1,..., k$.

2. The membership function of each A_{ij} is of the form $\mu_{a_{ij}^1, a_{ij}^2}(x)$ for $a_{ij}^1 < a_{ij}^2$; $a_{ij}^1, a_{ij}^2 \in \Re$, i.e.

$$A_{ij}(x) = \mu_{a_{ij}^1, a_{ij}^2}(x) \tag{1}$$

3. The membership function of each B_j is also of the form $\mu_{a,b}$ for some $a < b$; $a, b \in \Re$, i.e.

$$B_j(x) = \mu_{a_{ij}^1, a_{ij}^2}(x) \tag{2}$$

4. T_1 is the fuzzy conjunction operation. The generalized modus ponens is constructed with the other t-norm T_2 and the implication I

a) The rule R_j: *if* x_1 is A_{1j} and ... x_n is A_{nj} *then* y is B_j will be applied only if the n-dimensional input vector x matches with the anti-desiccant, i.e. iff $A_j(\underline{x}) \neq 0$, being

$$A_j(x) = T_1(A_{1j}(x_1), ..., A_{nj}(x_n)) \tag{3}$$

b) If the input vector $\underline{x}$ matches with the anti-desiccant then the inference is

if x_1 is A_1 and ... x_n is A_n *then* y is B

Inference: x is A' $\Rightarrow$ y is B'

$$B'(y) = Sup\{T_2(A'(x), I(A(\underline{x}), B(x)))|x \in \Re\} \tag{4}$$

where $A(\underline{x}) = T_1(A_1((x_1), ..., A_n(x_n)))$

In our decision making application the input $\underline{x} = \underline{x}_0$ is a point, hence $A'(\underline{x}) = \begin{cases} 1, \text{if } \underline{x} = \underline{x}^\circ \\ 0, otherwise \end{cases}$ and

$$B'(\underline{y}) = T_2(1, I(A(\underline{x}), B(\underline{y}))) = I(A(\underline{x}), B(\underline{y})) \tag{5}$$

c) In general, for the input $\underline{x} = \underline{x}_0$ the inference algorithm of the rule R_j is expressed by:

$$B_j'(\underline{y}) = \begin{cases} 0, \text{if } A(\underline{x}^\circ) = 0 \\ I(A_j(\underline{x}), B_j(\underline{y})) \\ \quad ,otherwise \end{cases} \tag{6}$$

which is in the case of a t-norm implication

$$B'(\underline{y}) = I(A_j(\underline{x}^\circ), B_j(\underline{y})) \tag{7}$$

5) The composition of all fuzzy rules is made by the t-conorm G:

$$B'(\underline{y}) = G(\{B_j'(\underline{y})\}) \tag{8}$$

6) The defuzzification method S is the center of the area:

$$y^\circ = S(x^\circ) = \frac{\int B'(y) \cdot (y) dy}{\int B'(y) dy} \tag{9}$$

The important parameters for the FRBS are number of fuzzy rules k and the position and width of the input and output membership functions correspondingly expressed by $a_{ij}^1, a_{ij}^2, b_j^1, b_j^2$.

Theorem *1*: Let $f: U \subseteq \Re^n \to \Re$ be a continuous function defined on a compact *U*. For each $\varepsilon > 0$ there exists a $(FRBS)_\varepsilon \in FRBS$ such that (Proof in [2])

$$Sup\{|f(\underline{x}) - FRBS(\underline{x})|, \underline{x} \in U\} \leq \varepsilon \tag{10}$$

The Mamdani fuzzy controller expressed by $FRBS_{Mam} = (MIN, MIN, MIN, MAX, COG, \mu_{a,b})$ is an universal approximator. In this paper we use a variant of the Mamdani controller which is often applied in the design system from Togai InfraLogic. The FRBS uses triangular or gaussian membership functions which are either linearly approximated by 2 or 6 straight lines or directly stored in look-up tables. For a fast evaluation the centroids *M* and areas *A* of the output membership functions are calculated during the design process and not during the evaluation process. The processing of multiple activated output membership functions and the output composition is achieved by scaling the centroids M_i and A_i with the truth value and addition:

$$y^\circ = \frac{\sum y \bullet M_i}{\sum y \bullet A_i} \tag{11}$$

The modified FRBS is also an universal approximator[2].

