# Neural Solutions for Information Security

Şeref SAĞIROĞLU*, Necla ÖZKAYA**
*Gazi University, Engineering and Architecture Faculty
Computer Engineering Department
06570 Maltepe, ANKARA
**Erciyes University, Engineering Faculty, Computer Engineering Department
38039, KAYSERİ

## ABSTRACT

This paper presents a new approach based on artificial neural networks (ANNs) for data security in electronic communication. Two ANN modules were used for encryption and decryption. ANN modules were trained with Levenberg-Marquardt learning algorithm. Character based learning and test processes were followed. A program developed in Delphi achieves the processes automatically. The results have shown that the cryptosystem based on ANNs is very successful and also provides more security with the help of adding an extra module. It is thought that the study presented in this work will enable new directions to the applications because of real-time application of ANNs.

**Key Words :** artificial neural network, information security, encryption, decryption, security

# Bilgi Güvenliği için Nörol Çözümler

## ÖZET

Bu makalede elektronik ortamlarda veri güvenliği için yapay sinir ağlarına (YSA) dayalı yeni bir yaklaşım sunulmuştur. Şifreleme ve deşifreleme için iki YSA modeli kullanılmıştır. Levenberg-Marquardt öğrenme algoritması kullanılarak YSA modelleri eğitilmiştir Karakter tabanlı bir eğitim ve test işlemi gerçekleştirilmiştir. Bu işlemleri otomatik olarak gerçekleştirmek için ise Delphi de bir program geliştirilmiştir. Elde edilen sonuçlardan, YSA tabanlı şifreleme ve deşifreleme işlemlerinin başarılı olduğu, ekstra bir modüllede güvenlik seviyesinin arttırılabileceği tespit edilmiştir. YSA tabanlı sunulan çalışmanın, gerçek zamanlı olarak ta gerçekleştirilebilecek olmasının gelecekte yeni çalışmalara yön verebileceği değerlendirilmektedir.

**Anahtar Kelimeler :** yapay sinir ağları, bilgi güvenliği, şifreleme, şifre çözme, güvenlik

## 1. INTRODUCTION

Cryptography is the study of mathematical techniques related to aspects of information security for personal, industrial or organizational usage. It covers the *data integrity* addressing the unauthorized alteration of insertion, deletion and substitution, the *authentication* being related to identification of two parties entering into a communication, the *confidentiality* being kept the content of information from all but those authorized to have it, and the *non-repudiation* providing entity prevention from denying previous commitments or actions (1). A cryptosystem refers to a set of cryptographic primitives used to provide information security services. Basic primitives used to provide information security are evaluated with respect to level of security, functionality methods of operation, performance and ease of implementation (1-5).

In the field of cryptography, one is interested in methods to transmit messages secretly between two parts. One (an opponent) who is able to listen to the communication should not be able to recover the secret message. Today a common secret key based on number theory could be created over a public channel accessible to any opponent but it might not be possible to calculate the discrete logarithm of sufficiently large numbers (1-3, 5). DES, IDEA, RC5, CAST, BLOWFISH, 3DES and RSA are the well-known and the mostly used (preferred) encryption and decryption systems. In general the cost of these systems is high and it requires more time for computation and applications.

Artificial neural networks (ANNs) have been applied to solve many problems (6). Learning, generalization, less data requirement, fast computation, ease of implementation, and software and hardware availability features have made ANNs very attractive for the applications. These fascinating features have also made them popular in cryptography as well (7-14).

Secure exchange of information by neural synchronization has been recently introduced by Kanter et.al (8). In the work, two neural networks in which each network tries to learn from the other network's outputs on common random inputs were presented. Kinzel and Kanter (9) have suggested a way that how neural networks produce a common secret key by exchanging bits over a public channel and by learning from each other. The importance of this work is also based on synchronisation. In those works, there are still possibilities to have encrypted messages. Two neural modules could learn encryption and decryption processes, and produce a common secret key for a public channel. ANNs have been also used in encryption and decryption processes (12-14). In the

*Digital Object Identifier 10.2339/2007.10.1.21-25*

works, character based learning and test processes were applied for encryption and decryption of Turkish and English letters with the help of ANNs. 26 and 29 letters were considered for encryption and decryption.

In this study, 95 characters used in written communication have been encrypted and decrypted with the help of ANNs for the first time. These characters are: (space)!"$%&'()*+-.,/0123456789:;<=>?@{} ABC ÇDEFGĞHIİJKLMNOÖPRSŞTUÜVYZ()`abcçdefgğhı ijk lmnoöprsştuüvyz.

## 2. ARTIFICIAL NEURAL NETWORKS

ANNs have been applied to solve many problems (6). A general form of an ANN is shown in Figure 1. It consists of three layers. One or more hidden layers might be used in the structure. Neurons in the input layer can be treated as buffer and distribute $x_i$ input signal to neurons in hidden layer. Output of each neuron $j$ in the hidden layer is obtained from sum of multiplication of all input signals $x_i$ and weights $w_{ji}$ that follows all these input signals. The sum can be calculated as a function of $y_j$ and can be expressed as:

$$y_j = f\left(\sum w_{ji} x_i\right) \tag{1}$$

where $f$ can be a simple threshold function, or a sigmoid function.The outputs of the neurons in other layers are calculated with the same way. ANNs might be trained with many different learning algorithms (6,15). Levenberg-Marquardt (LM) learning algorithm is used to train ANNs in this work. The weights are adopted according to the error occurred in the calculation with the help of a learning algorithm.

