Neural Network Based on Rough Sets and Its Application to Remote Sensing Image Classification

WU Zhaocong  LI Deren

1 Introduction

In most cases, the neural network classifiers have proved superior to conventional classifiers, often recording overall accuracy improvements in the range of 10-20%. Much of the neural network classification work uses multi-layer feed-forward networks that are trained using the back propagation algorithm based on a recursive learning procedure using a gradient descent search. However, this training procedure is sensitive to the choice of initial network parameters and over fitting. Moreover, its training may take a long time, often requiring thousands or even tens of thousands of iterations before convergence of the error minimization is reached.

The rough sets theory proposed by Zdzislaw Pawlak is a mathematical theory dealing with uncertainty and incompleteness of data. Some researchers are investigating methods to combining rough sets and neural network to solve the problems encountered in conventional neural network.

In this paper, we combine these two theories and propose a new kind of neural network based on rough sets.

2 Information system and rough sets

2.1 Information system and decision table

The rough sets theory deals with information represented by a table called information system.

The information system is composed of 4 tuples:
where $U$ is the universe, a finite set of $N$ objects $\{X_1, X_2, \cdots, X_n\}$, and $U \neq \emptyset$; $Q$ is a finite set of attributes, and $Q \neq \emptyset$; $V = \bigcup_{q \in Q} V_q$, where $V_q$ is a value domain of the attribute $q$; $f: U \times Q \rightarrow V$ is the total decision function called the information function and $f(x, q) \in V_q$ for every $q \in Q, x \in U$.

Some of the information systems can be designed as a decision table (DT):

$$DT = < U, C \cup D, V, f >$$

where $C$ is a set of condition attributes (inputs, features); $D$ is a set of decision attributes (actions, categories); and $f: U \times (C \cup D) \rightarrow V$ is a total decision function (information function, decision rule in DT) that $f(x, q) \in V_q$ for every $q \in Q$ and $x \in V^{[2]}$.

2.2 Rough sets

The rough sets theory is based on the concept of an upper and a lower approximation of a set. For a given information system $S$, a given subset of attributes $R \subseteq Q$ determines the approximation space $AS = (U, \text{IND}(R))$ in $S$. For a given $R \subseteq Q$ and $X \subseteq U$, the $R$-lower approximation $RX$ of set $X$ in $AS$ and the $R$-upper approximation $R-X$ of set $X$ in $AS$ are defined as follows:

$$RX = \{x \in U : [x]_R \subseteq X\}$$  

$$R-X = \{x \in U : [x]_R \cap X \neq \emptyset\}$$

For a given subset of attributes $R \subseteq Q$, the

$$\text{IND}(R) = \{(x, y) \in U : \text{for all } a \in R, f(x, a) = f(y, a)\}$$

is an equivalence relation in the universe, called an indiscernibility relation. It partitions the universe set $U$ into equivalence classes that contain all objects which are indiscernible in terms of attributes set $R$.

For a given set of condition attributes $R \subseteq C$, we can define a positive region $\text{POS}_R(D)$ in the relation $\text{IND}(D)$, as

$$\text{POS}_R = \bigcup \{RX \mid X \in \text{IND}(D)\}$$

2.3 Reduct and rule generation

For an information system $S$ and a subset of attributes $R \subseteq Q$, an attribute $a \in R$ is dispensable if $\text{IND}(R) = \text{IND}(R - \{a\})$. It means that indiscernibility relations generated by sets $R$ and $R - \{a\}$ are identical. The set of all indispensable attributes in the set $R \subseteq Q$ is called a core of $R$, and it is denoted by $\text{CORE}(R)$. Set $B$, any minimal subset of the attribute $R$ preserving indiscernibility, i.e., $\text{IND}(B) = \text{IND}(R)$, is called a reduct of $S$. The set of all reducts of $S$ is denoted by $\text{RED}(S)$ or $\text{RED}(R)$.

