Research on Classification of E-shopper Based on Neural Networks and Genetic Algorithm

Chong Wang
Business School, Huaihai Institute of Technology, Lianyungang, 222000, China
wangchongwang0916@yahoo.com.cn

Jian Liu
Library, Huaihai Institute of Technology, Lianyungang, 222000, China
liujian0305@yahoo.com.cn

Yanqing Wang
School of Management, Harbin Institute of Technology, Harbin, 150001, China
yanqing@hit.edu.cn

Abstract—With the rapid development of online shopping, the ability to intelligently collect and analyze information about E-shoppers has become a key source of competitive advantage for firms. This paper presents a optimal algorithm of modeling dynamic architecture for artificial neural networks (ANN) and a novel machine-learning algorithm for extracting rules from databases via using genetic algorithm. In the dynamic architecture, the number of hidden layers and the number of hidden nodes are sequentially and dynamically generated until a level of performance accuracy is reached. In addition, in this paper, a new genetic algorithm is presented, which does not need the computational complexity. The genetic algorithm is used to find the optimal values of input attributes X_m , which maximizes output function φ_k of output node k. The optimal chromosome is decoded and used to obtain a rule belonging to class k. The better result is achieved by applying the two new algorithms to a given database for e-shoppers buying computer.

Index Terms—e-shopper, neural network, genetic algorithm, rule extraction

I. INTRODUCTION

The advent of the Internet is arguably one of the most important innovations related to the field of marketing since 90's in the 20th century. Its diffusion rate has been quite amazing. The number of Internet users in China went up from 0.62 million in October 1997 to 400 million more than in June 2010, and 33.8 % Internet users often purchase merchandise via Internet[1]. Electronic Commerce enjoys a steady growth rate (about 10% per year) in our country. It becomes more and more familiar mode of shopping for many consumers. An understanding of e-shopper's purchase behavior has gotten more and more important for marketers in order to succeed in Internet marketing.

Marketers endeavor to learn customer's behavior of purchasing to develop marketing strategies to create new consuming markets, and discover latent loyal customers. Therefore, it is vital for companies to try and understand the nature of heterogeneity in consumer preferences. The web has made geographical boundaries weaker than ever before. This is one of the difficulties why e-firm is faced with in dealing with e-shoppers who might be quite different from one another in many ways. The ability to provide content and services to individuals on the basis of knowledge about their preferences and behavior has become an important marketing tool for marketers in Internet environment [2].

Complete customer information includes two parts: factual and behavioral information [3]. The factual information contains name, gender, and date of birth. Behavioral information describes customer's actions and reveals customer's preferences, which are usually derived from transactional data. This data might include histories of customers, web purchasing and browsing activities, as well as demographic and psychological information. The purchase transaction records of a customer for a certain period are used to build a customer profile[4] describing his or her likes and dislikes. Therefore, analyzing factual and behavioral information of e-shopper can contribute to discovering customer's preferences and purchase behavior.

Data mining, one of the most popular techniques, has mostly been adopted to generate predictions and describe behaviors from transaction databases in support of making better decisions [5]. Knowledge relating to e-shopper's purchase behaviors can be discovered from transaction databases by mining techniques. Data mining can discover potentially significant patterns and rules underlying the database by automatic or semi-automatic exploration and analysis on large amounts of data items set in a database [6]. The patterns and rules are mostly the habits of purchasing and other consumers' behavior. During the past decade, there have been a variety of significant developments. Some of the developments are implemented in customized service to develop customer relationship [7]. For example, businesses can get to know customers' wants and needs by utilizing data mining technology to discover potential information, and accordingly develop more appropriate marketing strategies to attract and retain customers in electronic commerce.

doi:10.4304/jcp.6.4.627-634

Predicting a customer's purchase behavior through extracting rules from a given database has already become an important marketing means[8]. There have been many such methods[9], for example, decision trees, association rules, and neural network etc are usually used to mine e-shopper's transaction database in order to classify consumers. Decision tree is a classification-based approach. Association rules mining that was first proposed by Agrawal and Srikant can discover correlations of events, which can be represented as probabilistic rules [10]. And artificial neural network is the important one of the methods[11]. In fact, neural networks[12] offer a novel technique that does not require a pre-specification during the modeling process because they independently learn the relationship inherent in the variables. So far, many methods[13] that extract rules from a given database via artificial neural networks have been proposed. Artificial neural networks (ANN), have shown to be an effective approach for pattern recognition, classification, clustering and prediction. Most ANN models use a multi-layer network with one or more hidden layers. The choice of the optimization algorithm and guidelines for selecting the parameter values to yield a better model have been studied and reported upon extensively in literature [14-16]. Chester [17] and Zhang [18] studies suggest that the ideal number of hidden layers in FFBP architectures is often two or three. S. Kang [19] have suggested n/2 nodes for each hidden layer, where n is the number of input nodes.

