Rough Neural Intelligent Approach for Classification

Sanaa Rashed Abdallah and Yasser Fouad Hassan
Department of Mathematics (computer science),
Faculty of Science, Alex University

This paper describes rough neural network where neural network systems and rough sets theory are completely integrated into a hybrid system and are used cooperatively for decision and classification support. Also, rough sets and neural network are chosen for the combined method because they can discover patterns in ambiguous and imperfect data, and provide tools for data and pattern analysis. The common characteristic of rough sets and neural networks is that both approaches have the ability to learn decision models by examples.

1 Introduction

Intelligent systems comprise various paradigms dedicated to approximately solving real-world problems, e.g., in decision making, classification or learning; among these paradigms are fuzzy sets, rough sets, neural networks, and genetic algorithms. Fuzzy sets provide a natural framework for the process in dealing with uncertainty. It offers a problem-solving tool between the precision of classical mathematics and the inherent imprecision of the real world. The imprecision in an image contained within color value can be handled using fuzzy sets. Neural networks and rough sets are widely used for classification and rule generation. Genetic algorithms (GAs) are involved in various optimization and search processes, like query optimization and template selection. Other approaches like case based reasoning and decision trees are also widely used to solve data analysis problems. Each one of these techniques has its own properties and features including their ability of finding important rules and information that could be useful for data classification.Neural networks provide a robust approach to approximating real-valued, discrete-valued and vector-valued functions. The well-known algorithm Back-propagation, which uses gradient descent to tune network parameters to best fit the training set with input-output pair, has been applied as a learning technique for the neural networks. Rough sets based systems provide domain knowledge expressed in the form of If-then rules and tools for data analysis. Unlike other intelligent systems, rough set analysis requires no external parameters and uses only the information presented in the given data. The combination or integration of more distinct methodologies can be done in any form, either by a modular integration of two or more intelligent methodologies, which maintains the identity of each methodology, or by integrate one methodology into another, or by transforming the knowledge representation in one methodology into another form of representation, characteristic to another methodology. This paper introduces an overview of the rough neural hybrid approach for decision making.

The paper is organized as follow. Section 2, gives a brief introduction to the basic of the methods, namely rough sets and rough neural network. Section 3, describes the rough neural model. Conclusion is given in section 4.

2 Preliminary: Intelligent Techniques

Recently various intelligent techniques and approaches have been applied to handle the different challenges posed by data analysis. The main constituents of intelligent systems include fuzzy logic, neural networks, genetic algorithms, and rough sets. Each of them contributes a distinct methodology for addressing problems in its domain. This is done in a cooperative, rather than a competitive, manner. The result is a more intelligent and robust system providing a human-interpretable, low cost, approximate solution, as compared to traditional techniques. Rough set theory is a relatively new intelligent technique used in the discovery of data dependencies; it evaluates the importance of attributes, discovers the patterns of data, reduces all redundant objects and attributes, and seeks the minimum subset of attributes. Moreover, it is being used for the extraction of rules from databases.

2.1 Rough sets

Rough sets theory has been proposed by Professor Pawlak a new intelligent mathematical tool for extracting classification rules from uncertain and incomplete data-based information [7,8]. It is based on the concept of an upper and a lower approximation of a set, the approximation space and models of sets. Unlike other intelligent methods, rough set analysis requires no external parameters and uses only the information presented in the given data.

An information system (IS) is an ordered pair (U, A), where $U = \{x_1, x_2, \ldots, x_n\}$ is a nonempty finite set of objects called the universe, and $A = \{a_1, a_2, \ldots, a_n\}$is a nonempty set and the elements of A, called attributes (in our case called image features). Decision

Table 1: An example of decision table

	a_1	a_2	a_3	D
x_1	0	0	1	0
x_2	1	1	1	0
x_3	0	2	1	0
x_4	1	2	1	0
x_5	1	0	1	0
x_6	1	2	1	1
x_7	0	0	1	1

System is an information system (IS) for which the attributes in A are further classified into disjoint sets of condition attributes C and decision attributes D..

