Annals of Operations Research 75(1997)291–301

Neural network models: Foundations and applications to an audit decision problem

Rebecca C. Wu
Department of Accounting, College of Management, National Taiwan University, Taipei, Taiwan 106, Republic of China
E-mail address: rwu@haydn.ntu.edu.tw

We investigate the possibility of applying artificial intelligence to solve an audit decision problem faced by the public sector (namely, the tax auditor of the Internal Revenue Services) when targeting firms for further investigation. We propose that the neural network will overcome problems faced by a direct knowledge acquisition method in building an expert system to preserve the expertise of senior auditors of the IRS in Taiwan. An explanation of the neural network theory is provided with regard to multi- and single-layered neural networks. Statistics reveal that the neural network performs favorably, and that three-layer networks perform better than two-layer neural networks. The results strongly suggest that neural networks can be used to identify firms requiring further auditing investigation, and also suggest future implications for intelligent auditing machines.

1. What is neurocomputing?

During the past decade, expert systems have become the first practical application of artificial intelligence. However, many expert system applications are rule-based systems, which require a time consuming and difficult knowledge acquisition process to extract expertise from domain experts [1–3].

Neurocomputing is the application of artificial neural networks to practical problems. An artificial neural network consists of processing elements (which is analogous to neurons in the biological neural system) in an interconnected network [4–11]. A processing element accepts inputs, processing elements. A neural network application is built by using real-world problem-solving examples in a particular application area as training cases. Each training case consists of input data and the associated decision output. For example, in a bank loan domain, an applicant's personal data and macro-economic data are modeled as input data, and the loan approval decision (yes/no) is modeled as the output value. The purpose of the network building process is to minimize the magnitude of errors between the actual output and the expected output value. With proper design and adequate training examples, an operational artificial neural network system can generate accurate output, and will be useful to solve practical problems. Recently, neurocomputing has been applied to a wide variety of real-world problems such as finance, engineering, real estate, and other business areas [12–14].

2. An audit decision problem

The auditor in this article refers to the auditor in the public sector, namely the tax auditor of the Internal Revenue Services. The senior tax auditor, based on the business income tax return filed by the profit-seeking firm, which includes the business income statement, balance sheet, and the supporting schedules and forms, performs the paper review and determines the likelihood of tax evasion and the necessity for further investigations. The further investigation will include the auditing of related account books, documentary evidence, bank account, cash flow and party fund transfers. This takes a lot of time and effort. Therefore, the expert auditor's effective identification of target firms for further investigation through a paper review process results in a high payoff.

Currently there are few senior auditors who can quickly identify firms worthy of further investigation, which will result in large amounts of tax levied. Such expertise is very scarce. The IRS in Taiwan is considering preserving this expertise through building an expert system. However, a direct knowledge acquisition method proved to be unsuccessful because the expert auditors were too busy to participate in the knowledge acquisition meetings. The artificial neural network is proposed to overcome the problem. There are many business income tax filing cases every year. The contents of those cases include firm attributes and accounting data, which serve as the network's input values, and the senior auditor's review decision, which serves as the network's output value.

3. Neural network theory

A neural network model consists of a number of processing units interconnected in a network. A processing unit (PE) receives input signals with weights from other processing units, aggregates these signals based on an input function, and generates an output signal based on an output transfer function. The output signal is then routed to other processing units as directed by the topology of the network. Neural networks adjust input weights to improve performance [15]. This capability to adapt, or learn, is essential for many types of intelligent activity such as decision making, combinatorial problem solving, and so forth.

3.1. Single-layer neural network

Figure 1 captures the basic elements of a typical artificial neural net. Activity is initiated by a vector of inputs labeled $x_1, x_2, \ldots, x_n$. Each input is multiplied by an associated

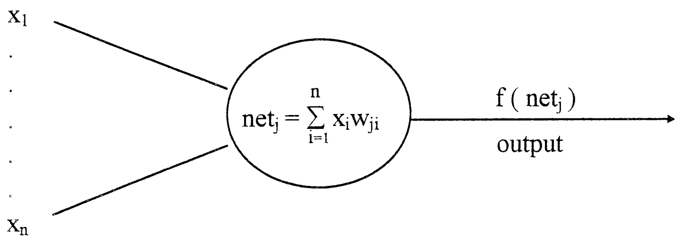

Figure 1. Processing element of a single-layer artificial net.

