A Hybrid Training Mechanism for Applying Neural Networks to Web-based Applications*

Ko-Kang Chu
Dept. of Info. and Comp. Eng.,
Chung-Yuan Christian Univ.
Chung-Li, 320, Taiwan
kirk@ms2.hinet.net

Maiga Chang
Program Office of National
Science and Technology Program
for eLearning in Taiwan
maiga@ms2.hinet.net

Yen-Teh Hsia
Dept. of Info. and Comp. Eng.,
Chung-Yuan Christian Univ.
Chung-Li, 320, Taiwan
hsia@ice.cycu.edu.tw

Abstract - *This paper proposes a hybrid training neural network and applying it to the Accuracy Counter (AC) developed previously. The neural network is used for detecting the cheating model for abnormal browsing behaviors performed by users in the conflicting environment. The most significant issue, training, should be taken into consideration while we are applying the neural network to web-based applications such like the Accuracy Counter. Therefore, we design a hybrid web-based training mechanism for neural networks to deal with this kind of training problem. Finally, we also find out that the AC's block rate for detecting the abnormal browsing behaviors is increasing from 61% (rule-based) to 76% (neural networks with hybrid training mechanism) in the efficient and acceptable training period.*

Keywords: Web counter, neural network, online training

1 Introduction

Neural Networks are usually using for detecting the possible malicious behaviors in network administration fields based on the well-trained intruder model.[2][3] The major characteristic of neural networks is trainable. Through the training stage before attaching the neural networks to the system application formally, the neural networks will be educated to identify the fed in specific patterns, which is so-called the training set.

The training mechanism is quite useful as we know, however, the trained neural network can not be used forever but a short-period.[4] Two reasons that cause the neural networks need to train continuously are similar patterns and new tricks.[5][6] Malicious users always use similar means with a minor differences to do the same job, moreover, new vulnerable holes will be discovered and used by users. Therefore, many researchers proposed mechanism to keep the neural networks own the ability to target which steps belong to the deleterious behavior.[7][8]

The simplest idea is to re-train the neural networks. Obviously, things will not be so easy since the neural networks will be unavailable while the training process takes place.[9] Some researchers developed some kinds of hybrid training mechanism by dividing the single one-way training-using process into an online/offline way, that is, the online using and offline training way.[10][11] This paper proposes a hybrid web-based training mechanism in order to enhance the learning and adaptive abilities of web applications.

Section 2 will describe the hybrid training mechanism and how to apply it to the web-based applications. The experiment, screen snapshot and results are discussed by Section 3, and Section 4 makes a simple conclusion and figures out what could be next.

2 Design of Neural Network Accuracy Counter

In general, a user will tend not to repeatedly access the same webpage within a short time interval. Therefore, when the same computer accesses the same webpage again in a very short time, it is feasible to consider the connection attempt as "unreasonable." The AC system is designed to block increments of the counter's value under such circumstances.[1]

Based on the architecture of AC, any web access should pass through these four stages' verification. Figure 1 demonstrates the architecture of the Accuracy Counter. The hybrid training mechanism for web-based application is designed as Figure 2 shown below. This mechanism will replace the original rule-based detection module and activate during the stage 4 illustrated in Figure 1.

* **0-7803-8566-7/04/$20.00 © 2004 IEEE.**

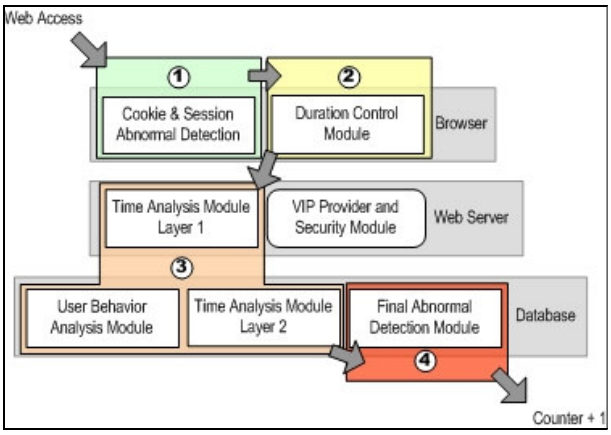

Figure 1. Three-tier architecture of AC

The operation processes of the hybrid neural network system could be explained according to Figure 2. There are three major processes, offline training, using and online training. **First**, the offline training process is working just like the old fashion neural networks. In the offline training stage we put several real and simulated dataset with expected results into the neural network for training. **After that**, the well-trained neural network can be used for analyzing the access behavior in order to judge whether the access is normal or abnormal. (According to process A1 to A8 in Figure 2)