Basically LM is a calculation method and based on maximum neighbourhood idea and it combines of the best features of Gauss-Newton and steepest-descent algorithms and removes constraints of these two algorithms (15). In this approach, assuming $E(w)$ be an error function, for m number of error term, $e_i^2(w)$ is in the form of

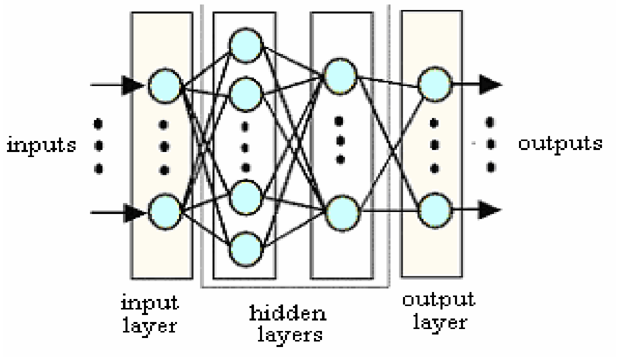


Figure 1. General form of ANN

$$E(w) = \sum_{i=1}^{m} e_i^2(w) = \|f(w)\|^2 \tag{2}$$

In this equation $e_i(w) \equiv (y_i^{(t)} - y_i)$. The target in LM, defining parameter vector $w$ when $E(w)$ is minimum. New weight vector $w_{k+1}$ is calculated from former weight vector $w_k$ by using LM. The weights are adapted according to

$$w_{k+1} = w_k + \delta w_k \tag{3}$$

where $\delta w_k$ is the factor that is computed from

$$(J_k^T J_k + \lambda I)\delta w_k = -J_k^T f(w_k) \tag{4}$$

where $J$ is Jacobian of $f'$ at $w_k$ weight, $\lambda$ is Marquardt parameter and $I$ is identity unit matrix. LM can be summarized:

*(a)* Calculate $E(w_k)$,

*(b)* Initialize $\lambda$ with a small value

*(c)* Solve equation (4) for $w_k$ and calculate $E(w_k + \delta w_k)$ value,

*(d)* If $E(w_k + \delta w_k) \geq E(w_k)$ then increase $\lambda$ 10 times and go to (c),

*(e)* If $E(w_k + \delta w_k) < E(w_k)$ then decrease $\lambda$ 10 times $w_k$: $w_k \longleftarrow w_k + \delta w_k$ and go to (c).

## 3. ANN BASED ENCRYPTION AND DECRYPTION

A block diagram for neural encryption and decryption modules was shown in Figure 2. In the block, A and B represent the source and the destination, respectively. In order to send the message from A to B securely, the message in plaintext is first converted in binary form. Binary forms of the message are applied to the neural encryption module as inputs and outputs. The neural encryption module encrypts the binary form of the message into different sequences. The outputs to the neural module were achieved from different orders of input sets. The decryption module is also trained in A with the established data sets. The both neural modules were trained with the Levenberg-Marquardt learning algorithm.

In order to test the neural systems for this approach, three different training sets were randomly established. After training, the neural models were tested. The ANN based encryption and decryption processes were successful for the three sets.

The key source belonging to the neural decryption module are the number of neurons in the input, hidden and output layers, the number of hidden layer, the weights, the biases and the type of transfer function used in layers. This source is transmitted to B from A only once in a secure channel. This channel might be a phone, post, fax, or a special delivery. Even if the key source had been achieved from one in the secure channel, it should be noticed that, the decimal numbers belonging to the source (weights, number of neurons and layers, type of activation function, etc.) would not make any sense to the opponent.

After receiving the source key from A, the neural decryption module is reconstructed in B. When B achieves the message from A via insecure channel, the message is applied to the neural decryption module. The binary form of the inputs obtained from the module is then converted to the plaintext or to the original form of the message/document. Thus, the communication is achieved securely.

