The Failure of A Neural Network for Solving Boolean Equations of Algebraic Attack

Rafeef Mohammed Hamza

Rafeef.hamza@qu.edu.iq

Al-Qadissiya University, College of Computer Science & Information Technology، Department of Information system

Received:- 14/9/2017 Accepted:-24/1/2018

Abstract

The neural network represents an important method for solving several problems in many applications. This paper will prove the failure of neural network with algebraic attack which makes it impossible to be used as a tool for solving the linear Boolean equations that generated after applying linearization on the nonlinear equation that generated from applied algebraic attack on the generators of stream cipher. Here when be used deferent types of neural networks such as perceptron , or Boolean neural network, will take long time without arrived to the correct solution (secret key for cipher system),compare with the mathematical method such as Gaussian elimination that given the correct solution with short time.

Keywords

Algebraic Attack, Neural Networks, Boolean equations, toycrypt

.

Physical Classification QA299.6- 433

1.Introduction

1.1.neural network

Neural network have recently been used in many fields such as pattren recognition, image processing, adaptive filtering, speech processing also used to do neural computations in linear algebra field[1,2]. But more important from all these fields using neural network in the information securety, where its represent as a branch of cryptography for use in encryption and cryptanalysis called Neural Cryptography [3,4]. Using neural network in the cryptanalysis to destroy the security of the information for known the meaning of the encrypted information or known the secret key, that what is interested with it in this paper, essentially how using it with algebraic attack. But before that must be explained its components and who its work.

A neuron in ANN is represented by a simple processing unit that has three functions:

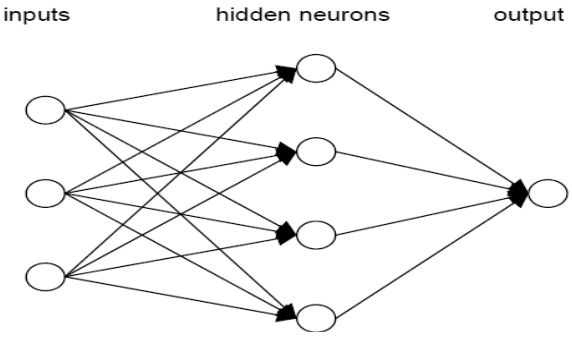
It takes one or more inputs, performs a mathematical transformation on these inputs and outputs the resulting value [5]. Where each unit(neural) has K inputs $x=\{x_1, x_2,\dots x_k\}$, and is capable of taking a number of states, each described by vector $w = (w_0, w_1, \dots, w_p) \in R^p$ of p real numbers, known as weights or parameters, each input connected with its corresponding weight and applied the active function to find the output.

$$\sigma = sgn\left(\sum_{j=1}^{N} w_{ij} x_{ij}\right) \quad (1)$$

Sigmoid is a simple example of active function, which returns -1,0 or 1:

$$sgn(x) = \begin{cases} -1 & if \quad x < 0, \\ 0 & if \; x = 0, \\ 1 & if \; x > 0. \end{cases} \quad (2)$$

The type of the active function determine how calculate the output of the neural, where there are several types of neural depends on the active function such as (linear, polynomial, spiking) [6, 5].
Also there another class of neural network called Boolean neural network where the input and output vectors are a string of binary bit (0,1), where its used for executing or implementing the Boolean functions [7,8,9].
Anural network is formed when we place units at vertices of directed graph, with the arcs of digraph representing the flows of signals between units. Some of units are called input units: these receive signals not from other units, but instead they take their signals from outside environment. Units that do not transmit signals to others units called output units. All other units that take and given signals from or to other units called hidden units. As shown in figure (1) [6].



Figure(1): Diagram of an artificial neural network

1.2.Neural network model for solving linear equations

The problem of solving linear equations is ubiquitous in numerous fields in science, engineering, and business. In mathematics, n order linear equations formulated as:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n \end{aligned} \quad (3)$$

The linear-equations problem is generally can define as follows:

$$Ax = b \quad (4)$$

Where coefficient matrix $A \in R^{n\times n}$, coefficient vector $b \in R^{n\times 1}$, and $x \in R^{n\times 1}$ is the unknown vector to be obtained.

$$A = \begin{bmatrix} a_{11}, a_{12}, \dots a_{1n} \\ a_{21}, a_{22}, \dots a_{2n} \\ \vdots \quad \vdots \quad \vdots \\ a_{n1}, a_{n2}, \dots a_{nn} \end{bmatrix}, \qquad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

and $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$

There are two general types of solution to the problem of linear equations. One is the numerical algorithms performed on digital computers. Usually, the minimal arithmetic operations for such a numerical algorithm are proportional to the cube of the coefficient matrix dimension n, i.e., $O(n^3)$ operations. Evidently, such serial-processing numerical algorithms may not be efficient enough for large-scale online or real-time application [10]. There for the neural dynamic approach is now regarded as a powerful alternative for solving a system of linear equations because of its parallel distributed nature and convenience of hardware implementation. Where proved its successfully for solving the linear and nonlinear equations with different type of neural network in many researches such as [10,11,12,13,…etc.] .