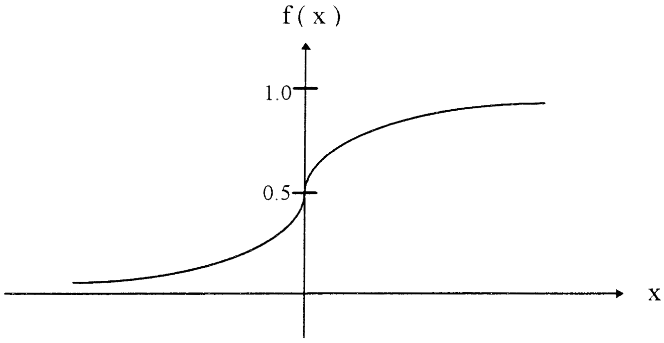

Figure 2. Typical sigmoid transfer function.

weight $w_1, w_2, \ldots, w_n$. The resulting products are summed at the output node, which is the network value of node j:

$$net_j = \sum x_i w_{ji}, \tag{1}$$

$$F(net_j) = 1/(1 + e_j^{-net}) = o_j = \text{actual output of node } j. \tag{2}$$

The sigmoid function, $1/1 + e_j^{-net_j}$, is used as a transfer function because the transferred value remains within limits of one regardless of the network values of the summed inputs; e is the mathematical exponential constant. Figure 2 presents the sigmoid function.

3.2. Multi-layered neural network

An example of a three-layer neural net is presented in figure 3. In addition to input layer and output layer, there exists a hidden layer of nodes. Hidden layers can slow the neural net's computation time, but for many problems there is a significant offsetting advantage. The multi-layer net can resolve the classification problem where the decision region is complex and nonlinear.

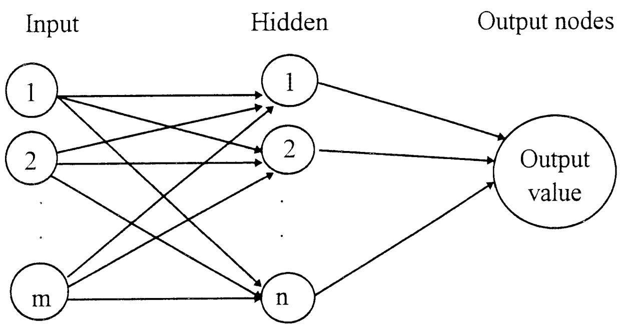

Figure 3. Three-layer neural net.

3.3. Neural network learning paradigm

A fundamental problem in mapping a classification task to a neural network model is to derive network connection weights. A possible solution is to let the network learn the task by training it with examples. In this article, the application is in the auditing domain. Since the application requires supervised training, and has continuous and discrete input data, the back propagation paradigm is used in the study. It has the ability to handle a large number of nodes, and software tools based on back propagation paradigms are readily available. Back propagation consists of two phases: forward propagation and backward propagation.

3.2.1. Forward propagation

In forward propagation, x_i is fed into the input units. Messages are then propagated up the network according to the current weight matrix W, and an output o_i is generated by the output units. The output value o is then compared with the desired output value d, at each output unit. It then determines the direction and degree of adjustments to individual connection weights. It performs a gradient descent, namely,

$$E_j = (d_{ij} - o_{ij}) * (f'(net)). \qquad (3)$$

In words, the product of the error, $d_{ij} - o_{ij}$ times the gradient of descent $f'(net)$ is taken. Recall that

$$F(net_j) = 1/(1 + e_j^{-net}) = \text{actual output of node } j = o_j,$$

which is the sigmoid function. While other functions could be used, the sigmoid function limits the values of $f'(net)$ to that less than 1; it is non-decreasing, and it is everywhere differentiable. Moreover, the derivatives of the sigmoid function result in the function

