AGENT BASED MODELLING FOR DATA MINING TO DETECT TRENDS IN DATASETS

SARATHA SATHASIVAM, MURALY VELAVAN

Abstract: The application of neural networks in the data mining has become wider. Although neural networks may have complex structure, long training time, and uneasily understandable representation of results, neural networks have high acceptance ability for noisy data and high accuracy and are preferable in data mining. In this paper, we describe how Hopfield network is able to induce logical rules from large database by using reverse analysis method: given the values of the connections of a network, we can hope to know what logical rules are entrenched in the database. In this paper, an agent based modeling (ABM) in reverse analysis will be developed by using Netlogo. We also carried out some computer simulation to verify and test the ABM developed.

Keywords: Data mining, neural networks, Hopfield network, reverse analysis

Introduction: Data mining is not merely automatic collecting of knowledge. The aim is to extract novel, plausible, relevant and interesting knowledge from the database. We do not provide an introduction to data mining techniques in this paper, but instead refer the interested reader to one of the good book in the field Neural network is a parallel processing network which generated with simulating the image intuitive thinking of human, on the basis of the research of biological neural network, according to the features of biological neurons and neural network and by simplifying, summarizing and refining. Initially, the application of the neural network in data mining was not optimistic, and the main reasons are that the neural network has the defects of complex structure, poor interpretability and long training time.

But its advantages such as high affordability to the noise data and low error rate, the continuously advancing and optimization of various network training algorithms, especially the continuously advancing and improvement of various network pruning algorithms and rules extracting algorithm, make the application of the neural network in the data mining increasingly favored by the over helming majority of users.

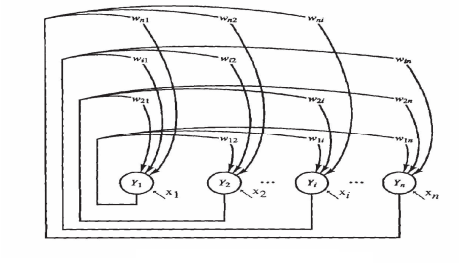


Figure 1: Discrete Hopfield Model

Fig. 1 shows a Hopfield network of *N* neurons. The input vector is $X = [x_1, x2, .x_n]$, and the state of the network is given by the output vector $Y = [y_1, y2, .y_n]$, where y_i denotes the output of neuron *i*, which can only be +1. Hopfield network is uniquely defined by (**W**, **b**), there into, $W = [w_{ij}]$ is a *N* x *N*-dimensional zero –diagonal matrix, and $[w_{ij}]$ is the weight connecting neuron *i* and *j*. *B* is a *N* –dimensional vector, where b_i denotes the fixed threshold value of each neuron *i*. and the relation of them is shown as followings:

In section 2, we will focus on Hopfield neural network. Next, we will look into how logic learning been carried out in Hopfield network. In this section also, we will explained on how rules can be extracted from a data base. Following that, in section 4, algorithm of reverse analysis algorithm is discussed. Later we will discuss about agent based modeling. Finally, conclusion and discussion occupied the last section.

Hopfield Network: The discrete Hopfield network (DHNN) [3, 4, 5] is used as associative memory, in which stored data is recalled by association with input data, rather than by an address.

$$\begin{cases} x_j(t) = \sum_{j=1}^{N} w_{ij} y_j(s) + B_j \\ y_j(t+1) = \mathrm{sgn}(x_j(t)) \end{cases} i, j = 1,2,....N \quad (1)$$

In (1), the output of neuron *i* am given by *sgn (.),* which is a symmetric signum function, whose output is +1 or -1. if the argument of the signum function is zero, then the output of neuron *i* remains unchanged. When simple asynchronous updating is used for the Hopfield network outputs, an energy function E (Lyapunov function) is given by:

$$E = -\frac{1}{2}\sum_{\substack{i=1 \\ i\neq j}}^{N}\sum_{\substack{j=1 \\ j\neq i}}^{N} W_{ij} x_i(s) x_j(s) - \sum_{i=1}^{N} B_i y_i(s) \quad (2)$$

As the network evolves according to the dynamics in (2), the energy E can only decrease or stay unchanged at each update. Eventually the network will converge to a (local) minimum energy state because E is

bounded from below.