Although a good number of efficient algorithms for extracting rules have been proposed, there still is much space to improve the algorithms. There is no algorithm which can be applied to any type of networks, to any training algorithm, and to both discrete and continuous values. The parameters of the model such as the choice of input nodes, number of hidden layers, number of hidden nodes (in each hidden layer), and the form of transfer functions, are problem dependent and often require trial to find the best model for a particular application [20-21]. The present algorithms for artificial neural networks have various questions such as computational complexity, low accuracy and narrow scope of application. A method for extracting M-of-N rules from trained artificial neural networks was presented [22]. The algorithm was based on the standard three—layered feed forward networks. However, the attributes of the database are assumed to have binary values −1 or 1. A decomposition algorithm that can be applied to multi-layer ANN and recurrent networks had been presented [23]. But the computational complexity of the approximation is exponential. An approach for extracting rules from trained ANN for regression was presented [24]. Each rule in the extracted rule set corresponds to sub-region of the input space and a linear function. However, the method extracts rules from trained ANN by approximating the hidden activation function. This approximation yields to less accuracy and makes the computation burdensome.

Basing on the advantages and shortcomings of the mentioned algorithms above, an optimal algorithm of modeling dynamic architecture for artificial neural networks and a new algorithm for extracting rules via neural network using genetic algorithm are presented in this paper. Comparing with classical structure of ANN, a dynamic Architecture of ANN presents optimal performance accuracy. The genetic algorithm takes all input attributes into consideration and does not make any approximation to the hidden unit activation function, so produces accurate rules. It also has the less computational complexity.

This paper is organized as follows. Section 2 presents the methodology, which includes model architecture and formulation of weights and the genetic algorithm. Section 3 illustrates an example and segments network customers based on the presented method. Finally, Section 4 presents summary and conclusion.

II. DESCRIPTION OF THE METHODOLOGY

First architecture of neural network need be determined before using artificial neural network to classify e-shoppers. The architecture includes the number of input nodes, output nodes, hidden layers and hidden nodes (in each hidden layer), as well as the form of transfer functions.

A. Modeling Architecture

Input layer for ANN accepts external data to the model. So, the number of input layer nodes is determined according to external data. Assume that these records include N attributes. Each attribute, A_n ($n = 1, 2, \ldots, N$), can be encoded into a fixed length binary sub-string $\{x_1 \ldots x_i \ldots x_{mn}\}$, where m_n is the number of possible values for an attribute A_n. The element $x_i = 1$ if its corresponding attribute value exists, while all the other elements = 0. So, the proposed number of input nodes, I, in the input layer of ANN can be given by

$$I=\sum_{n=1}^{N} m_n \tag{1}$$

The input attributes vectors, X_m, to the input layer can be rewritten as:

$$X_m = \{x_1 \ldots x_i \ldots x_I\}_m \tag{2}$$

Where $m = (1, 2, \ldots, M)$, M is the total number of input training patterns.

The next modeling decision is the choice of the number of hidden layers and hidden nodes. In this model, unlike classical artificial neural networks, the number of hidden layers and hidden nodes is not fixed a priori, but are sequentially and dynamically generated until a level of performance accuracy is reached. According to achieved studies and the experiments on ANN, the number of hidden layers is determined within the scope of 2-3, and the number of hidden nodes for each hidden layer is determined within the scope of (n-2)/2−(n+2)/2. The optimal number of hidden layers and hidden nodes is generated dynamically.

The output class vector, Y_k $(k = 1, 2, \ldots, K)$, can be encoded as a bit vector of a fixed length K as follows:

$$Y_k\{\varphi_1 \ldots \varphi_k \ldots \varphi_K\} \tag{3}$$

Where K is the number of different possible classes. If the output vector belongs to class k, then the element φ_k is equal to 1 while all the other elements in the vector are zeros. Therefore, the proposed number of output nodes in the output layer of ANN is K.

The optimal algorithm is as follows:

Begin

{

{Separate database into input vectors (X_m) and corresponding output vectors (Y_m). Database is coded as bit string.

