Fourth International Workshop on Advanced Computational Intelligence
Wuhan, Hubei, China; October 19-21, 2011

freepaper.me FREE paper

Fuzzy Rough Neural Network and Its Application to Feature Selection

Jun Y. Zhao, and Zhi L. Zhang

***Abstract*—For the sake of measuring fuzzy uncertainty and rough uncertainty of real datasets, the fuzzy rough membership function (FRMF) defined in fuzzy rough set is introduced. A new fuzzy rough neural network (FRNN) is constructed based on neural network implementation of FRMF. FRNN has the merits of quick learning and good classification performance. And then a new neural network feature selection algorithm based on FRNN is designed. The input nodes of FRNN are pruned according to the descent of classification accuracy; thereby the search of optimal feature subset is realized with reference to residual input nodes. The test results on UCI datasets show that the algorithm is quick and effective, and has better selection precision and generalization capability than RBF feature selection.**

I. Introduction

IN the application field of pattern recognition, the mass features under analysis are often irrelevant or redundant. Therefore, the collected data in practice should be preprocessed to remove redundant features before utilizing these datasets. Feature selection is one of the most important preprocessing methods in the research fields such as data mining and pattern recognition. Feature selection picks a feature subset with excellent capability from the initial high-dimensional feature set based on certain evaluation criterion, and aims to preserve or even improve the classification performance of the data set. Originally, feature selection attracts not enough attention, but with the development of computer technology and the explosive of information, feature selection gets increasingly importance to information processing. Some famous feature selection algorithms as B&B, Focus, Relief, LVF and CFS are proposed.

Neural network adapts to deal with noise data for its merits of strong approaching ability and good fault-tolerance performance. So the combination of neural network and feature selection can further enhance the capability of complex data processing. There are two kinds of combinative modes: one is preprocessing data firstly by feature selection, and then inputting the data to neural network for further research; the other is utilizing directly the configuration character and classification performance of neural network to realize feature selection. The article emphasizes particularly on the second mode. This mode was initially implemented in Neural Network Feature Selector (NNFS) proposed by Setiono and Liu in 1997 [1], which realized feature selection by pruning a three-layer feedforward neural network to remove redundant network connections. Subsequently, Basak described a new feature selection method by examining the parameters of a trained radial basis function network [2]. The connection weights of hidden and output layers indicate the relative importance of clusters and classes, and the intraclass and interclass distances are calculated to evaluate feature subsets. Sankar K. Pal proposed a neuro-fuzzy approach for unsupervised feature evaluation by training network weights which indicate the importance of corresponding features to rank these features in 2000 [3]. A. Verikas presents a new neural network training method with an augmented cross-entropy error function based on NNFS [4], and a new pruning approach for identifying salient features according to classification accuracy in 2002.

Fuzzy uncertainty and rough uncertainty are existent in real data. In order to completely measure data uncertainty, a new four-layer feedforward fuzzy-rough neural network is constructed based on fuzzy rough membership function of fuzzy rough set, and a new feature selection method is realized based on this network. The article will first introduce the notions of fuzzy rough set and fuzzy rough membership function in section 2; and then the structure and input-output connections of FRNN are presented in section 3. Finally, a new feature selection algorithm based on FRNN is proposed in section 4, and the performance of the algorithm is tested by UCI datasets and compared with RBF feature selection method.

Manuscript received June 7, 2011.

Z. J. Zhao is with Xi'an Research Inst. Of Hi-tech Hongqing Town, Xi'an, P.R.China (phone: +86-029-84744096; e-mail: zhaojy802@sina.com).

Z. L. Zhang is with Xi'an Research Inst. Of Hi-tech Hongqing Town, Xi'an, P.R.China.

II. Fuzzy Rough Membership Function

Firstly, let us introduce fuzzy rough set. Fuzzy rough set can be defined by a pair of fuzzy lower approximation and fuzzy upper approximation of fuzzy approximation space as follows [5].