Reverse Analysis Method: In this paper, we define a machine learning method: Reverse Analysis Method [6, 7, 8], that uses Hopfield neural network to discover trends in datasets. They are few methods of rules extraction strategy such as subset method [9], M of N method [10], RULEX method [11] and few other methods. However by using Reverse Analysis method we can learn the inherent relationships among the data. Furthermore, this method is additive and need less computation effort as illustrated in previous section.

This method consists of these following steps:

i) Initialize number of trials, energy relaxation loops, number of patterns.

ii) Extract the events from the database and represent in binary/bipolar pattern, where 0 indicates false state and 1 indicates true state (for bipolar -1 represent false state and 1 represent true state).

iii) Calculate the connection strengths for the events using Hebbian learning.

iv) List out all the connection strengths obtained for third order connections, second order and first order connections

v) Capture all the nonzero values (connection strengths) for third order connection.

vi) By using the method in section 3, list out all the corresponding clauses for

vii) Calculate the connection strengths for the extracted clauses in step (vii) and deduct the value of the corresponding clauses connection strengths from (v).

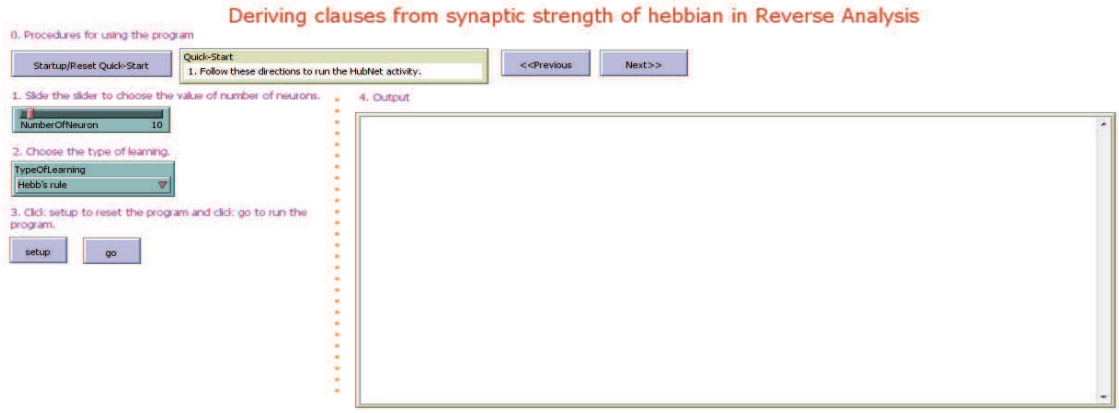
viii) Repeat the similar steps to extract the clauses corresponding to the first order and second order connections.

Agent Based Modeling: Netlogo is an agent-based model programming environment built on the programming language JAVA. It was first authored by Uri Wilensky [12]. It was designed to stimulate natural and social phenomena over time using a group of autonomous agents. It is a well suited method for modeling complex systems that can give instructions to hundreds or thousands of "agents" to operate independently. Moreover, the model can be viewed in either 2D or 3D form.

Firstly, a simulator of Hopfield networks that using a conventional computer had created instead of every time build up a new network design or store a new set of memories. It saves lots of energies and times for the programmer to rebuild new system from time to time. Thus, an agent based modelling had designed for the user to run the simulator. In this paper, an agent based modeling which was implemented the reverse analysis had created. The screenshot below shows the developed agent based modeling.

Figure 2: The layout of deriving clauses from synaptic strength of hebbian in reverse analysis.

Below is the explaination on the development of the agent based model.



PHASE 1: Entering Values

1) Press the start up / Reset Quick-Start button for the new user.

2) Later, key in Number of neuron (NN). The maximum of NN is 100.

3) Choose type of learning which is either Hebb's rule or Wan Abdullah's method [3] .

4) After all the value had been set, press the setup button to fix and set those values in the program.

5) Then, press go button to run the program.