The number of Input nodes, $I=\sum_{n=1}^{N} m_n$

The number of output nodes, K is equal to the number of different possible classes

}

For H=2 to 3 do

Where H is the number of hidden layers.

{For (J=n-2/2; J ⩽ n+2/2; n++)

{

While (termination conditions not satisfied) do

{$Z = Z + 1$

Where Z is the number of iteration.

Train ANN on the encoded vectors of the input attributes and the corresponding vectors of the output classes}

Obtain groups of weights

}

End for

Create structure of ANN with optimal the parameters

}

End.

Accordingly, the structure of ANN is determined and is shown in Fig. 1. The training of ANN is processed until the convergence rate between the actual and the desired output will be achieved. The convergence rate can be improved by changing the number of hidden layers (H) and hidden nodes (J), the number of iterations (Z), the learning rate, and the momentum rate.

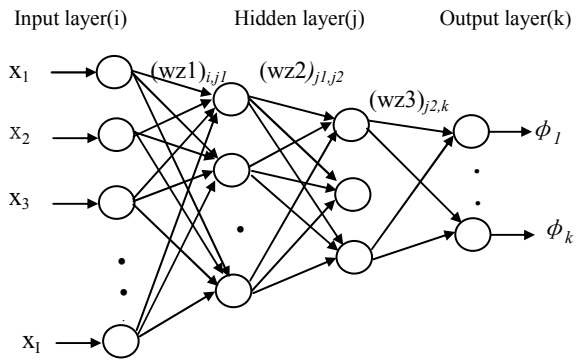

Figure1. ANN with two hidden layers

B. Formulation of Weights and the Genetic Algorithm

After training the ANN, three groups of weights can be obtained. The first group, $(WZ1)_{i,\,j1}$ includes the weights between the input node i and the hidden node j_1. The second group, $(WZ2)_{j1,j2}$ includes the weights between the hidden node j_1 and the hidden node j_2. The third group, $(WZ3)_{j3,\,k}$ includes the weights between the hidden node j_3 and the output node k. The activation function used in the hidden and output nodes of the ANN is a sigmoid function, which segment conveniently classes of consumers.

The total input to the hidden node $j1$, IPT_{j1} is given by;

$$IPT_{j1}=\sum_{i=1}^{I} x_i(wz1)_{i,\,j1} \tag{4}$$

The output of the hidden node $j1$, OPT_{j1}, is given by;

$$OPT_{j1}=\frac{1}{1+e^{-\left[\sum_{i=1}^{I} x_i(wz1)_{i,j1}\right]}} \tag{5}$$

The total input to the hidden node $j2$, IPN_{j2}, is given by;

$$IPN_{j2}=\sum_{j=1}^{J}(wz2)_{j1,\,j2}\,\frac{1}{1+e^{-\left[\sum_{i=1}^{I} x_i(wz1)_{i,j1}\right]}} \tag{6}$$

The output of the hidden node $j2$, OPT_{j2}, is given by;

$$OPT_{j2}=\frac{1}{1+e^{-\left[\sum_{j=1}^{J}(wz2)_{j1,j2}\left(1\big/1+e^{-\left[\sum_{i=1}^{I} x_i(wz1)_{i,j1}\right]}\right)\right]}} \tag{7}$$

The total input to the kth output node, IPN_k, is given by;

$$IPN_k=\sum_{j=1}^{J}(wz3)_{j2,\,k}\,.\,OPT_{j2} \tag{8}$$

So, the final value of the kth output node, $\varphi k(x_i)$, is given by;

$$\varphi k=\frac{1}{1+e^{-IPN_K}} \tag{9}$$

The function, $\varphi_k = f(x_i,(WZ1)_{i,j1}, (WZ2)_{j1,j2}, (WZ2)_{j2,k})$ is an exponential function in x_i since $(WZ1)_{i,j1}$, $(WZ2)_{j1,j2}$ and $(WZ3)_{j2,k}$ are constants. Its maximum output value is equal one, $\varphi_k = 1$, if an input vector, X_m, belongs to a class k, and all other elements is equal to zero in Y_k.