However, although the neural network is well-trained, there are still some situations that may cause the neural network confuses such as similar patterns and new tricks. Under such condition the neural network will store those access records that it not 100% sure into the database. (The step A7 in Figure 2)

In order to apply the hybrid training mechanism to the web-based applications, a web-based supervised learning interface (webpage) is designed. The webpage is developed for filtering those records that the neural network can not tell definitively what kinds of access it is. Since that, there could be a human being to review these records periodically and categorize them as the new training set. (As process B1 to B3 in Figure 2)

Finally, the online training process could be engaged by the human supervisor. During the online training stage the neural network in the using stage will not be affect, because we use another exactly same neural network for training separately. Once the online training is complete, the new neural network will replace the old one. (According to process B4 to B8) Although the mechanism proposed here looks very simple, it works!

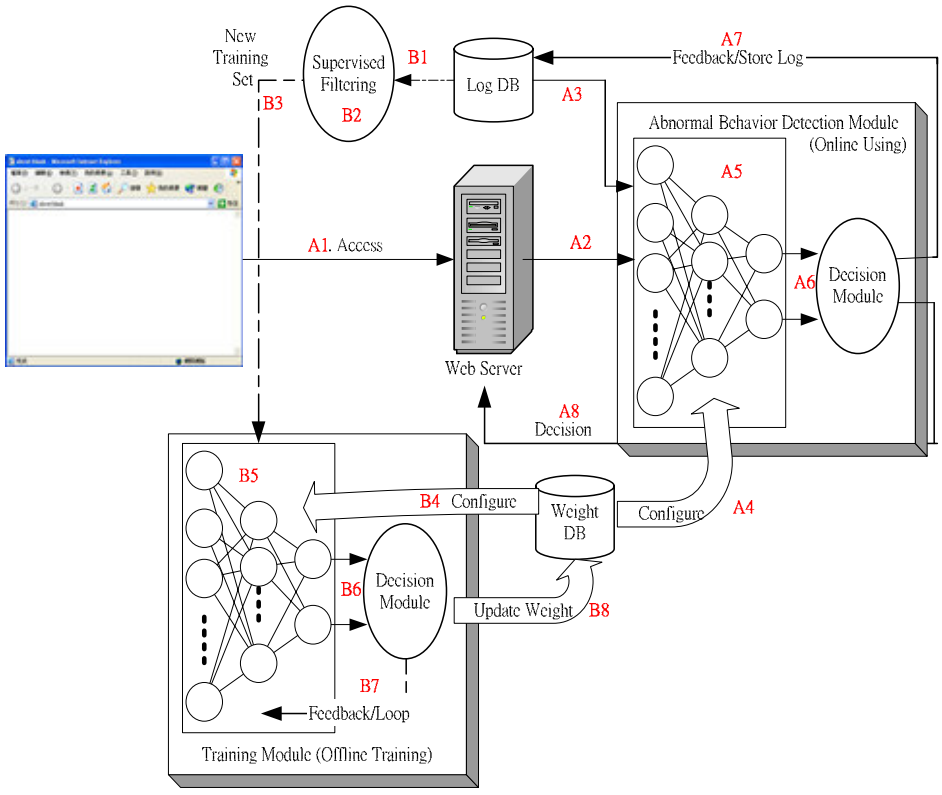

Figure 2. Hybrid training mechanism

3 Experiment and Results

To better understand the survival rate and performance of the AC system, we established a conflicting environment to test it. The system was tested for about four months. In addition to an AC, we also put a traditional counter (TC for short) on each webpage. However, the TC was invisible. We set to be 15 minutes and time-span threshold to be 10 seconds for the rule-based AC and use records of last year stored by the rule-based AC as the training set to establish the neural-network AC. At the end of the test, the blocking rate of the rule-based AC was 61% and the neural-network AC was 76%.

In this experiment system we first organize two kinds of training set. The first one is some real data collected from the Accuracy Counter last year, and the second is some simulated data that made by us. The reason that we choose to use simulated data as training set is the lack of malicious access methods in the real dataset.