Definition 2.1 Let (U,R) be a fuzzy approximation space, U is a non-empty set of finite objects, $x \in U$, R is a fuzzy equivalence relation on U, $\forall X \in f(U)$, the lower approximation and upper approximation denoted by $\underline{R}X$ and $\overline{R}X$ are defined as:

$$\mu_{\underline{R}X}(x) = \inf_{y \in U} \max\{\mu_X(y), 1-\mu_R(x,y)\} \quad (1)$$

$$\mu_{\overline{R}X}(x) = \sup_{y \in U} \min\{\mu_X(y), \mu_R(x,y)\} \quad (2)$$

Definition 2.2 Let (U,R) be a fuzzy approximation space, U is a non-empty set of finite objects, $x \in U$, R is a fuzzy equivalence relation on U, $U/R = \{F_1, F_2, \cdots, F_H\}$, $\forall X \in f(U)$, the degree of

fuzzy rough membership [6] of x relative to X is defined by

$$\rho_X(x)=\begin{cases}\dfrac{1}{H}\sum_{i=1}^{H}\mu_{F_i}(x)\tau_X^i(x) & if\ \mu_{F_i}(x)>0\\ 0 & else\end{cases} \tag{3}$$

Where, $\mu_{F_i}(x)$ is the degree of fuzzy membership of x relative to F_i, and the second term $\tau_X^i(x)=\dfrac{|F_i\cap X|}{|F_i|}=\dfrac{\sum_{x\in U}\min(\mu_{F_i}(x),\mu_X(x))}{\sum_{x\in U}\mu_{F_i}(x)}$ is the degree of rough membership of x relative to X, H is the number of fuzzy classes containing x and $\mu_{F_i}(x)>0$.

III. Fuzzy Rough Neural Network

In literature [7], Sarkar once proposed a three-layer fuzzy rough neural network based on fuzzy rough membership function, and applied the network to vowel classification. This network can dispose the rough uncertainty issue of fuzzy clustering, but has the limitations of low adaptation to new data and low classification accuracy. Therefore, Zhang dongbo proposed an improved method, but this method needs a great deal of time for training weights [8]. To reduce training time, a four-layer feedforward fuzzy-rough neural network including input layer, cluster layer, membership layer and output layer is designed as showed in Figure 1. The neuron number of input layer lies on the dimensions of input pattern; the neuron number of cluster layer lies on the cluster number of input data which clustered by unsupervised FCM algorithm; the neuron number of membership layer and output layer are equal to the output classes.

Let net_j^i and O_j^i be the input and output signals of the j th neuron in the i th layer, and ω_{ji}^{pq} be the connection weight coming from the i th neuron in the p th layer to the j th neuron in the q th layer. The detailed input-output relations of each layer will be described below:

1) Input layer: The original data is imported to the input layer without any transformation, and each node is corresponding to a feature of input sample.

$$net_j^1=x_j \quad j=1,2,\cdots,N \tag{4}$$

$$O_j^1=net_j^1 \tag{5}$$

Where, x_j is the j th feature of input sample $x=(x_1,x_2,\cdots,x_N)^T$, N is the number of features.

2) Cluster layer: Cluster layer calculates the fuzzy memberships of each input sample to clusters by gauss function.

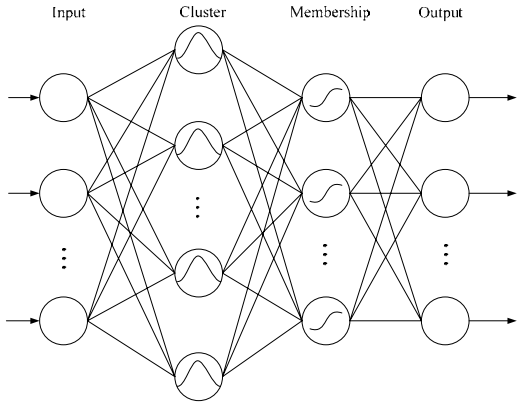

Fig. 1. Structure of improved fuzzy rough neural network

$$net_j^2=x=(x_1,x_2,\cdots,x_N)^T \tag{6}$$

$$O_j^2=\exp\left[-\frac{(x-m_j)^T(x-m_j)}{2\sigma_j^2}\right] \tag{7}$$

Where, m_j and σ_j are the cluster center and deviation of the j th cluster. If F_j represents the j th cluster, then O_j^2 can be expressed as $\mu_{F_j}(x)$.