From the equation (9) above, one must find the input vector x_m that maximizes φ_k in order to extract relation rule between the input attributes, x_m relating to a specific class k. This is an optimization problem and can be stated as:

Maximize $\varphi k(x_i)$

Subjected to:

x_i is binary value(0 or 1) (10)

Since the objective function $\varphi k(x_i)$ is nonlinear and the constraints are binary. So, it is a nonlinear integer optimization problem. The genetic algorithm (GA) can be used to solve it as a dominant input selector because GA is theoretically and empirically proven to provide robust search capabilities in complex spaces. GA conforms to survival evolution principles for the fittest to live, the better to be accepted and the worse to be eliminated throughout natural selections. As shown in Fig.2, the process of GA is the combination of artificial crossovers, mutations and selections. GA searches for the optimal solution by operating population of individuals. The initial population is randomly generated.

The following algorithm explains how GA can be used to obtain the best chromosome.

Begin

{

Assume the fitness function as $\varphi_k(x_i)$

{

Generate number of slots equal to I, which represent input vector X_m.

Put a random value 0 or 1 in each slot}

Z= 0;

where Z is the number of generation.

Create initial population (S chromosomes);

$P(s)^Z$;

where s = 1 to S. S is population size.

Calculate $\varphi_k(x_i)$, $k = 1,2 \ldots K$;

Fitness function $\varphi_f(x_i)$ = Max($\varphi_1, \varphi_2, \ldots \varphi_K$)

Evaluate the fitness function according to $\varphi_f(x_i)$;

While (termination conditions not satisfied)

Do {$Z = Z + 1$;

Selection operation to $P(s)^Z$;

Crossover operation to $P(s)^Z$;

Mutation operation to $P(s)^Z$;

Modify the population from $P(s)^{Z-1}$ to $P(s)^Z$;

Evaluate the fitness function according to $\varphi_f(x_i)$;

}

Display the best chromosome that satisfies the conditions}

End.

The overall process of extracting rules is shown in Fig.2.