$$F_j'(net) = o_j(1 - o_j). \qquad (4)$$

3.3.2. Backward propagation

The back propagation algorithm earns its name because of its mechanism to propagate the error information back, layer by layer, from the output units to input units. It uses a recursive algorithm starting at the output nodes and working back to the first hidden layer. It adjusts weights by

$$W_{ij}(t+1) = n(e_j - o_i), \tag{5}$$

where

w_{ij} : the weight from hidden node i, or from an input, to node j at time $t+1$,
o_i : either the output of node i, or an input for the first layer,
n : a learning constant, and
e_j : the error term for node j.

If node j is an output node, then

$$E_j = (d_j - o_j)[o_j(1 - o_j)]. \tag{6}$$

Recall that d is the desired output value, o is the actual output value, and $o(1-o)$ is $f'(net)$, where $f(net)$ is the sigmoid function.

If node j is an output node, then

$$E_j = \sum_k E_j w_{kj}(o_j(1 - o_j)), \tag{7}$$

where k ranges over all nodes in the layers above node j. The process will repeat for each iteration of computation.

4. Designing the neural network application in auditing

Recently, a number of researchers [16–19] have proposed using machine learning techniques for modeling expert knowledge. Neural networks represent an approach to this issue. Unlike traditional expert systems, where knowledge is made explicit in the form of rules, neural networks generate their own rules by training examples. Learning is achieved by a learning rule that adapts or changes the weights of the network in response to example inputs and intermediate outputs.

The neural network model is applied in classifying tax cases based on some signals. The system's output is the audit decision (no further audit required or further audit required). The cases selected for auditing are carefully decided. The sample examples are gathered from the IRS expert level auditors' audit case file. The resulting testing sample consists of 180 cases, 90 of which require a further audit and 90 of which require no further audit. This auditors' audit case reports are useful in predicting the audit decisions. They consist of information about a firm's business income tax behavior. Input data attributes identified by those expert-level auditors in each case are collected for the diagnostic purpose. The input data attributes are:

- Sales in appearance relative to sales reported.
- Sales expense reported relative to sales revenue reported.
- Purchase frequency relative to sales reported.
- Ending inventory reported relative to sales reported.
- Borrowing from shareholders.
- Receivable from shareholders.
- Short-term borrowing.
- Line of business.
- Net income ratio relative to regulated standard ratio.
- Net income ratio relative to industry average ratio.
- Net income ratio relative to priors year ratio.
- Gross profit ratio relative to regulated standard ratio.
- Gross profit ratio relative to industry average ratio.
- Gross profit ratio relative to prior years ratio.
- Unit price relative to unit cost.
- Custom duties reported as material cost relative to customs duties refund reported as revenue.

The input value for each attribute is discrete value 0 and 1; 0 represents an abnormal situation, 1 represents a normal situation. Half of the 180 cases were used as the training set, the other half of the cases were held as a testing set to validate the implemented network.

5. Implementing the audit neural network

The implementation environment was a PC, a 586 Pentium machine with a math co-processor. A commercial development package, Neural Work, was selected for prototyping. The package uses the back propagation learning paradigm, and uses the sigmoid function as the transfer function. It was reasonably efficient.

Two-layer and three-layer networks were tested. For three-layer networks, different numbers of hidden units were tried. Finally, two networks, one with no hidden layer (two-layer) and the other with one hidden layer (three-layer), were constructed for better performance reasons. All hidden units were fully connected in the three-layer network to the input units. The constructed three-layer neural network is presented in figure 4.