4.0 Initialization of the FRB

First task in the automatic design of a fuzzy rule base is the estimation of the appropriate membership function. The basic elements for the problem of practical estimation of membership functions are defined within the theory of possibility introduced by Zadeh. He states that " --- contrary to what has become a widely accepted assumption --- much of the information on which human decisions are based is possibilistic rather than probabilistic in nature." Furthermore, possibility is related to probability, because "... a lessening of the possibility of an event tends to lessen its probability --- but not vice versa."

Therefore in our application the fuzzy membership functions are defined by gaussian distribution functions (normal distribution).

$$\mu(x) = \sigma \cdot \sqrt{2\pi} \cdot \varphi(x;m,\sigma^2) = exp\left(-\frac{(x-m)^2}{2\cdot\sigma^2}\right) \quad (12)$$

Each linguistic variable $j: j = 1, ..., n; n \in \Re$, is represented by a set of q_j membership functions, which are distributed equidistantly in the universe of discourse [0,1]. Mean m and standard deviation σ_j are evaluated by:

$$n_{kj} = e_{j,min} + \frac{e_{j,max} - e_{j,min}}{q_j - 1}(k-1 \quad (12.a)$$

and $\sigma_j^2 = \frac{e_{j,max} - e_{j,min}}{2.9(q_j - 1)}$ where k=1,...,q_j, (12.b)

e represent the evaluation limits of referential data set. The membership function of output linguistic variables are defined as "Yes" and "No" for some triangular, gaussian or singleton fuzzy sets, e.g $\mu_{no(-0.5, 0.5)}$ and $\mu_{yes(0.5, 1.5)}$. In this case the output classifier is a value in the interval [0,1].

After the definition of membership functions fuzzy rule base will be initialized through the following algorithm:

Algorithm 1: Initialize Rule Base

1.Set fuzzy rule base to empty, *FR*=$\underline{0}$.

2.Define minimum activation threshold $\varepsilon \in [0, 0.5]$ of a valid fuzzy rule.

3. Form the referential data set $RV = \{\underline{r}_1,..., \underline{r}_p\}$

4. a)Choose an input vector $r_i \in RV$ randomly.

b) Evaluate the fuzzy rules with the input vector $\underline{r}_i$. Test if $\lambda = T(\mu_{A_1}(r_{i1}), ..., \mu_{A_j}(r_{ij})) \geq \varepsilon$ that means the truth value of the premise is greater than ε.

5. If $\lambda \geq \varepsilon$ then generate no rules. Reduce the referential data to ($RV = RV/ \{\underline{r}_i\}$) and continue from step 4 if $RV \neq 0$.

6. a) If $\lambda < \varepsilon$ then calculate all linguistic variables of the the fuzzy set with the largest truth value for the input r_{ij}.

$$\mu_{A_j}(r_{ij}) = max(\mu(m_{1j}, \sigma^2{}_{1j}), ..., \mu(m_{qj}, \sigma^2{}_{qj})) \quad (13)$$

where $j=1,...,n$.

b) Construct a new fuzzy rule with the most activated fuzzy sets A_{jimax}. In the conclusion set the linguistic variable for the actual cluster to "Yes" and all other output linguistic variables to "No". Again reduce the referential data $RV = RV/\{r_i\}$ and continue from step 4 if $RV \neq 0$. Meanwhile there exist a fuzzy rule for vector r_j with $\lambda > 0.5$, while the membership functions are 50% overlapped.