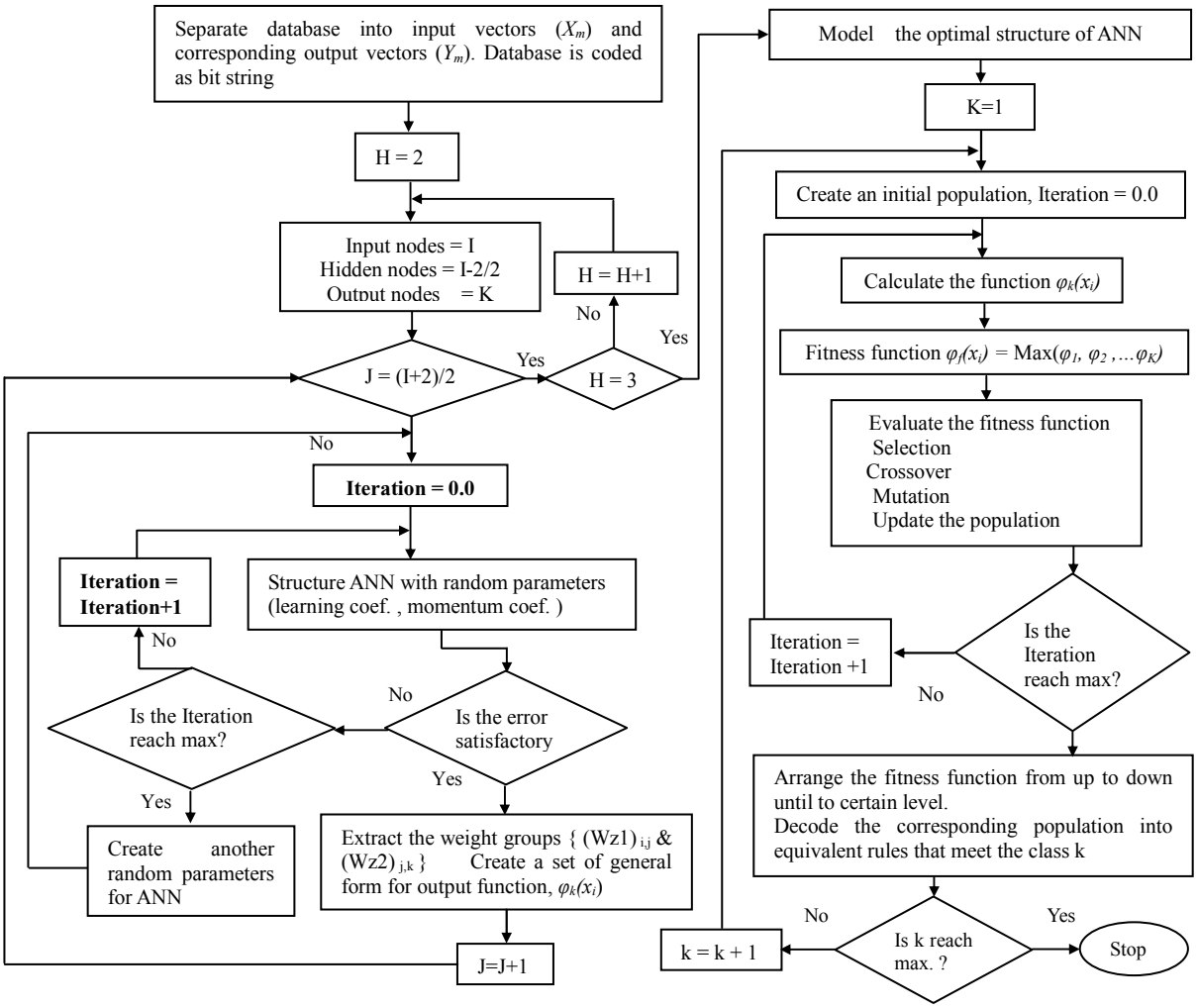

Figure 2. Overall process of rules extraction by GA on trained artificial neural networks

III. APPLICATION AND RESULTS

It is clear that the individual user behavior will act based on his/her preferences, attitude and personality. Each individual behavior is different from the others. The customers' preferences, attitude and personality can be found through purchasing products or services. Therefore, mining customers' factual and transactional data can reveal customers' purchase behavior in electric commerce, and discover e-shopper's preferences. A proposed record for products and inventories have the following attributes: product name, color, store size, city, month, quantity, quantity sold, and profit. A record for factual data includes: customer ID, customer name, gender, birth date, and nationality. A record for

transactional data may include the attributes: customer ID, date, time, store, product, and coupon used.

A given database for e-shoppers buying computer is shown in Table1. The given database includes twenty-eight transactional records that have four attributes and two different output classes. Firstly, the attributes and the output classes were encoded. The encoding values of the given database are shown in Table 2. Then extracting rules from the transaction database is carried out by using the method proposed above. Accordingly, we discover e-shoppers' preferences, and classify e-shoppers basing on their purchase behavior.

TABLE I.
EXAMPLE FOR SHOPPERS BUYING COMPUTER

ID	Age	Income	Student	Credit-rating	Buying	ID	Age	Income	Student	Credit-rating	Buying
1	＜30	High	No	Fair	No	15	＜30	Medium	No	Fair	No
2	＜30	High	No	Excellent	No	16	＜30	Low	Yes	Fair	Yes
3	30~40	High	No	Fair	Yes	17	＞40	Medium	Yes	Fair	Yes
4	＞40	Medium	No	Fair	Yes	18	＜30	Medium	Yes	Excellent	Yes
5	＞40	Low	Yes	Fair	Yes	19	30~40	Medium	No	Excellent	Yes
6	＞40	Low	Yes	Excellent	No	20	30~40	High	Yes	Fair	Yes
7	30~40	Low	Yes	Excellent	Yes	21	＞40	Medium	No	Excellent	No
8	30~40	High	No	Fair	Yes	22	＞40	Medium	Yes	Fair	Yes
9	＞40	Medium	No	Fair	Yes	23	＜30	Medium	Yes	Excellent	Yes
10	＜30	High	No	Fair	No	24	＜30	Medium	No	Fair	No
11	＜30	High	No	Excellent	No	25	＜30	Low	Yes	Fair	Yes
12	＞40	Low	Yes	Excellent	No	26	30~40	High	Yes	Fair	Yes
13	30~40	Low	Yes	Excellent	Yes	27	＞40	Medium	No	Excellent	No
14	＞40	Low	Yes	Fair	Yes	28	30~40	High	No	Fair	Yes