A small example of decision table can be found in Table 1. The table has seven objects, where a_1, a_2, and a_3 are conational attributes (i.e., image features in our case) and d is the decision attributes.

The discernibility matrix of A is the nxn matrix with (i,j)th entry defined as follows:

$$DM_{ij} = \{a \in A : a(x_i) \neq a(x_j)\}. \tag{1}$$

The discernibility matrix contains all the attributes that differentiate between two given objects x_1 and x_2 Every subsets of attributes of P is associated an indiscernible on on U is defined as follows:

$$I_p = \{(x, y) \in UxU : a(x) = a(y), \forall a \in P\} \tag{2}$$

Where U/I_P is the set of all equivalence classes in the relation I_P, we say that the objects x and y are $P-$indiscernible if $(x, y) \in I_p$.

In Table 1, objects x_1 and x_7 are indiscernible by attributes a_1, a_2 and a_3.

The partition constructed by attributes $\{a_1, a_2, a_3\}$ for the objects in Table 1 is: $\{\{x_1, x_7\}, \{x_2\}, \{x_3\}, \{x_4, x_6\}, \{x_5\}\}$

Due to imprecision which existed in the real world data, there are always conflicting objects contained in a decision table. Here conflicting objects refer to the two or more objects that are indiscernible by employing any set of condition attributes, but they belong

$$\underline{P}X = \{x_2, x_3, x_5\}, \quad \overline{P}X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}, \quad BN_P(X) = \{x_1, x_4, x_6, x_7\}$$

to different decision classes. Such objects are called inconsistent; therefore, the decision table is called inconsistent decision table.

Now a new partition of the universe U can be found by the indiscernibility relation. Let $IS = (U, A)$ is an information system, and let $P \subseteq A$ and $X \subseteq U$. X can be approximated using only the information contained in P by constructing B-lower and B-upper approximation of X. These basic operations in rough sets theory are defined as follows:

$$\underline{P}X = \bigcup\{Y : Y \in U/I_p, Y \subseteq X\} \tag{3}$$

$$\overline{P}X = \bigcup\{Y : Y \in U/I_p, Y \cup X \neq \phi\} \tag{4}$$

Where $\underline{P}X$ is the set of all objects of U that cab be certainly classified by set P as a members of X and $\overline{P}X$ is a set of objects that can be probably classified by P as members of X. The set

$$BN_P(X) = \overline{P}X - \underline{P}X \tag{5}$$

is referred to as the B-boundary region of X and thus consists of those objects that cannot be classified into X on the basis of knowledge X.

As a measure of quality of a partition approximation by attribute set B and a deacons attribute d. It takes the following form:

$$\gamma(B, d) = \frac{\sum_1^n (card\underline{P}X)}{card(U)} \tag{6}$$

Where $card$ denotes the cardinality. It is expresses the ratio of elements that can be properly classified employing attributes in B to all elements of the universe. If $\gamma(B, d) = 1$, it is said that d depend totally on B and if $\gamma(B, d) < 1$, it is said that d depends partially on B. An information system may contain unnecessary attributes. That means, that all conditions attributes are not needed to describe dependencies between conational and decisional attributes. For example, the reduce set of attributes in table 1 is $\{a_1, a_2\}$.
The simplest way of rule generation is to interpret each row of the reduced decision table as

a rule (i.e. the values of ci=ondition attributes imply a certain value of decision attribute. For example, the first row in table 1 can be read as follows; if $a_1 = 0$ and $a_2 = 0$ then $d = 0$

Given a classification task mapping a set of variables C to a set of labeling D, a reduct is defined as any $R \subseteq C$, such that $\gamma(C, D) = \gamma(R, D)$. The set of all reducts of A is denoted $Red(A)$. An information may have more than one reduct.