All data in the training set will have several attributes, including Real IP, Virtual IP, Access Time, Browsing Period. Those attribute values are then translated into different form. Take Real IP for example, the Real IP are composed by four 0-255 numbers, when put it into neural network we translate it to 32 binary number (0/1). Beside the Real IP, the Virtual IP is a whole number will be also translated into binary number. The Access Time here will be a milliseconds value represents the number of milliseconds that have passed since January 1, 1970 00:00:00.000 GMT. Similar to the Access Time, the Browsing Period is also a milliseconds value.

The expected outputs from the neural networks are four numbers each number represents the answer of a question, including "Is it a normal access? (0/1)", "What kinds of malicious actions do I think? (000/001/010/011)", "Can I positively sure? (0/1)". Since we have input attribute values and expected outputs, the preparation for training the neural network is done.

The neural network we took for implementing the experiment system is the simplest neural network, the Back Propagation Network. We use three layers BPN, which means there is one input layer, one hidden layer and one out put layer. The neuron number in each layer is 115 (input layer), 60 (hidden layer) and 5 (output layer). The size of training set is about 3,000 real records and 320 records for four possible abnormal access patterns (each pattern has 40 positive and 40 negative records). The training epochs are 50 epochs for each kind of training set and 100 epochs for the mix sets.

The output will be judged by a decision module, once the decision module observes the output is not definitively, the record will be stored for further processing. The supervisor can log on the supervised training system and browse the current unsolved access behaviors as Figure 3 shown below.

In Figure 3 there is a Confirm button for each unsolved record, the human supervisor can either change the action type or tell the neural network whether the judgment is right or wrong by using the Confirm button. Also, sometimes there are records either can not be processed or still can not tell even by the human brain, at this situation the supervisor can choose ignore such records via check them and press the Ignore button. Finally, the supervisor can decide to make the neural network re-training based on those new records with the Online Training button.

4 Conclusions

The Accuracy Counter was designed for using in the conflicting environment such as Internet Advertising Industry and eLearning Industry. Take Internet Ad Industry for example, according to the statistics collected by Interactive Advertising Bureau (IAB) in 2002, Cost Per thousand iMpressions (CPM) is the predominant pricing model of choice in year 2002 [12][13]. With this pricing model, the advertisers can more or less ensure that potential customers really saw the ad that they have paid for.

However, all the performance reports provided by the publisher are according to the traditional Web Counter. Hence, it is hard for an advertiser to give 100% trust to the performance report furnished by a publisher, since the charge is according to the number of clicks as recorded in a Web Counter. What is in need is to further enhance the accuracy of the number of clicks as recorded in a Web Counter, so as to increase the degree of trust on the part of the advertisers in the publishing mechanism. An accuracy counter is then designed to do just that. It provides a more robust and accurate data to the advertisers.

Once again, although a rule-based detection module is developed in the Accuracy Counter, there are still many possible ways to cause the records either incorrect or not 100% correctness as we mentioned above. Malicious users may take similar means, sequences and even new vulnerabilities revealed at some time to make the data stored in database incorrect. This paper proposes a hybrid training mechanism for applying neural networks to the Accuracy Counter.

The hybrid training mechanism is not only can use in the Accuracy Counter we designed, but also can apply to

any web-based applications which need the adaptive ability. Besides, this kind of mechanism, online using/offline training, also provides other industries such like Network Security and Antivirus a workable solution.