TABLE II.
ENCODING DATABASE

i/p pat X_m	Age $m_1 = 3$			Income $m_2=3$			Student $m_3=2$		Credit-rating $m_4=2$		O/P patt	Buys-computer	
	＜30 X_1	30~40 X_2	＞40 X_3	High X_4	Medium X_5	Low X_6	No X_7	Yes X_8	Fair X_9	Excellent X_{10}	Y_m	φ_1 No	φ_2 Yes
X_1	1	0	0	1	0	0	1	0	1	0	Y_1	0	1
X_2	1	0	0	1	0	0	1	0	0	1	Y_2	1	0
X_3	0	1	0	1	0	0	1	0	1	0	Y_3	0	1
X_4	0	0	1	0	1	0	1	0	1	0	Y_4	0	1
X_5	0	0	1	0	0	1	0	1	1	0	Y_5	0	1
X_6	0	0	1	0	0	1	0	1	0	1	Y_6	1	0
X_7	0	1	0	0	0	1	0	1	0	1	Y_7	0	1
X_8	0	1	0	1	0	0	1	0	1	0	Y_8	0	1
X_9	0	0	1	0	1	0	1	0	1	0	Y_9	0	1
X_{10}	1	0	0	1	0	0	1	0	1	0	Y_{10}	0	1
X_{11}	1	0	0	1	0	0	1	0	0	1	Y_{11}	1	0
X_{12}	0	0	1	0	0	1	0	1	0	1	Y_{12}	1	0
X_{13}	0	1	0	0	0	1	0	1	0	1	Y_{13}	0	1
X_{14}	1	0	0	0	1	0	1	0	1	0	Y_{14}	1	0
X_{15}	1	0	0	0	0	1	0	1	1	0	Y_{15}	0	1
X_{16}	0	0	1	0	1	0	0	1	1	0	Y_{16}	0	1
X_{17}	1	0	0	0	1	0	0	1	0	1	Y_{17}	0	1
X_{18}	0	1	0	0	1	0	1	0	0	1	Y_{18}	0	1
X_{19}	0	1	0	1	0	0	0	1	1	0	Y_{19}	0	1
X_{20}	0	0	1	0	1	0	1	0	0	1	Y_{20}	1	0
X_{21}	0	0	1	0	1	0	0	1	1	0	Y_{21}	0	1
X_{22}	1	0	0	0	1	0	0	1	0	1	Y_{22}	0	1
X_{23}	1	0	0	0	1	0	1	0	1	0	Y_{23}	1	0
X_{24}	1	0	0	0	0	1	0	1	1	0	Y_{24}	0	1
X_{25}	0	1	0	1	0	0	0	1	1	0	Y_{25}	0	1
X_{26}	0	0	1	0	1	0	1	0	0	1	Y_{26}	1	0
X_{27}	0	0	1	0	0	1	0	1	1	0	Y_{27}	0	1
X_{28}	0	1	0	1	0	0	1	0	1	0	Y_{28}	0	1

The ANN is trained on the encoding input attributes vectors, X_m, and the corresponding output classes' vectors, Y_m. The number of input nodes is determined by the Equation (1). $m_n = m_1 + m_2 + m_3 + m_4 = 10$. The number of output nodes is $K = 2$. Accordingly, ANN has six possible different structures: (I=10, J_1=J_2=4, K=2), K=2). The final architecture of ANN is generated by selecting the optimal one from the results after six different structures are trained on the samples. The convergence rate between the actual and the desired output is achieved by: I=10, J_1=J_2=4, K=2, 0.53 learning coefficient, 0.64 momentum coefficient and 29,000 iterations. The allowable error is equal to 0.000001. Table 3 shows the first group of weights $(WZ1)_{i,j1}$ between each input node and the hidden nodes. The third group of weights $(WZ3)_{j2,k}$ between each hidden node and the output nodes is shown in Table 4. The results indicate the architecture of ANN is feasible.

TABLE III.
GROUP OF WEIGHTS $(WZ1)_{i,j1}$ BETWEEN INPUT AND HIDDEN NODES

Hidden nodes	Input nodes									
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
H_1	−4.09699	6.154562	−0.82675	−0.42227	4.128692	−2.73254	−4.93463	5.282225	3.060052	−3.63009
H_2	3.74124	−4.56639	1.11498	0.29616	−3.07741	2.59521	4.00533	−4.36782	−3.11607	2.28422
H_3	−1.2106	0.34984	0.15332	−0.19704	−0.15498	−0.56767	−1.17037	0.235355	1.106763	−1.36338
H_4	−1.42853	1.10953	−0.47917	−0.55404	0.651919	−0.32539	−0.89697	0.616702	0.56799	−1.02158

TABLE IV.
GROUP OF WEIGHTS $(WZ3)_{jk}$ BETWEEN OUTPUT AND HIDDEN NODES

Output nodes	Hidden nodes				Output nodes	Hidden nodes			
	H_1	H_2	H_3	H_4		H_1	H_2	H_3	H_4
ϕ_1	−9.20896	9.012731	−1.2113	−0.90564	ϕ_2	9.22879	−9.00487	0.773881	1.21892