From a decision table $DT$, decision rules can be derived. Let $U \setminus \text{IND}(C) = \{X_1, X_2, \cdots, X_n\}$ be $C$-definable classification of $U$ and $U \setminus \text{IND}(D) = \{Y_1, Y_2, \cdots, Y_m\}$ be $D$-definable classification of $U$. A class $Y_i$ from a classification $U \setminus \text{IND}(D)$ can be identified with the $i$th ($i = 1, 2, \cdots, l$) decision rule

$$\text{Des}_C(X_i) \Rightarrow \text{Des}_D(Y_j)$$

for $X_i \in R$, $Y_j \in D$

where $\text{Des}_C(X_i)$ denotes the description of $C$-elementary set $X_i \subseteq C$. The set of decision rules for all classes $Y_j \in D$ is denoted as follows:

$$\{|r_{ij}| = \{|\text{Des}_C(X_i) \Rightarrow \text{Des}_D(Y_j) : X_i \cap Y_j \neq \emptyset\} = \{|X_i \setminus \text{IND}(D)\}$$

3 Rough back propagation neural network

From the concepts mentioned above, we construct a BPNN based on rough sets, called rough back propagation neural network (RBPNN). This network includes five layers, as depicted in Fig. 1.

The first layer of network represents the input from the feature space $X = \{X_1, X_2, \cdots, X_n\}$, where $n$ is the dimension of input variables.

The second layer stands for the discretion of input variables. An input $X_i$ is partitioned into $r_i$ segments by using indiscernibility relations. The activation function of this layer is Gaussian Function as follows:

$$\mu(i, j) = \exp \left(-\left(\frac{x_i - c_{ij}}{\sigma_{ij}}\right)^2\right)$$

where $c_{ij}$ represents the center of equivalence class; $\sigma_{ij}$ denotes the standard deviation of equivalence class, and $i = 1, 2, \cdots, n$; $j = 1, 2, \cdots, r_i$.

The third layer denotes the rules found by rough
set theory, whose connections to the nodes of adjacent layers are determined by the conditions and conclusions of the rule.

\[ a_i = \min(\mu(1, l_{i1}), \mu(2, l_{i2}), \ldots, \mu(n, l_{in})) \]

or

\[ a_i = \mu(1, l_{i1}) \times \mu(2, l_{i2}) \times \ldots \times \mu(n, l_{in}) \]

(10)

Fig. 1 Architecture of RBPNN

where \( i = 1, 2, \ldots, k \), and \( k \) is the number of rules; \( l_{ij} \in \{1, 2, \ldots, r_j \}, j = 1, 2, \ldots, n \).

The fourth layer denotes the equivalence classification of output variables. It corresponds to the conclusion of the rules.

\[ v(i, j) = \sum_{l=1}^{k} w_{ij} \alpha_l \]  

(11)

where \( i = 1, 2, \ldots, m \), and \( m \) equals the number of classes; \( j = 1, 2, \ldots, q_i \), and \( q_i \) is the number of segments into which the \( i \)th class is partitioned for fuzzy classification.

The fifth layer is the output layer, standing for the classification of objects.

\[ y_i = \sum_{j=1}^{q_i} v(i, j) w(i, j) \]  

(12)

where \( i = 1, 2, \ldots, m \).

The objective function can be defined as:

\[ E = \frac{1}{2} \sum_{i=1}^{l} (d_i - y_i)^2 \]  

(13)

During training, the weights connecting the third and the fourth layers or the fourth and the fifth layers or all of them can be updated by back propagating error. The rule employed for weight updating is:

\[ \frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial w_{ij}} = -(d_i - y_i) v(i, j) \]  

(14)

\[ w_{ij}(k + 1) = w_{ij}(k) - \beta \frac{\partial E}{\partial w_{ij}} \]  

(15)

where \( \beta > 0 \) is the learning rate; \( d_i \) denotes the decision attribute.

4 Application of RBPNN to remote sensing data classification

The procedure of remote sensing image classification using RBPNN based on rough sets will be described in detail as follows.