3) Membership layer: The input of membership layer denotes the value of fuzzy rough membership, and the actuation function adopts *tansig*, then

$$net_j^3=\sum_{i=1}^{H}\omega_{ji}^{23}O_i^2 \tag{8}$$

$$O_j^3=\frac{2}{1+\exp(-2net_j^3)}-1 \tag{9}$$

H is the neuron number of membership layer. The connection weights ω_{ji}^{23} of layer two to layer three can be calculated by formula (10), namely

$$\omega_{ji}^{23}=\frac{1}{H}\frac{|F_i\cap C_j|}{|F_i|}=\frac{1}{H}\frac{\sum_{x\in C_j}\mu_{F_i}(x)}{\sum_j\sum_{x\in C_j}\mu_{F_i}(x)} \tag{10}$$

Where, C_j is the j th output class.

4) Output layer: The output layer finally calculates the memberships of input samples to each output class, and the samples will be classified to the class which has the maximum membership value.

$$net_j^4=\sum_{i=1}^{c}\omega_{ji}^{34}O_i^3 \tag{11}$$

$$O_j^4=net_j^4 \tag{12}$$

Where, c is the output class number. The connection weights ω_{ji}^{34} of layer three to layer four can be trained by gradient descending method, or calculated directly. The gradient descending method needs a lot of time, and can't

freepaper.me FREE paper

ensure the convergence, so ω_{ji}^{34} will be calculated directly in the article.

Let S be the target outputs, then $W^{34}O^3 = S$, the weight matrix is calculated as

$$W^{34} = (O^3)^{-1} S \tag{13}$$

IV. Feature Selection Algorithm based on FRNN

A. The Essence of the Algorithm

In this section, FRNN is applied to feature selection by heuristic backward search strategy. Originally, all features set A is imported to FRNN which will be trained to satisfy desired precision in the training set S_1, and then the precision variations of the network in the validating set S_2 after deleting each feature of set A are compared. The feature which has the least precision variation is chosen to delete and correspondingly FRNN is trimmed. If the precision descending level of the trimmed FRNN in the validating set S_2 is controlled under the pre-designed threshold, then this feature is not important and will be deleted, else retained. The process is repeated until precision descending levels of all input features remained in the network exceed the threshold, and all residual features can't be deleted.

As described above, the criterion to evaluate features is the network precision descending level after deleting features. In the article, the network precision is defined according to the network classification accuracy to input samples:

$$Accu = X_{Accu} / X_{All} \tag{14}$$

Where, X_{Accu} is the number of correctly classified samples, X_{All} is all input samples.

B. The Realization of the Fuzzy Rough Neural Network Feature Selection Algorithm (FRNN_FS)

Algorithm: FRNN_FS

1) Construct a four-layer feedforward fuzzy-rough neural network, let $A = \{A_1, A_2, \cdots, A_N\}$ be the set of all features, train network Q to satisfy desired precision, and calculate network precision R^1 and R^2 in the training set S_1 and validating set S_2, ΔR is permitted precision descending level;

2) To all input nodes $k = 1, 2, \cdots, N$

a) Delete the kth input node of network Q and all weights connected to this node to construct a new network Q_k;

b) Calculate the classification precision R_k^2 of Q_k in the validating set S_2;

c) Calculate precision descending ratio $\delta_k = (R^2 - R_k^2)/R^2$ of Q_k;

3) Rank δ_k, let $\delta_{\min}$ be the ratio of most unapparent feature;

4) If $\delta_{\min} < \Delta R$, then delete this feature and corresponding input node, $N = N - 1$, return to step 1);

Else, end.

This feature selection algorithm adopts heuristic backward search strategy, and doesn't need exhaustive search or complete search. Let the number of conditional features be N, then the time complexity of the algorithm is $O(N \log N)$.