Applying the GA to solve the equation φ_k (9) in order to get the i/p attributes vector which maximizes that function. The GA has population of 10 individuals evolving during 1500 generations. The crossover and the mutation were 0.26 and 0.01 respectively. The output chromosomes of buying computer and not buying computer target classes are sorted descendingly according to their fitness values. The threshold levels of the two target classes are 0.99996 and 0.999849, respectively. Therefore, both the local and global maximum of output chromosomes has been determined and will be translated into rules. Table 5 and Table 6 present the best set of rules belonging to not buying computer and buying computer, respectively.

TABLE V.
THE RULES EXTRACTION FOR CLASS NO (φ_1 IS MAXIMUM)

Rule no	Fitness	X_i vector from GA										Rules refinement (don't buying computer)
		x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	
1	0.99998	1	0	0	1	1	1	1	0	0	1	If age="＜30" AND student=no THEN Buys-computer=no
2	0.999874	1	0	1	0	0	1	1	0	0	1	If age="＞40" AND Credit-rating = Fair THEN Buys-computer=no

TABLE VI.
THE RULES EXTRACTION FOR CLASS YES (φ_2 IS MAXIMUM)

Rule no	Fitness	X_i vector from GA										Rules refinement (buying computer)
		x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	
1	0.999978	0	1	1	0	0	0	0	0	1	0	If Age="＜30" AND student=yes THEN Buys-computer=yes
2	0.999972	0	1	0	0	0	0	0	0	0	0	If Age="30~40" THEN Buys-computer=yes
3	0.999969	1	1	0	0	0	0	0	1	0	0	If Age="＞40" AND Credit-rating =excellent THEN Buys-computer=yes

TABLE VII.
CLASSIFICATION OF E-SHOPPERS

Age	Student	Credit-rating	Buying-computer
Age="＜30"	No		No
Age="＞40"		Fair	No
Age="＜30"	Yes		Yes
Age="30~40"			Yes
Age="＞40"		Excellent	Yes

The extracted rules show this method proposed in this paper is verified. The method produces accurate rules, and also has the less computational complexity. Therefore, it is acceptable to extract e-shopper's purchase rules via dynamic architecture neural networks and genetic algorithm. It is successful in classifying e-shoppers.

From the presented example of extracting rules above, conclusion is made as follows. Firstly, the best chromosome is divided into N segments. Each segment

represents one attribute, *An* ($n = 1, 2, \ldots N$), and has a corresponding bits length m_n which represents their values. Then, the attribute values are existed if the corresponding bits in the best chromosome equal one and vice versa. The operators "OR" and "AND" are used to correlate the existing values of the same attribute and the different attributes respectively. Finally, Rule refinement is made and redundant attributes are canceled after getting the set of rules.

IV. CONCLUSION

An attempt has been made in this study to classify e-shoppers to find out their purchase characteristics by adopting neural networks and genetic algorithm in machine learning for data mining. The obtained result indicates that the method proposed in this paper is successful in classifying e-shoppers.

An optimal algorithm of modeling architecture for ANN and a novel machine-learning algorithm for extracting rules from databases by means of genetic genetic algorithm have been presented in this paper. In this model, the number of hidden layers and hidden nodes are sequentially and dynamically generated until a level of performance accuracy is reached. Comparing with classical architecture of ANN, the dynamical architecture of ANN presents optimal performance accuracy. In addition, this paper presents a new genetic algorithm that does not need the computational complexity as deterministic finite state automata algorithm. It takes all input attributes into consideration, so it produces accurate rules. But other algorithms such as DNF use only the input attributes up to certain level. Also, it uses only part of weights to extract rules belongs to certain class. So the proposed genetic algorithm has a less computational time compared with another algorithm and does not make any approximation to the activation function.

Finally, the good result is achieved by applying the new methods to a given database for customers buying computer, but there is still space for improvement. In this paper, the algorithm is verified in the only database which includes twenty-eight samples. Therefore, in the future, research work should still consist of more experiments with other data sets, as well as more elaborated experiments to optimize the parameters of the proposed algorithms. The method will be further verified in different data sets. It is believed the better result will be obtained. It contributes to revealing e-shoppers' purchase characteristics.