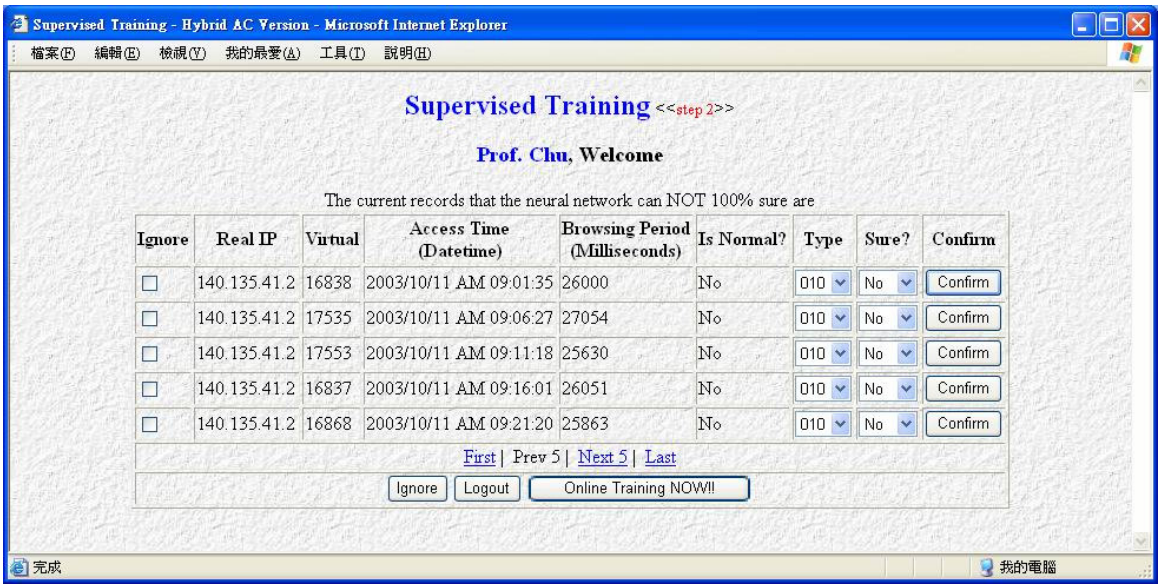

Figure 3. Supervised Training (Offline Training)

References

[1] Ko-Kang Chu and Maiga Chang (2003), "Designing a Mechanism to Assure the Reasonableness and Fairness of Information Stored in Databases on WWW," *IEEE International Conference on Systems, Man and Cybernetic 2003*, (IEEE CSMC 2003), Washington, D.C., USA, October 5-8, 2003, pp. 2302-2307

[2] Jean-Philippe Planquart (2001), "Application of Neural Networks to Intrusion Detection," *SANS Institute*, 2001, Retrieved from http://www.sans.org/rr/papers/30/336.pdf

[3] Luc Girardin (1999), "An eye on network intruder-administrator shootouts," *the Proceedings of the 1st Workshop on Intrusion Detection and Network Monitoring, Santa Clara, California, USA, April 9-12, 1999,* pp. 19-28

[4] A. Carlevarino, R. Martinotti, G. Metta and G. Sandini, "An Incremental Growing Neural Network and its Application to Robot Control," *the Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks*, (IEEE IJCNN 2000), July 24-27, 2000, Vol. 5, pp. 323-328, 2000

[5] Bernd Fritzke (1995), "Growing Grid - a self-organizing network with constant range and adaptation strength," *Neural Processing Letters*, Vol. 2, No. 5, pp. 9-13, 1995

[6] Marcus Ranum (1998), "Intrusion Detection: Challenges and Myths," *Network Flight Recorder, Inc.*, 1998

[7] Sungzoon Cho, Chigeun Han, Dae Hee Han and Hyung-Il Kim (2000), "Web-Based Keystroke Dynamics Identity Verification Using Neural Network," *Journal of Organizational Computing and Electronic Commerce*, Vol. 10, pp. 295-307, 2000

[8] Joseph Barrus and Neil C. Rowe ("A Distributed Autonomous-Agent Network-Intrusion Detection and Response System," *the Procedings of the 1998 Command and Control Research and Technology Symposium, Monterey CA,* June-July 1998

[9] Manuel Laguna and Rafael Marti, "Neural Network Prediction in a System for Optimizing Simulations," 2001, Retrieved from http://www-bus.colorado.edu/faculty/laguna/articles/nnpred.pdf, Retrieved by July 1, 2004

[10] Modjtaba Rouhani and Mohamad-Bagher Menhaj, "A Novel Neuro-Based Model Reference Adaptive Control for

A Two Link Robot Arm," Retrieved from http://mining.ubc.ca/ipmm/Papers/Rouhani.pdf, Retrieved by July 1, 2004

[11] D.K. Tasoulis, L. Vladutu, V.P. Plagianakos, A. Bezerianos and M.N. Vrahatis, "Online Neural Network Training for Automatic Ischemia Episode Detection," L. Rutkowski et al. (Eds.), *ICAISC 2004, LNAI 3070*, Berlin: Springer-Verlag, pp. 1062–1068, 2004

[12] Interactive Advertisement Bureau and PricewaterhouseCoopers, (June 12, 20003), "IAB / PwC Release Final Full Year 2002 Internet Ad Revenue Figures," *Interactive Advertisement Bureau*, Retrieved from http://www.iab.net/news/pr_2003_06_12.asp, Retrieved by June 18, 2003

[13] Interactive Advertisement Bureau, "CPM (Cost-Per-Thousand)," *Interactive Advertising Bureau UK*, Retrieved From http://www.interactivejargonguide.org/Glossary/Term/CPM+(Cost-Per-Thousand), Retrieved by Apr. 10, 2003