4.1 Discretization of input variables

Suppose the input variables are the gray value of bands. They must be discretized before composing the DT. One of approaches is using entropy method according to train dataset. The entropy of set (sample data) is defined as follows:

\[ E(S) = - \sum_{j=1}^{l} p_j \log p_j \]  

(16)

where \( l \) is the number of classes in the sample set; \( p_j \) is the percentage of the \( j \)th class in the set of sample data.

After the set being divided into two subsets by the point \( T \), the entropy of set introduced by the partition is defined as follows:

\[ E(T, S) = \frac{\text{Card}(S_1)}{\text{Card}(S)} E(S_1) + \frac{\text{Card}(S_2)}{\text{Card}(S)} E(S_2) \]  

(17)

where \( \text{Card}(B) \) represents the number of elements in the set \( B \). The Point \( T \) which makes \( E(T, S) \) get the minimal value is the best partition point.

The partition is going until the following equation is true for all the subsets \( S^k \).
\[ G_{\min}(T, S^k) \leq \frac{\log_2(N_k)}{N_k} + \frac{\nabla(T, S^k)}{N_k} \quad (18) \]

where \( k \) is the partition times; \( G_{\min}(T, S^k) = E(S^k) - E(T, S^k) \); \( N_k \) is the number of pixels in the subsets \( S^k \); and \( \nabla(T, S^k) = \log_2(3^k - 2) - [kE(S^k) - \text{Card}(S^k)E(S^k) - \text{Card}(S^k)E(S^k)] \).

More details are given in Reference [4].

4.2 Composing the DT

The decision table (Table 1) composed of the training data set include attributes (columns) and objects (rows). In this work, objects are sample pixels, and their attributes include condition attributes, gray values of bands (after discretization, \( B1-B7 \)) and decision attributes, the ground truth of sample pixels (class, \( C \)).

4.3 Reduction and extracting rules

The procedure of 2D reduction on DT is briefly introduced as follows:

1) Deleting the pixel if its gray value of bands and classes are equal to that of the previous pixels.

2) Computing the core of condition attributes using the method introduced in Section 2.3.

3) Deleting the redundant attributes according to the core.

4) Deleting the duplicate pixel again.

More details are given in Reference [5].

After employing 2D reduction (attribute and object reduction) on DT, the residual pixels and their attributes form the classification rules (rough sets rules). And the dependency factors of these rules are used to encode the initial connection weights of the network.

4.4 Network configuring and training

After getting the rules from the DT, we can configure initially the RBPNN. The number of nodes in the input layer is the number of the spectral bands. The number of nodes in the second layer equals to number of the sum of segment resulting from the processing of discretization. And the number of nodes in the third layer is set to the number of rules. The far end of the network is the output layer and its number is equal to the number of classes.

The fourth layer of the network is an equivalence classification of output variables. For the fuzzy classification, the output classes can be partitioned into some segments according to fuzzy membership. But for crisp classification, we need not partition the output variables. It means that the numbers of the fourth layer and the fifth layer are equal and the relation between them is determined.

The training of neural network is performed with the reduced DT. The connection between the first layer and the second layer is determined by the discretization (equivalence classification). The second layer simply passes the information via the Gaussian Function. There are no weights to be trained in these two connections. The network learns the weights connecting the third layer and the fourth layer. Using the back propagation algorithm, And for fuzzy classification, the weights connecting the fourth layer and the fifth layer should also be learned.

5 Experiment

In order to test the practicality of the approach mentioned above, a land cover classification experiment using RBPNN is done. The remote sensing data in this experiment comes from a seven bands Landsat TM image of the Sanxia area, Hubei province. The training data set includes 800 pixels selected from the original image. The testing data set consists of 400 pixels. The correct output classes of these two sets are provided by visual interpretation. The land cover is classified to six classes: \( C1 \) is forest land, \( C2 \) is grass land, \( C3 \) is agriculture land, \( C4 \) is bare land, \( C5 \) is water bodies, and \( C6 \) is built-up area.