In the next design phase a genetic algorithm is applied to adjust the centers and the widths of the fuzzy points so that they achieve certain flexibility to cope with the varying nature of incoming symbols to be recognized,

<u>Notes on fuzzy sets and rules generation</u>

1.Computing time increases approximately linearly with the number of the input patterns (Size of the referential data set).

2. Storage capacity increases linearly with the number of linguistic variables.

3. A fuzzy point (set) is defined by only two parameters namely σ and *m*.

4. In addition to fuzzy methods conventional algorithms can also be used to determine the fuzzy rules.

5.0 Optimization of FRB with GA

A robust handwriting recognition system aims for an improvement in system behavior with respect to changing conditions, e.g. incorporating new users or differently written characters. Therefore with optimization we will seek more flexible fuzzy rules. For optimizing the FRB of handwritten characters, a highly robust algorithm is needed which can cope with complex and relatively huge search (parameter) spaces. There is no theoretical approach that defines an optimal fuzzy system for a given task. Therefore adaptation is necessary to generate well designed FRB. A general FRB is very complex. Even when parts of the structure are fixed, e.g. by a hardware implementation, a lot of parameters remain unset[4][9]. Especially the fuzzy sets are important for fine tuning the transfer function of the system. In contrast, the rule base is less flexible, even when the rules are weighted. Considering the number of parameters of all fuzzy sets, the search space becomes very complex.

A suitable optimization algorithm for the adaptation of a FRB is the genetic algorithm (GA) introduced by Holland, which is described in detail by Davis in [3]. This is regarded as a typical solution of the NP-complete problems.

Algorithm 2: Optimize Rule base with **GA**

begin

$t \leftarrow 0$

initialize $P_{FRB}(t)$

evaluate structures in $P_{FRB}(t)$

While termination condition

is not satisfied **do**

begin

$t \leftarrow t+1$

select $P_{FRB}(t)$ from $P_{FRB}(t-1)$;

recombine structures in $P_{FRB}(t)$;

evaluate structures in $P_{FRB}(t)$;

end

end.

Binary coded parameters are used to construct a string. Every string A will hold all fuzzy set parameters m_{ij} and σ_{ij}, i = 1...n, j = 1...q of a FRB as proposed by Karr[5]. The parameters are linearly transformed to integers of l bits accuracy (e.g. l=8 bits) with constant limits for every variable. Denoting the coded fuzzy set parameters $\hat{m}_{ij}$ and $\hat{\sigma}_{ij}$, a string A is constructed by adding the parameters in a row:

$$A = \hat{m}_{11}\hat{\sigma}_{11}\hat{m}_{12}\hat{\sigma}_{12}...\hat{m}_{nq}\hat{\sigma}_{nq} = \underbrace{a_1 a_2 ... a_l}_{m_{11}}\, a_{l+1} ... a_L$$

where $a \in \{0, 1\}$. The length L of the string can be computed as $L = 2.l.n.q.$ The parameters of the algorithm can be selected as follows. Fixed values are chosen for the algorithm's random parameters, so the mutation rate p_{mut} = [0.002,0.02] which increases exponentially, and crossing over rate p_{cross} = 0.8. Size of population M and time interval T will be selected according to the system complexity (in our case M=500, T=500).

The fitness function constitutes of the correctness measure of the classification space and the entropy Q of the FRB as defined in [10].

$$Q_N = \sum_{i=0}^{N} a_i \text{ where } a_i = \begin{cases} 1, if(P_i)\text{Recognized} \\ 0, else \end{cases}$$

$$Q = \begin{cases} Q_N / \left(\left(\frac{R_{act}}{R_{max}} - 1\right) \cdot a + \left(\frac{R_\phi}{R_{max}} - 1\right) \cdot b + 1\right) \\ ,if(R_{max} < R_{act}) \\ \dfrac{Q_N}{\left(\frac{R_\phi}{R_{max}} - 1\right) \cdot b + 1} \\ ,Otherwise \end{cases} \tag{14}$$

Where $R_{max} \in N$: nominal number of activated rules, $R_{act} \in N$: current number of activated rules, $a, b \in R$ are the local and global parameters to punish the rules according to the computing and memory limitations.