V. Experimental Analysis by UCI Data

In order to validate the feasibility and performance of the algorithm, six common UCI datasets are introduced to test the algorithm, which are breast_cancer, breast_cancer_wisconsin, Cleveland, Glass, Wine and Wdbc [9]. For the convenience of writing, breast_cancer and breast_cancer_wisconsin are shortened as B.C. and B.C.W. These datasets contain not only discrete features, but also continuous features, and need to deal with two-class or multi-class problems. Besides, this algorithm is compared to RBF feature selection algorithm with backward pruning strategy which has good approaching capability. The orthogonal least square (OLS) method which has the merits of high training precision, fast convergence velocity and self-determination of hidden neurons is applied to train RBF network. In the experiments, the neuron number of cluster layer is twice the number of decision classes, each newly constructed network is trained at least 30 times, precision descending ratio ΔR is set to 0.03.

Table 1 shows the classification precision comparisons of original feature sets and subsets selected by FRNN on Cleveland, Wine and Wdbc datasets. The training and test precisions of the selected subsets on these three datasets are all heightened, except the training precision on Wine. Besides, the feature number of the subsets decreases greatly. For example, the subset feature number on Wdbc is only 11.3% of original features; synchronously the average test precision improves 3.2%. Another character is that the test precision of FRNN is higher than the training precision regardless original feature sets or subsets, and improves 4% on Wine, only except on original Cleveland. It indicates that FRNN has favorable generalization capability.

Table 2 shows the performance comparisons of subsets selected by RBF and FRNN on the six datasets, such as classification precision, feature numbers of subsets, run time. According to the No Free Lunch theory, different algorithms have respective merits and faults. There is no algorithm absolutely excellent than another algorithm, but also it is true that one algorithm can be synthetically excellent than another algorithm. From the data in table 2, FRNN_FS obviously performs well than RBF_FS in most indexes. The only exception is that the training precision of FRNN is not as good as RBF. As showed on the results of these six datasets, the average training precision of RBF is

freepaper.me FREE paper

96.76%, and that the average training precision of FRNN is 79.78%, but the performances of these two networks on test precision are quite different. The average test precision of RBF is only 74.56%, and the average test precision of FRNN is 81.70% with 9.6 percent improvement. It illustrates the generalization capability of FRNN is better than RBF. This character represents much obviously on B.C., Cleveland and Glass. Taking B.C. as an example, the training precision of FRNN is lower than RBF by 21.1%, but the test precision of FRNN is higher than RBF by 16.49%.

Another important objective of feature selection is to select features as less as possible while ensuring classification precision. The average feature number of subsets selected by FRNN is 3.92, which is only 64.1% of the average feature number of subsets selected by RBF. Especially on Cleveland, the subset feature number of FRNN is less than RBF by 61.5%, but the test precision of FRNN is higher than RBF by 15.62%. Besides, the run time of FRNN_FS averagely reduces than RBF by 92.5%. As we know, the reduction extent is certainly related to each dataset. It does not mean that the run time of FRNN_FS can be reduced greatly to all datasets, but at least it indicates the running efficiency of FRNN_FS is relatively high.

TABLE I
CLASSIFICATION ACCURACIES OF ORIGINAL FEATURE SETS AND SUBSETS SELECTED BY FRNN_FS

Evaluation Indexes	Cleveland		Wine		Wdbc	
	All features	*Selected features*	*All features*	*Selected features*	*All features*	*Selected features*
Average feature number	13(0.00)	3(0.47)	13(0.00)	5.5(2.17)	30(0.00)	3.4(0.97)
Average training precision (%)	59.51(1.19)	61.43(0.77)	98.13(0.00)	94.77(1.48)	92.38(0.00)	93.70(0.95)
Average test precision (%)	57.52(1.57)	62.73(1.32)	98.59(0.00)	98.73(1.23)	92.94(0.14)	95.88(0.59)