For achieving simple fast-computation characteristics, the linear activation function f(u) = u will be used for constructing all the neurons of such a BP neural network, and each neuron's threshold value is set to be zero for the same purposes. We could define the neural network weights as:

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \in R^{n\times 1} \qquad (5)$$

Evidently, the relationship between network coefficients and linear equations coefficients represent as: A is the input to the neural network, b is the ideal output of the network, and w is the approximate solution for the system equations. As shown in figure (2).That convert eq(4) to:

$$Aw \approx b \qquad (6)$$

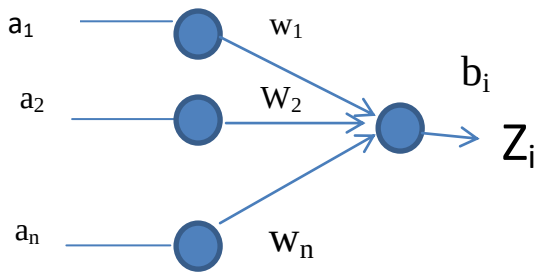


Figure2. A simple BP neural network for solving Aw ≈ b

Eventually, applying sequences of iteration of changing the weights of network, will leads to the approximate solution to the equations [1,10]

2.Algebraic attack

The algebraic attack is a new cryptanalysis method to destroy the secret message by finding the secret key. Its convert the problem from finding the secret key (initial value) to solving multivariable system of algebraic equations, where an algebraic equation is related to the initial key bits and the output keystream bits, and then it is solved through linearization techniques or any other proper techniques, where the solution of the system equations represent the initial value (secret key) [14][15].

Algebraic attack applied on several cipher system, in this paper will take algebraic attack on the toycrypt system as example on the combiner generator without memory for stream cipher, where it consist of two registers, where the system of equations for it ,will represented with binary matrix as in eq(11). After then insert the key stream (Z) to the equation, to calculate the variables (Si) that represent secret key(initial value) [16].

3. Applying neural network on Boolean equations of the algebraic attack

As we mentioned in the previous, successful using neural networks for solving the linear and nonlinear system of equations and finding it's approximate solutions. And successful applying algebraic attack on several systems by using mathematical tools such as Gaussian elimination.

Now, the algebraic attack will be applied by using neural network as a tool. But when we applying neural network on the system of

Boolean equations, must be make some change on the operation of the classical network to sufficient the nature of Boolean equations, where the input, output, and weights of the network must be (0,1), and the operation that used in the connected among them and a modification the weights must be Boolean operation (AND, XOR).

$$\sigma = \bigoplus_{i=1}^{n} x_i \bullet w_i \quad (7)$$

$$f = \begin{cases} 0, & \sigma < t \\ 1, & \sigma \geq t \end{cases} \quad (8)$$

Where f refer to the active function (actual output of the neuron), t is a threshold =0.5.

$$w_{i+1} = w_i \oplus \eta x_i \left| (z_i - f_i) \right| \quad (9)$$

$$\text{error} = \sum_{i=1}^{p} (z_i - f_i) \quad (10)$$

$\oplus$: mean (xor function), η : is the learning rate equal to 1, z_i is the ideal output of the equation(pattern) i, and will choice w_0 initial weights equal to zeros .

Before applying neural network on the algebraic attack, will be attempt to apply it on the simple system of Boolean equations to prove failure or corrected its work.

Example 1:

$$\begin{aligned} x_1 + x_2 + x_4 &= 1 \\ x_2 + x_3 + x_4 &= 0 \\ x_1 + x_4 &= 0 \\ x_1 + x_2 + x_3 &= 0 \end{aligned}$$

Must mentioned (+) operation her refer to (xor) operation in the Boolean system. This system have four variables can be represent as input coefficient matrix M, and ideal output as a vector Z=(1,0,0,0).

$$M = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}, \quad z = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Can be solving this system using the neural networks with the same time of Gaussian elimination for arriving to the initial value (x_1=1, x_2= 1, x_3=0, x_4=1).

Example 2:

$$\begin{aligned} x_1 + x_2 + x_4 + x_5 &= 0 \\ x_1 + x_3 + x_4 &= 0 \\ x_2 + x_3 + x_4 &= 0 \\ x_1 + x_4 + x_5 &= 1 \\ x_1 + x_2 + x_3 + x_5 &= 1 \\ x_1 + x_3 + x_4 + x_5 &= 1 \end{aligned}$$

This system have five variables can be represents as:

$$M = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 \end{bmatrix}, \quad z = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

The solution of this system as (x_1=1, x_2=1, x_3=0, x_4=1, x_5=1), can be found using neural network and Gaussian elimination with the same time.