An attribute $C_j \in C$ is a core attribute in C with respect to D if $Lower_{[C]/[D]} \neq Lower_{[C-C_j][D]}$. It is the intersection of all reducts:

$$Core(C) = \bigcap_{R_i \in Red(B)} Red_i, i = 1, 2, \ldots \tag{7}$$

It is now possible to define the significance of an attribute. This is done by calculating the change of dependency when removing the attribute from the set of considered conditional attributes.
Given P, Q and an object $x \in P$, the significant $\sigma_x(P, Q)$ of x in the equivalence relation denoted P and Q is $\sigma_x(P, Q) = \gamma(P, Q) - \gamma(P - \{x\}, Q)$.
Now, attribute reduction involves removing attributes that have no significance to the classification at hand.

2.2 Neural Networks

The effectiveness of artificial neural networks as tools that aid human decision-making in the medical field has been reported in many recent papers. Experimental results indicate that neural networks perform particularly well in solving complex pattern classification problems, due to their ability to model nonlinear relationships. Neural networks are also robust in handling data with noise or missing values, due to their inherently parallel data processing. There are a wide variety of areas in which artificial neural networks have been applied to problems in the sciences. These include pattern recognition, optimal control, adaptive filtering, inversion, target tracking, general purpose modeling, and medical diagnosis. Detailed theory and applications of are readily available in the literature [9].One of the most-used types of ANN architecture is the Feed-forward with the Back-Propagation Neural Network (FFBNN), as shown in Figure (1). The signals flow from neurons in the input layer to the neurons in the output layer, passing through the hidden neurons where there could be more than one hidden layer.

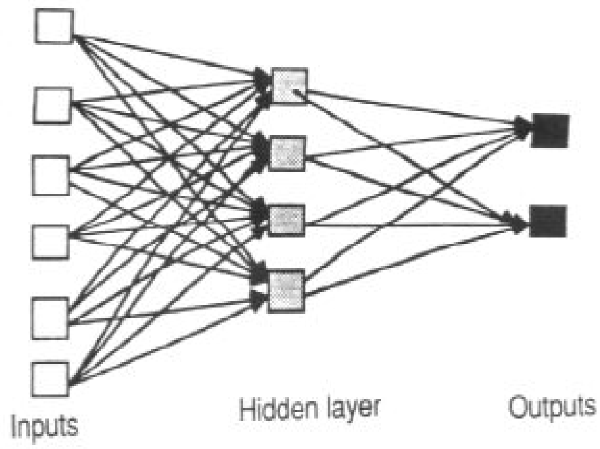

Figure 1: ANN architecture

In this subsection, we only briefly outline the main facts about networks that will be further discussed. In most of the applications presented in this work the classical multilayer feed forward network as described in [3] is utilized. The most commonly used learning algorithm is back-propagation. The signals flow from neurons in the input layer to the neurons in the output layer, passing through the hidden neurons where there could be more than one hidden layer.

By a sigmoidal excitation function for a neuron we will understand a mapping of the form:

$$f(x) = \frac{1}{1 + e^{-\beta x}} \tag{8}$$

where x represents weighted sum of inputs for a given neuron and β is the coefficient called gain, which determines the slope of the function.

Let Icn_i,Ocn_j and w_{ij} are input of the conventional neuron i, output from neuron j, and weight of the connection between neuron i and neuron j, respectively. Icn_i,Ocn_j and w_{ij} are defined as follows:

$$Icn_i = \sum_{j=1}^{n} w_{ij} Ocn_j \tag{9}$$

Neuron i is connected to neuron j.

$$Ocn_i = f(Icn_i) \tag{10}$$

Where f is the sigmoid function defined in equation (8).

3 Rough neural network: rough neuron

Rough neural networks [2,3,6,9,10]used in this study consist of one input layer, one output layer and one hidden layer. The input layer neurons accept input from the external environment. The outputs from input layer neurons are feed to the hidden layer neurons. The hidden layer neurons feed their output to the output layer neurons which send their output to the external environment.