TABLE II
CLASSIFICATION ACCURACIES OF FEATURE SUBSETS SELECTED BY FRNN AND RBF

Datasets	RBF_FS				FRNN_FS			
	Training precision (%)	*Test precision (%)*	*Feature number*	*Time(s)*	*Training precision (%)*	*Test precision (%)*	*Feature number*	*Time(s)*
B. C.	92.44(0.78)	57.02(1.05)	4(0.00)	30.25	71.34(1.86)	73.51(1.93)	3.4(1.17)	0.82
B. C. W	98.57(0.86)	93.91(0.34)	4(0.00)	17.41	93.86(0.98)	97.81(0.55)	3.9(1.10)	0.78
Cleveland	97.25(0.65)	47.11(0.52)	7.8(0.32)	23.70	61.43(0.77)	62.73(1.32)	3.0(0.47)	1.52
Glass	95.31(1.82)	54.65(1.28)	4(0.00)	2.86	63.59(3.72)	61.51(2.29)	4.3(0.95)	0.73
Wine	98.13(0.85)	98.59(0.76)	8(0.00)	1.16	94.77(1.48)	98.73(1.23)	5.5(2.17)	0.95
Wdbc	98.83(0.88)	96.05(0.58)	8.9(0.85)	49.38	93.70(0.95)	95.88(0.59)	3.4(0.97)	4.54
Ave.	96.76	74.56	6.12	20.79	79.78	81.70	3.92	1.56

As synthesized from the above analysis, FRNN has the merits of high classification precision, fast training velocity and good generalization capability. The feature selection algorithm based on FRNN can select subsets with smaller features, at the same time the subsets possess high average classification precision. Therefore, the algorithm FRNN_FS is stable and effective.

VI. CONCLUSION

In order to deal with fuzzy uncertainty and rough uncertainty of real data, the concept of fuzzy rough membership function defined in fuzzy rough set is introduced. An improved fuzzy rough neural network combining FRMF and neural network is constructed. This new FRNN is further applied to feature selection by repeatedly trimming network input nodes to improve classification precision, and the FRNN_FS algorithm is proposed based on this method. Datasets experiments show that FRNN_FS can deal with complex real data, select feature subsets effectively, and also with good generalization capability. Simultaneously, the data imported to FRNN_FS need to be clustered beforehand, and the number of cluster layer neurons can't be determined directly, but need to be presented according to experiences. These disadvantages should be ameliorated in the future.

REFERENCES

[1] R. Setiono and H. Liu, "Neural-Network Feature Selector," IEEE Trans. on Neural Network, vol. 8, pp. 654-661, 1997.

[2] J. Basak and S. Mitra, "Feature Selection Using Radial Basis Function Networks," Neural Comput & Applic, vol. 8,pp. 297–302, 1999.

[3] Sankar K. Pal, Rajat K. De, and J. Basak, "Unsupervised Feature Evaluation: A Neuro-Fuzzy Approach," IEEE Trans. on Neural Network, vol. 11, pp. 366–376, 2000.

[4] A. Verikas and M. Bacauskiene, "Feature selection with neural networks," Pattern Recognition Letters, vol. 23, pp. 1323–1335, 2002.

[5] D. Dubois and H. Prade , "Rough fuzzy sets and fuzzy rough sets," Int. J. Gen. Syst., vol. 17, pp. 191–209, 1990.

[6] M. Sarkar and B. Yegnanarayana, "Fuzzy-rough Membership Functions," Proc. IEEE Int. Conf. on Systems, Man and Cybernetics, California, USA, pp. 2028–2033, 1998.

[7] M. Sarkar and B. Yegnanarayana,"Fuzzy-Rough Neural Network for Vowel Classification," Proc. IEEE Int. Conf. on Systems, Man and Cybernetics, California, USA, pp. 4160–4165, 1998.

[8] D. B. Zhang and Y.N.Wang, "Fuzzy-rough Neural Network and Its Application to Vowel Recognition," Control and Decision, vol. 21, pp. 221–224, 2006.

[9] Http://www.mlearn.ics.uci.edu/MLRepository.html

freepaper.me FREE paper