6. Performance evaluation

Table 1 shows the calculation of deviation between system-generated output value and the desired output value for 90 test cases. Since it would take an infinite

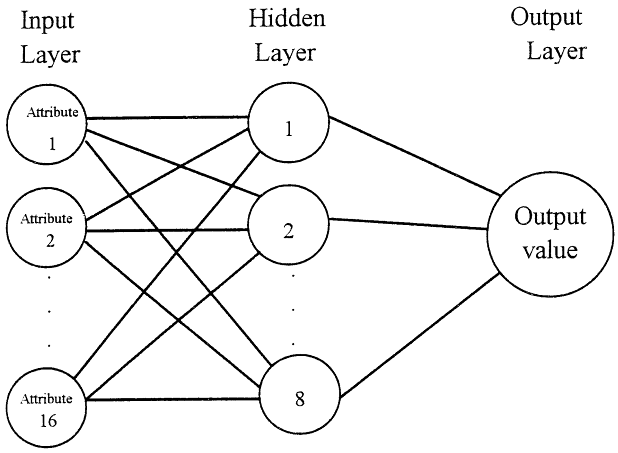

Figure 4. Three-layer audit neural network.

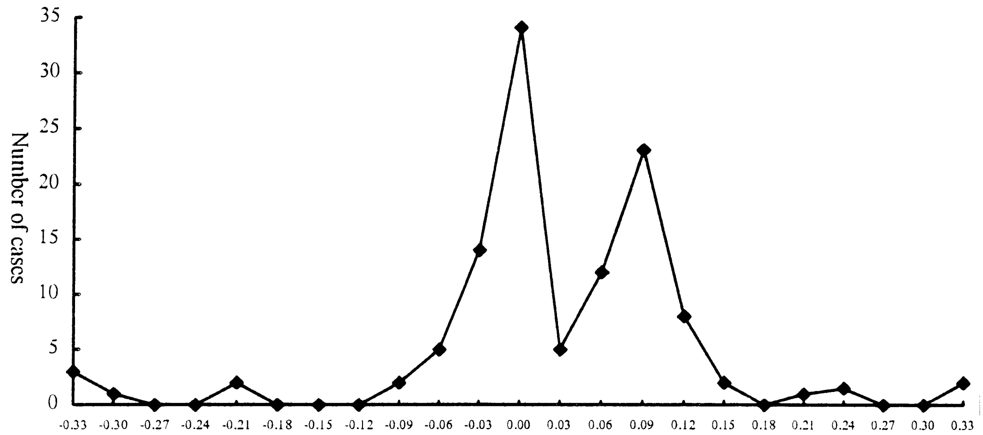

Figure 5. Three-layer neural net deviation plot.

amount of time for the output node to arrive at either a +1 or a 0 result; therefore any activation greater than +0.9 is considered a +1 on the final output layer, and an activation less than 0.1 will be considered a 0 output for the output layer [5]. Figure 5 plots the deviation. Examining the distributions of the estimated decision output also confirms that the neural network has the desirable properties of an unbiased bell-shaped curve (figure 5).

A performance measure, predictive accuracy, is used to evaluate relative algorithm performance. The measure has been used by other researchers in comparing various induction algorithms. Predictive accuracy is defined here as the number of test examples correctly classified. For example, if 80 out of 90 testing cases are classified correctly, the predictive accuracy is 89%.

Table 1

Three-layer audit neural net deviation.