The number of hidden neurons is determined by the following equation:

$$N_{hn} \leq \frac{N_{ts} * T_e * N_f}{N_f + N_o} \tag{11}$$

Where N_{hn} is the number of hidden neurons, N_{ts} is the number of training samples, T_e is the tolerance error, N_f is the number of attributes (features), and N_o is the number of the output. The output of a rough neuron is a pair of upper and lower bounds, while the output of a conventional neuron is a single value.

3.0.1 characteristic of rough neuron

Rough neuron was introduced in 1996 by Lingras [2] It was defined relative to upper bound (U_n), lower bound (L_n) and inputs were assessed relative to boundary values. Rough neuron has three type of connections:

Step 1. Input-Output connection to U_n
Step 2. Input-Output connection to L_n
Step 3. Connection between U_n and L_n

A rough neuron R_n is a pair of usual rough neurons $R_n = (U_n, L_n)$, where U_n and L_n are the upper rough neuron and the lower rough neuron, respectively.

Let (Ir_{L_n}, Or_{L_n}) is the input/output of the lower rough neuron and (Ir_{U_n}, Or_{U_n}) is the input/output of the upper rough neuron. The calculation of the input/output of the lower/upper rough neuron is given by the following equations:

$$Ir_{L_n} = \sum_{j=1}^{n} w_{L_{nj}} On_j \tag{12}$$

$$Ir_{U_n} = \sum_{j=1}^{n} w_{U_{nj}} On_j \tag{13}$$

$$Or_{L_n} = min(f(Ir_{L_n}), f(Ir_{U_n}))) \tag{14}$$

$$Or_{U_n} = max(f(Ir_{L_n}), f(Ir_{U_n}))) \tag{15}$$

The output of the rough neuron (O_{rn}) will be computed using the following equation:

$$O_{rn} = \frac{Or_{U_n} - Or_{L_n}}{avarge(Or_{U_n}, Or_{L_n})} \tag{16}$$

The basic stricture of rough neural network is given in Figure (2).

The rough neural network classification algorithm is described as follows Algorithm-3]:

Algorithm-1: The classification algorithm

Input: A new data to be classified, set of features (i.e., attributes), set of neurons inputs and the set of rules
Processing:

Step-1 For each attribute in the attribute set Do

Step-2 Compute the upper and lower rough neuron

Step-3 Build rough neural networks

Step-4 Compute the relative error

Step-5 Calibrate the rough neural network

Step-6 Repeat 4 and 5 until the error become minimum

Step-7 Return Class with minimum error.