PHASE 2: Training

6) Enumerate number of neurons and pattern in the system and initialize the value that had key in by user in the clauses

Next, the system will extract out the events that users had entered and named them according to 0 and 1. From here, the connection strengths for the events will be calculated using Hebbian learning. After that,

system will list out all the connection strengths obtained from third order connections to first order connections. Later by capture the nonzero in level 3 then sorting the neurons by declare them in third dimensions array and based on the condition that declared, if condition fulfil, the clause will assume pass (neurons formed will declare as a clause) and undergo deduct value process. However, if fail, if will go to next level to check the condition had fulfil or not. If until level 1 it still fails to fulfil, the clause will not generate. It will start again by running another way of neurons arrangement. After all the clauses in every level had formed, the testing is come to its end. Lastly, the system will print out the output for each run.

Simulation and discussion: Reverse analysis method has been tested in a small simulated data set as shown in Table 1. The logical rules extracted from the method seem same with the frequent observations.

Table 1: Customers daily purchased from a supermarket:

	Burger Loaf	Peanut Butter	Jam	Sausage	Mini Bun
Sarah	√				
Mikey		√	√	√	
Johnny	√				√
Lenau		√			√
Susani	√		√	√	

By using Reverse Analysis method approached we was discussed previously, we obtained the following rules:

Burger loaf ← Peanut Butter, Jam

Mini Bun ← Sausage, Jam

From the rules, we can assume that a customer who purchased peanut butter and jam has a high probability of purchasing burger loaf also. On the other hand, a customer who purchased sausage and jam also has a high probability of purchasing mini bun.

The logical rules that were extracted by using Reverse Analysis method can help the store in check their stock according to the customers need. Significant patterns or purchasing trends in the data set have been identified by using reverse analysis method. The store can use the extracted patterns to improve its sales process according to customers shopping habits. Furthermore, the knowledge obtained may suggest new initiatives and provide information that improves future decision making in monitoring the sales

Conclusion: In this paper we integrated Reverse Analysis and agent based modeling to us in revealing knowledge hidden in data and turn this knowledge into a crucial competitive advantage faster. By using this technique we able to unearth more hidden relationships in the data sets and get more patterns to describe certain data sets.

Acknowledgement: This research is partly financed by Fundamental Research Grant Scheme (203/ PMATHS/6711368) from the Ministry of Higher Education and University Sains Malaysia.

References:

1. Michael, B & Gordon, L. 2004. Data Mining Techniques. New York: John Wiley & Sons, Ltd.
2. Michalski, R. S., Bratko, I., & Kubat, M. 1998. Machine Learning and Data Mining: Methods and Applications. New York: John Wiley & Sons, Ltd.
3. Hopfield, J.J. 1982. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Proceedings. Natl. Acad. Sci. USA. , **79(8)**, pp 2554-2558.
4. Hopfield, J.J. 1984. Neurons with Graded Response Have Collective Computational Properties like Those of Two-State Neurons. Proceeding. Natl. Acad. Sci. USA. , **81(10)**, pp 3088-3092.
5. Hopfield, J.J. and Tank, D.W. 1985. Neural Computation of Decisions in Optimization Problems. Biol. Cyber, **52,** pp 141-152.
6. Wan Abdullah, W.A.T. 1993. The Logic of Neural Networks. Phys. Lett. A., **176**. pp 202-206.
7. Saratha Sathasivam. & Wan Abdullah, W.A.T. 2008. Logic Learning in the Hopfield Networks, Modern Applied Science, 2(3), pp 57-62.
8. S. Sathasivam. 2007. Logic Mining in Neural Network. PhD Thesis, Malaysia.
9. Bochereau, L & Bourgine, P. 1990. Extraction of semantic features and logical rules from a multilayer neural network. In International Joint Conference on Neural Networks, Washington, pp 579-582.
10. Towell, G. & Shavlik, J. 1993. The Extraction of Refined Rules From Knowledge Based Neural

Networks, Machine Learning, **131**, pp 77-101.
11. Andrews, R & Geva, S. 1994. Rule extraction from a constrained error back propagation MLP. In Proc 5th Australian Conference on Neural Networks, Brisbane Queensland, pp 9-12.
12. Levent Yilmaz,Tuncer I. Ören 2009. Agent-Directed Simulation and Systems Engineering, Wiley series in systems engineering and management, pp 280-283.

***

School of Mathematical Sciences, Universiti Sains Malaysia, USM, Penang,saratha@usm.my
School of General and Foundation Studies, AIMST University, Bedong,Kedah, Malaysia,dsmuraly@yahoo.com