After applying neural network for solving several systems of Boolean equations, its proved successfully with system that have less than 7 variables, while failed with the systems that have more than 7 variable. Where when be using for solving system with 8 variables cannot arrived to the correct solution until 100000 iteration through around 4 minute. This time is able to increment with increment the number of the variables and the equations. Also when using it for solving small system cannot be found the correct result such as:

$$\begin{aligned} x_1 + x_2 &= 1 \\ x_2 + x_3 &= 1 \\ x_1 + x_3 &= 0 \end{aligned}$$

This system if we solved manually can be found two correct result given the same output these are : either (x_1=1, x_2=0, x_3=1), or (x_1=0, x_2=1, x_3=0). These multi choice of solutions make the network entered on the open looping without stopped, because it cannot detected any one solution is the correct.

Now, neural network will be applied on the system of Boolean equations that result from applied algebraic attack on the toyocryt cipher

where its display on eq(11). But it must be mentioned that the practical attack does not need all these equations, rather it needs at least number of steps equal to the number of monomials that are expected to appear in the system. We can describe system of toycrypt equation with multiplying the matrix(coefficients of monomials) with the vector of monomials and equal the final to vector z (keystream) as shown in eq(11)[16].

$$\begin{pmatrix} 1\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,1\,0\,0\,0\,0\,1\,0\,0 \\ 1\,1\,0\,0\,1\,0\,0\,0\,0\,1\,1 \\ 1\,0\,1\,0\,1\,1\,0\,0\,0\,1\,0 \\ 0\,1\,1\,1\,1\,0\,1\,0\,1\,0\,1 \\ 1\,1\,1\,1\,0\,1\,1\,1\,1\,0\,0 \\ 1\,0\,0\,1\,1\,0\,0\,1\,0\,1\,0 \\ 0\,1\,1\,0\,0\,0\,1\,0\,0\,0\,0 \\ 1\,1\,0\,1\,0\,0\,0\,1\,1\,0\,0 \\ 1\,0\,0\,0\,1\,0\,0\,0\,0\,1\,0 \\ 0\,1\,1\,0\,1\,0\,1\,0\,0\,0\,1 \end{pmatrix} \bullet \begin{pmatrix} s_0^{(1)} \\ s_1^{(1)} \\ s_0^{(2)} \\ s_1^{(2)} \\ s_2^{(2)} \\ s_0^{(1)}.s_0^{(2)} \\ s_1^{(1)}.s_0^{(2)} \\ s_0^{(1)}.s_1^{(2)} \\ s_1^{(1)}.s_1^{(2)} \\ s_0^{(1)}.s_2^{(2)} \\ s_1^{(1)}.s_2^{(2)} \end{pmatrix} = \begin{pmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \\ z_8 \\ z_9 \\ z_{10} \end{pmatrix} \qquad (11)$$

After insert the keystream on the system as example (Z=11111111011), can be solved this system using Gaussian elimination with a round 0.03105 s and arrived to the correct solution (secret key) for the first register as ($s_0^{(1)}$=1, $s_1^{(1)}$=1), and for the second register as ($s_0^{(2)}=1, s_1^{(2)}=0, s_2^{(2)}=1$).

While using neural network model for solving the same system of Boolean equations, will be arrived to the 100000 iterations in 6 minute around without appear the correct result (correct secret key).That proved the failure of using neural networks for solving the system of Boolean equations of algebraic attack.

4. Conclusion

As mentioned above, neural network using in many application among them mathematical fields, where its successful for solving linear and nonlinear system of equations using classical neural network. Also can be used Boolean neural network for implement and execute the Boolean functions. But when we using the classical neural network for solving system of Boolean equations that generated from algebraic attack, then failure because the larger and real value of the weights of the network that represent the solution, where it's not convenient with the desired solution(binary secret key). Therefor have be used Boolean neural network with some change on the operation to convenient with the nature of the system, where weight must be as binary number and applied (xor) operation in the activation function. This type of network successful with system have small number variables until 6 or 7 variables, can be solved these system with the similar time of using mathematical methods (Gaussian). But its **failure** with linear system of Boolean equation such as that generated from applied algebraic attack on the toycrypt cipher, where its arrived to 100000 in time around 6 minute without appear the correct solution (secret key), and that time able to increase with increment the number of variables. As you know in simple cipher system must there are large number of variables, that make the algebraic attack failure since it cannot found the secret key in the desire time, where the time represent the challenge on the any type of attack. While can solve the system of equations and find the secret key using Gaussian elimination with small time arrived to 0.0305s around for toycrypt. Must be mentioned the number of equations must be at least equal or larger than the number of the variables, when we using Gaussian elimination or neural network.