Output: The final classification

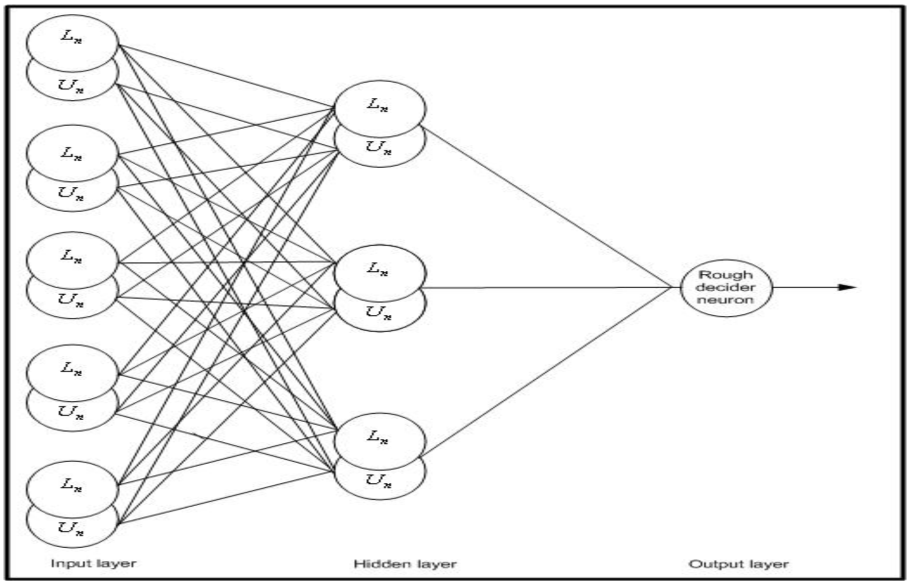

Figure 2: Rough neural network

This study proposed rough sets theory, rough neural networks for estimating rough output patterns from rough input patterns. A rough pattern uses upper and lower bounds of the values as opposed to precise values. The rough neural networks use a combination of rough and conventional neurons. A rough neuron can be viewed as a pair of neurons. One neuron corresponds to the upper bound and the other corresponds to the lower bound. Upper and lower neuron exchange information with each other during the calculation of their outputs. The paper discussed different types of connections to and from rough neurons. The errors in estimation from rough neural network models are significantly lower than the conventional neural network model. Moreover, the addition of rough neurons in hidden layer seems to improve the prediction performance.

4 Summary and Conclusion

This study proposed rough sets theory, rough neural networks for estimating rough output patterns from rough input patterns. A rough pattern uses upper and lower bounds of the values as opposed to precise values. The rough neural networks use a combination of rough and conventional neurons. A rough neuron can be viewed as a pair of neurons. One neuron corresponds to the upper bound and the other corresponds to the lower bound. Upper and lower neuron exchange information with each other during the calculation of their outputs. The paper discussed different types of connections to and from rough neurons. The errors in estimation from rough neural network models are significantly lower than the conventional neural network model. Moreover, the addition of rough neurons in hidden layer seems to improve the prediction performance.

References

[1] Grzymala-Busse J. Pawlak Z. Slowinski R. and Ziarko W. (1999) Rough Sets. In Communications of the ACM, vol.38, no. 11.pp.

[2] Lingras P. J. (1996) Rough neural networks. In: Proc. of the 6th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU96), Granada, Spain,pp.1445-1450.

[3] Aboul Ella Hassanien (2006) Rough Neural Intelligent Approach for Image Classification: A Case of Patients with Suspected Breast Cancer. International Journal of Hybrid Intelligent System, IOS press, 2006 (to appear)

[4] Ning S. Xiaohua H. Ziarko W. and Cercone N. (1994) A Generalized Rough Sets Model. In Proceedings of the 3rd Pacific Rim International Conference on Artificial Intelligence, Beijing, China, Int. Acad. Publishers, Vol. 431, pp. 437-443.

[5] Ning S. Ziarko W. Hamilton J. and Cercone N. (1995) Using Rough Sets as Tools for Knowledge Discovery. KDD'95 Proceedings First International Conference on Knowledge Discovery Data Mining, U.M. Fayyad, R. Uthurusamy (eds.), Montreal, Que., Canada, AAAI, pp. 263-268.

[6] Pal S.K. Polkowski S.K. and Skowron A. (Eds.) (2002) Rough-Neuro Computing: Techniques for Computing with Words. Berlin: Springer-Verlag.

[7] Pawlak Z. (1982) Rough Sets. Int. J. Computer and Information Sci., Vol. 11, pp. 341–356,

[8] Pawlak Z. Grzymala-Busse J. Slowinski R. and Ziarko W. (1995) Rough sets. Communications of the ACM, vol. 38, no. 11, pp. 89-95.

[9] Peters J.F. Liting H. and Ramanna S. (2001) Rough Neural Computing in Signal Analysis. Computational Intelligence vol. 17, no.3: pp. 493-513.

[10] Peters, J.F. Andrzej Skowron, Liting H. and Ramanna S. (2000) Towards Rough Neural Computing Based on Rough Membership Functions: Theory and Application. Rough Sets and Current Trends in Computing 2000, pp. 611-618

[11] Setiono R.(2000)Generating concise and accurate classification rules for breast cancer diagnosis. Artificial Intelligence in Medicine, vol. 18, no. 3, pp. 205-219.

[12] Slowinski R. (1993) Rough set approach to decision analysis. AI Expert, pp. 19-25, March 1995. 27. Stefanowski J., "Classification support based on the rough sets" Foundations of Computing and Decision Sciences, vol. 18, no. 3-4, pp. 371-380.