ACKNOWLEDGMENT

This research was funded by National Natural Science Foundation of China, and the contract numbers is 70801022, and thank National Natural Science Foundation Council of China. The authors heartily thank Professor Qiang Ye to give many advices. In addition, we sincerely thank Associate professor Jian Liu and Wei Zhao, respectively. Under their help, we modify this paper again.

REFERENCES

[1] China Internet Network Information Center, China Internet the 26th Report [M], pp.4-5, July 2010.
[2] A. Bhatnagar, S. Ghose, A latent class segmentation analysis of e-shoppers, Journal of Business Research, issue 57, pp.758– 767, 2004.
[3] Y. B. Cho, Y. H. Cho, S. H. Kim, Mining changes in customer buying behavior for collaborative recommendations, Expert Systems with Applications, issue 28, pp.359–369, 2005.
[4] P. Giudici, G. Passerone, Data mining of association structures to model consumer behavior, Computational Statistics and Data Analysis, issue 38, pp.533–541, 2002.
[5] M. Chen, S. Han, Data mining: an overview from a database perspective, IEEE Transactions on Knowledge and Data Engineering, vol. 8(6), pp.866–883, 1996.
[6] A. Dhond, A. Gupta, and S. Vadhavkar, Data mining techniques for optimizing inventories for electronic commerce. Proceeding of the ACM-SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 480-486, 2000.
[7] J. Schafer, B. Konstan, E-commerce recommendation application, Journal of Data Mining and Knowledge Discovery, vol. 6, pp.125–153, 2001.
[8] H. S. Song, J. K. Kim, and S. H. Kim, Mining the change of customer behavior in an Internet shopping mall, Expert System with Applications, vol. 21(3), pp.157–168, 2001.
[9] G. Adomavicius, A. Tuzbilin, Using data mining methods to build customer profiles, IEEE Computer. vol.34 (2), pp. 74–82, 2006.
[10] R. Agrawal, R. Srikant, Fast algorithms for mining association rules[C]. Proceedings of the International Conference on Very Large Data Bases, pp. 407-419, 1995.
[11] V. Mitchell, F. Davies, L. Moutinho et al, Using Neural Networks to Understand Service Risk in the Holiday Product, Journal of Business Research, issue 46, pp.167–180, 1999.
[12] K. Chakraborty, K. Mehrotra, C. K. Mohan, Forecasting the behavior of multivariate time series using neural networks, Neural Networks, vol. 5(6), pp.961–970, 1992.
[13] H. Tsukimoto, Extracting rules from trained neural networks, IEEE Trans. Neural Networks, vol. 11 (2), pp.377–389, 2000.
[14] R. Battiti. First and second order methods for learning: between steepest decent and Newton's method. Neural Comput, vol. 4 (2) , pp.141–166, 1992.
[15] R.A. Jacob, Increased rates of convergence through learning rate adaptation, Neural Networks, vol.1(4), pp.295–308, 1988.
[16] X.H. Yu, G.A. Chen, S.X. Cheng. Dynamic learning rate optimization of the back-propagation algorithm, IEEE Trans. Neural Networks, vol. 6 (3), pp.669–677, 1993.
[17] D.L. Chester, Why two hidden layers are better than one? The Proceedings of the International Joint Conference on Neural Networks, pp.1265–1268, 1990.
[18] X. Zhang, Time series analysis and prediction by neural networks, Optim. Meth. Software, vol.4, pp.151–170, 1994.
[19] S. Kang, An investigation of the use of feed-forward neural networks for forecasting, Ph.D. Thesis, Kent State University, pp.30–35, 1991.
[20] M. Ghiassia, H. Saidane, A dynamic architecture for artificial neural networks, Neurocomputing, vol.63, pp.397–413, 2005.
[21] A. E. Elalfi, R. Haque, M. E. Elalami, Extracting rules from trained neural network for managing e-business,

Applied Soft Computing, issue 4, pp.65–77, 2004.
[22] R. Setiono, Extracting M-of-N rules from trained neural networks, IEEE Trans. Neural Networks, vol.11(2), pp.512–519, 2000.
[23] F. Wotawa, G. Wotawa, Deriving qualitative rules from neural networks—a case study for ozone forecasting AI communications, vol. 14, pp.23-33, 2001.
[24] R. Setiono, J.M. Zurada, Extraction of rules from artificial neural networks for nonlinear regression, IEEE Trans. Neural Networks, vol. 13 (3), pp.564–57, 2002.