(1) Actual output value	(2) Neural network output value	(1)−(2) Deviation	(1) Actual output value	(2) Neural network output value	(1)−(2) Deviation
1	0.997	0.003	0	0.009	– 0.009
1	0.903	0.097	0	0.031	– 0.031
1	0.904	0.096	0	0.029	– 0.029
1	0.900	0.100	0	0.051	– 0.051
1	0.922	0.078	0	0.011	– 0.011
1	0.918	0.082	0	0.041	– 0.041
1	0.893	0.107	0	0.013	– 0.013
1	0.910	0.090	0	0.010	– 0.010
1	0.924	0.076	0	0.020	– 0.020
1	0.925	0.075	0	0.006	– 0.006
1	0.923	0.077	0	0.310	– 0.310
1	0.910	0.090	0	0.008	– 0.008
1	0.922	0.078	0	0.001	– 0.001
1	0.978	0.022	0	0.014	– 0.014
1	0.954	0.045	0	0.007	– 0.007
1	0.920	0.080	0	0.005	– 0.005
1	0.967	0.033	0	0.060	– 0.060
1	0.901	0.099	0	0.011	– 0.011
1	0.947	0.053	0	0.003	– 0.003
1	0.907	0.093	0	0.020	– 0.020
1	0.934	0.066	0	0.011	– 0.011
1	0.967	0.033	0	0.013	– 0.013
1	0.927	0.073	0	0.208	– 0.208
1	0.870	0.130	0	0.011	– 0.011
1	0.967	0.033	0	0.065	– 0.065
1	0.921	0.079	0	0.003	– 0.003
1	0.903	0.097	0	0.019	– 0.019
1	0.921	0.079	0	0.057	– 0.057
1	0.971	0.029	0	0.038	– 0.038
1	0.989	0.011	0	0.023	– 0.023
1	0.950	0.050	0	0.039	– 0.039
1	0.699	0.301	0	0.031	– 0.031
1	0.942	0.058	0	0.025	– 0.025
1	0.978	0.022	0	0.002	– 0.002
1	0.917	0.083	0	0.055	– 0.055
1	0.933	0.067	0	0.033	– 0.033
1	0.970	0.030	0	0.010	– 0.010
1	0.927	0.073	0	0.018	– 0.018
1	0.966	0.034	0	0.018	– 0.018
1	0.925	0.075	0	0.012	– 0.012
1	0.899	0.101	0	0.011	– 0.011
1	0.776	0.224	0	0.010	– 0.010
1	0.900	0.100	0	0.031	– 0.031
1	0.961	0.039	0	0.001	– 0.001
1	0.921	0.079	0	0.027	– 0.027

Table 2

Relative predictive accuracy (%).

Two-layer neural net	Three-layer neural net	ID3
94.0	95.0	90.0

The validation result on the sample is shown in table 2. The three-layer neural networks perform better than two-layer neural networks. ID3 is also used to compare the predictive accuracy. Comparing the statistics in table 2, the neural network approach appears to have performed favorably.

The effectiveness of an algorithm (e.g. back propagation) in identifying structural characteristics of a set of training data (i.e. the examples used to develop the neural network model) can function as a useful relationship between attribute values and decisions on classification. If the training set data is a good representation of the problem domain and that domain exhibits structural similarities between attributes and decisions on classification, an effective algorithm will be able to identify that structure.

7. Discussion and conclusion

Since the predictive accuracy for the audit neural network is 94% (two-layer) and 95% (three-layer), there appears to be a high level of correspondence between the actual classification and the resulting network classification. The 16 attributes that describe the tax evasion signals seem to be quite effective in predicting.

Two further researches are needed. First, how does one know how large a training set is sufficient? In general, this question is in need of research, although recent work [20] has resulted in a theory of lower bounds on the number of examples required for learning. Quinlan [19] has done some preliminary work on estimating the necessary size of the training set. Still, currently there is no established theory that applies. Second, there is a need for better tools for estimating convergence time. Some neural networks, back propagation included, cannot be guaranteed to converge. Thus, the user may give up after a large number of trials, not knowing that the algorithm would have converged after a few more trials.

Many people have raised the question: “Will neural networks replace expert systems?” Neural networks used as pattern classification tasks are very effective. However, neural networks cannot do many of the things that expert systems do well, for example, in the construction problem domains.

A hybrid expert system, integrating a neural network model and an expert system, has been suggested for future research. The framework is illustrated in figure 6.