The network uses seven input units and six output units, and each output unit represents a possible...
land cover class. The entropy method partitions the input variables into segments with a maximum of 28. The output variables are not separated because crisp classification is chosen in this experiment. After 2D reduction, rough sets methodology extracts 138 rules from the decision table. While the learning rate is set to 0.1, the network gets terminal condition after 168 training iterations. Table 2 shows the results of the RBPNN classifier.

To comparing the results, the conventional BPNN classifier is also performed with the same data set. The network consists of three layers, using the same number units of input layer and output layer. The number of the second layer units in BPNN is chosen to 14 based on similar application in the remote sensing field. The learning rate is set to 0.1. After 6,000 training iterations, the mean square error in the outputs converged to a minimum achievable with these settings. Table 3 shows the results from the conventional BPNN classifier.

| Truth | C1 | C2 | C3 | C4 | C5 | C6 | Σ |
|-------|----|----|----|----|----|----|----|
| C1    | 75 | 2  | 0  | 0  | 0  | 0  | 77 |
| C2    | 5  | 35 | 0  | 0  | 0  | 0  | 40 |
| C3    | 0  | 3  | 35 | 0  | 0  | 0  | 44 |
| C4    | 0  | 0  | 5  | 34 | 0  | 0  | 47 |
| C5    | 0  | 0  | 0  | 0  | 84 | 3  | 87 |
| C6    | 0  | 0  | 0  | 0  | 6  | 101| 107|
| Σ     | 80 | 40 | 40 | 70 | 90 | 110| 400|

Table 3 Classifications by RBPNN classifier (89%)

| Truth | C1 | C2 | C3 | C4 | C5 | C6 | Σ |
|-------|----|----|----|----|----|----|----|
| C1    | 71 | 0  | 0  | 0  | 0  | 0  | 71 |
| C2    | 6  | 37 | 0  | 0  | 0  | 0  | 43 |
| C3    | 3  | 3  | 34 | 7  | 0  | 0  | 47 |
| C4    | 0  | 0  | 6  | 33 | 0  | 0  | 43 |
| C5    | 0  | 0  | 0  | 0  | 83 | 9  | 91 |
| C6    | 0  | 0  | 0  | 0  | 7  | 98 | 105|
| Σ     | 80 | 40 | 40 | 90 | 90 | 110| 400|

On a HP LC2000, P 1111 800 CPU, 256 M memory, the BPNN based on rough sets takes 170 CPU seconds to train the network while the conventional BPNN 1,800 seconds. In other words, the BPNN based on rough sets extremely reduces the training time while keeping approximately the same accuracy with the conventional BPNN.

6 Conclusion

Integrating two computing tools, artificial neural networks and rough sets, a new method for remotely sensed data classification is presented. Using this method to classify remote sensing data has some advantages. First, the algorithm directly extracts rules from a decision table consisting of condition and decision attributes. Second, the learning rate is the only parameter of network to be set. Third, the connection weights and the fourth layer represents the credit of rules. Finally, the fourth layer denotes the equivalence classification of output variables. It makes the network easier to develop a fully fuzzy classifier.

References

1. Gopal S., Woodcock C. E., Strahler A. H. (1999) Fuzzy neural network classification of global land cover from a 1' AVHRR data set, Elsevier, Remote Sensing. Environment, 67: 230-243
2. Swiniarski R. W., Hargis L. (2001) Rough sets as a front and of neural networks texture classifiers. Elsevier, Neurocomputing, 36: 85-102
3. Cho Y., Lee K., Yoo J., et al. (1998) Autogeneration of fuzzy rules and membership functions for fuzzy modeling using rough set theory, IEEE proc. -Control Theory Appl., 145(5): 437-442
4. Sun L X, Gao W (1999) Hyperspectral band set global reduction based on rough sets and genetic algorithm. Journal of Wuhan Technical University of Surveying and Mapping, 24(4): 306-311 (in Chinese)
5. Ying Y., Cao C X, Zhang B L (2000) Classification rule discovery based on rough set theory. Journal of Chongqing University (Natural Science Edition), 23(1): 62-65 (in Chinese)