For fitness function calculation, a FRB has to be constructed from each string and tested by evaluating the referential data set. Therefore, a large number of FRB executions are required. A discrete FRB with a very fast evaluation method is used for this task. The coding used for the GA corresponds to the discrete implementation of FRB to facilitate the real time evaluation. In addition, the coding's limitations due to discretization and constant intervals are of little importance, because FRBs are robust and the range of the referential data set is also limited.

Notes on Genetic Algorithms

1. GAs work with a population of competing solutions. Therefore, the algorithms can escape from local fitness maxima in the search (parameter) space [1].

2. The genetic operators are probabilistic and are applied to coded parameter sets only. They are independent from the structure of the search space. Therefore GAs are robust.

3. Execution time rises linearly with population size *M*, evolution steps *T* and length of strings *L* (under the assumption that FS calculation time is proportional to the number of fuzzy sets).

6.0 Experimental Results

We have acquired handwriting data for five different users which was limited to the decimal numbers. Reference data set contained twenty sets of extracted features. Obtained results were extremely satisfying. In the first stage where rules were initialized with the help of statistical information, the classification rate was 72.5%, but after the optimization in the second stage classification rate was drastically improved and all the tested symbols were recognized except only one. It should also be mentioned that the characters of the referential data set were written in entirely different manners to guarantee the robustness. Testing of these rule base was also satisfactory and the obtained classification results were correct in 97.25% cases for the users belonging to the reference data sets. and 90.4% for 5 new users who followed the prototypes from referential data set.

7.0 References

[1] T. Bäck and H.-P. Schwefel,"An overview of evolutionary algorithms for parameter optimization," *Evolutionary Computations*, Vol.1, pp. 1-24, 1993.

[2] J.L. Castro,"Fuzzy Logic Controllers are Universal Approximators," *Tech. Report* #DECSAI-93101, Universidad de Granada, Spain, June 1993.

[3] L. Davis, *Handbook of genetic algorithms*, Van Nostrand Reinhold, NY, 1991.

[4] A. Kanstein, H. Surmann et al,"Nochmals ein genetischer Algorithmus zum Optimieren von regelbasierten Fuzzy-Systemen ?," 4th Dortmunder Fuzzy Tage, 1994.

[5] C.L. Karr et al,"Improved fuzzy process control of spacecraft autonomous rendevouz using a genetic algorithm," SPIE, Vol.1196, pp.274-288, 1989.

[6] A. Malaviya, L. Peters et al, "A Fuzzy Online Handwriting Recognition System: FOHRES," *2nd Int'nl Conf. on Fuzzy Theory and Tech.*, NC, Oct. 1993.

[7] H. Mühlenbein,"How Genetic Algorithms Really Work: Mutation and Hill-climbing," in *Parallel Problem Solving from Nature*, Ed. R. Männer and B. Manderick, pp. 15-26, Amsterdamm, North-Holland, 1992.

[8] M. Nakagawa,"Principles of Pen Interface Design for Creative Work," *Proc. of the 2nd Int'nl Conf. on Document Analysis and Recognition*, Japan, Oct. 1993.

[9] H. Surmann et al,"What kind of Hardware is necessary for a fuzzy rule based system?," Submitted to FUZZ-IEEE-94 in Orlando, 1994.

[10] H. Surmann et al.,"Self-Organizing and Genetic Algorithms for an Automatic Design of Fuzzy Control and Decision Systems," EUFIT'93, pp. 1097-1104, Aachen, Sept. 1993.

[11] C.Y.Suen,"Handwriting Generation, Perception and Recognition," *Acta Psychologica*-54, pp. 295-312, 1983.

[12] L.A. Zadeh,"Quantitative Fuzzy Semantics," *Information Sciences*-3, pp. 159-176, 1971.