Based on the previous experts’ audit cases, the key attributes that can identify the tax evasion cases serve as input to neural network model and then the output of the

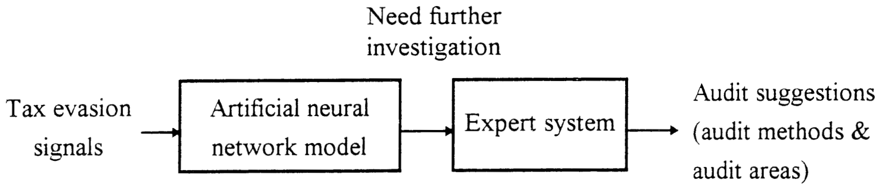

Figure 6. Integrated audit expert system and audit neural network.

neural net, namely, decision on further investigation, is fed to expert system and the audit suggestion for auditors becomes the expert system output. Future research in the development and application of artificial neural networks may ultimately increase our understanding of fundamental functions of human intelligence and perhaps lead to truly intelligent auditing machines.

References

[1] A. Barr and E.A. Feigenbaum, *The Handbook of Artificial Intelligence*, Vol. 2, Pittman, Boston, MA, 1983.

[2] R.O. Duda and E.H. Shortliffe, Expert systems research, Science (April 1983) 261–268.

[3] R.S. Michalski and R.L. Chilausky, Learning by being told and learning from examples: An experimental comparison of the two methods of knowledge acquisition in the context of developing an expert system for soybean diagnosis, International Journal of Policy Analysis and Information Systems (1980)125–61.

[4] M. Caudill, Neural networks primer, part 1, AI Expert (December 1987)46–52.

[5] M. Caudill, Neural networks primer, part 2, AI Expert (February 1988)55 61.

[6] M. Caudill, Neural networks primer, part 3, AI Expert (June 1988)59–61.

[7] M. Caudill, Neural networks primer, part 4, AI Expert (August 1988)61–67.

[8] M. Caudill, Neural networks primer, part 5, AI Expert (November 1988)57–65.

[9] M. Caudill, Neural networks primer, part 6, AI Expert (November 1988)67–65.

[10] M. Caudill, Neural networks primer, part 7, AI Expert (May 1989)51–58.

[11] M. Caudill, Neural networks primer, part 8, AI Expert (August 1989)61–67.

[12] M. Caudill and C. Butler, *Naturally Intelligent Systems*, MIT Press, Cambridge, MA, 1990.

[13] P.M. Noaker, Smart sensing on the factory floor, Production 100 (December 1988)42–48.

[14] W. VerDuin, Solving manufacturing problems with neural nets, Automation (July 1990)54–58.

[15] R.P. Lippman, An introduction to computing with neural nets, IEEE ASSP Magazine 3(4)(1987) 4–22.

[16] J.V. Hansen, G.J. Koehleer and W.F. Messier, A comparative performance of the predictive performance of the predicting ability of inductive learning algorithms and logistic regression, Working Paper, University of Florida, 1991.

[17] J.V. Hansen and W.F. Messier, Inducing rules for expert system development: An example using default and bankruptcy data, Management Science 34(1988)1403–1415.

[18] R.S. Michalski, A theory and methodology of inductive learning, in: *Machine Learning: An Artificial Intelligence Approach*, R.S. Michalski, J.G. Carbonell and T.M. Mitchel, eds., Tioga, Palo Alto, CA, 1983.

[19] J.R. Quinlan, Learning efficient classification procedures and their application to chess end games in: *Machine Learning: An Artificial Intelligence Approach*, R.S. Michalski, J.G. Carbonell and T.M. Mitchel, eds., Tioga, Palo Alto, CA, 1983.

[20] A. Elrenfeucht, D. Haussler, M. Kearns and L. Valiant, A general lower bound on the number of examples needed for learning, *COLT88, Proceedings of the Conference on Learning Theory*, Morgan Kaufmann, Los Altos, CA, 1988, pp. 110–20.

[21] Y. Yoon, G. Swales and T. Margavio, A comparison of discriminant analysis versus artificial neural networks, Journal of the Operational Research Society 44(1993)pp. 51–60.

[22] D. Triguerios and R.T. Taffler, Neural networks and empirical research in accounting, Working Paper, The City University Business School, London, 1995.

[23] R. Sharda and R. Wilson, Performance comparison issues, in: *Neural Network Experiments for Classification Problems*, E. Nunamaker and R. Sprague, eds., IEEE Computer Society Press, 1993, pp. 649–657.