5.Refernces

1- De-shuang huang and Zheru chi,(2002), "Solving linear simultaneous equations by constraining learning neural networks", NFS of china.

2-Prompong Sugunnasil, Samerkae Somhom, Watcharee Jumpamule, and Natee Tongsiri,(2014), " Modelling a neural network using an algebraic method", research article Science Asia 40: 94–100.

3-Khalil Shihab,(2006)," Aback propagation neural networks for computer network security",ISSN,Journal of computer science 2(9):710-715.

4-Andreas Ruttor , (2006), "Neural Synchronization and Cryptography", PhD thesis, Bayerische Julius-Maximilians-Universität Würzburg.

5- David Christopher Wedge,(2006),"Wave Overtopping Prediction Using Global-Local Artificial Neural Networks", Thesis of Doctor of Philosophy, Department of computing and

Mathematics, Manchester Metropolitan University.

6-Marthin Anthony,(2003),"Boolean functions and Artificial Neural networks", CDAM research report LSE.

7-Xiaomin MA, Xian Yang YI, and Zhaozhi zhang, (1999),"Boolean Neural network Design using set covering in Hamming Geometrical space", IEICE trans. Fundamentals,vol.E82-A.

8-Leonardo franco ,2006, "Generalization ability of Boolean functions implemented in feed forward neural networks", Elsevier, Neurocomputing 70:351-361.

9-Nripendra N. Biswas, T.V.M.K.Murthy, and M.Chandrasakhar,"IMS algorithm for learning representations in Boolean neural networks", grand from council of scientific and industrial research, NewDelhi, India.

10- Yunong Zhang, Zhan Li, Ke Chen, and Binghuang Cai,(2008) , "Common Nature of Learning Exemplified by BP and Hopfield Neural Networks for Solving Online a System of Linear Equations", IEEE, 978-1-4244-1686-8.

11- Karl Mathia and Richard Saeks, (1995), " Solving Nonlinear Equations Using Recurrent Neural Networks", World Congress on Neural Networks (WCNN'95).

12-Deepak Mishra, Prem K. Kalra,(2007), "Modified Hopfield Neural Network Approach for Solving Nonlinear Algebraic Equations", Engineering Letters, 14:1.

13-Minrui Fei, Jian Zhang, Huosheng Hu and Taicheng Yang,(2006), "A novel linear recurrent neural network for multivariable system identification", Transactions of the Institute of Measurement and Control, pp. 229_242.

14- Martin voros,(2007),"algebraic attack on the stream cipher", Master thesis submitted to Comenius University, Faculty of mathematics, physics and informatics, department of computer science.

15-Cameron McDonald,(2010), " Analysis of Modern Cryptographic Primitives", Ph.D. thesis submitted to Macquarie University, Department of Computing, Faculty of Science.

16-Susil Kumar Bishoi,(2010)," Efficient Solution of Large Sparse Linear Equations over GF2 for Algebraic Attacks On Stream Ciphers", Master project report submitted to Supercomputer Education and Research Centre, Indian Institute of Science, Bangalore - 560 01.

فشل الشبكة العصبية في حل المعادلات البوليانية للهجوم جبري

رفيف محمد حمزة

Rafif_prog@yahoo.com

Rafeef.hamza@qu.edu.iq

جامعة القادسية ، كلية علوم الحاسوب وتكنلوجيا المعلومات ، قسم نظم المعلومات الحاسوبيه

تاريخ الاستلام:- 2017/9/14 تاريخ القبول :- 2018/1/24

الخلاصة

تمثل الشبكة العصبية وسيلة هامة لحل العديد من المشاكل الموجودة في العديد من التطبيقات . في هذه البحث سيتم إثبات فشل استخدامها مع الهجوم الجبري حيث لا يمكن أن تستخدم كأداة لحل انظمة المعادلات المنطقية الخطية التي نتجت من تطبيق ألية (linearization) على المعادلات الغير خطية المتولده من الهجوم الجبري على مولدات انظمة التشفير الانسيابي. حيث انه عند استخدام النوع الاعتيادي من الشبكة العصبية مثل perceptron ، أو الشبكة العصبية المنطقية، سوف يستغرق وقتا طويلاً جداً دون أن يصل إلى الحل الصحيح والذي يمثل (المفتاح السري لانظمة التشفير) ، مقارنة مع أسلوب رياضي مثل Gaussian التي ممكن ان تصل إلى الحل الصحيح بوقت القصير .

الكلمات المفتاحية

الهجوم الجبري ، الشبكات العصبية ، المعادلات البوليانية ، toycrypt