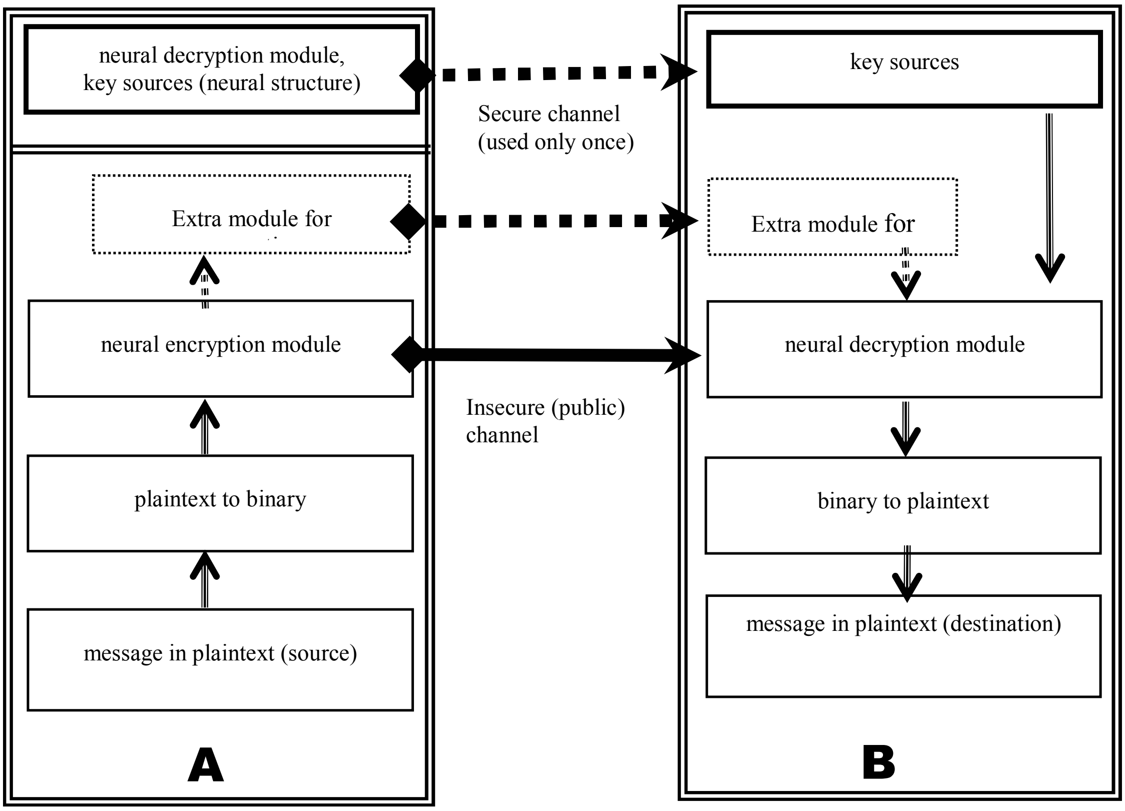


Figure 2. Block diagram for neural encryption and decryption processes

In order to increase the security, an extra encryption module is designed and might be used as shown in Figure 2. This module inserts random binary numbers into the encrypted message line within a sequence and provides more complexity to the encrypted messages and also increases the security. If this module is selected in encryption, the binary numbers inserted randomly must be filtered before the decryption module. This filtering is achieved by the extra decryption module. In this case, the insertion sequence parameter must be known by B for proper filtering. Also, this parameter should be sent to B from A using secure channel once again or the parameter can be sent to B at once with the source key together.

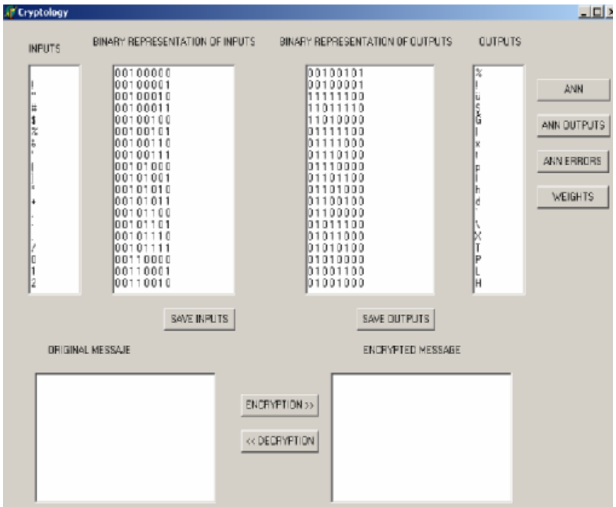
Even if a cryptanalyst gets the messages from insecure channel, he/she might not be able to understand the sequence of the binary codes in public channel. Moreover, the binary numbers are converted to the large decimal numbers within a sequence in A. This raises the security and the numbers could not be distinguished by an opponent.

When dual channel communication is required, the both neural modules must exist in A and B. In this case, the source keys belonging to the neural encryption and decryption modules must be sent once to A or B with the use of a secure channel.

Total 95 letters and characters used in writing are considered for precise communication. Each neural module used in this work has 8 input and output neurons because of representing the letter or character within 8 bits. Two hidden layers were used in the both neural structures and 8 and 16 neurons were selected in the first and the second hidden layers, respectively. Each neural module used in the process required at least 100 epochs in training. Training process is stopped when a satisfactory result is achieved from the neural modules.

These processes can be achieved automatically by a computer program written in Delphi. The screenshots of the developed software presented in this work are shown in Figures 3 and 4. The windows of inputs, outputs and binary representations, the buttons of ANN training, errors, outputs, weights, the saving input and output options, and the buttons for encryption and decryption belonging to the developed software can be used easily in designed neural cryptosystem (see Figure 3). The selections of ANN parameters for the both neural processes are given in Figure 4 for the developed software as well. All neural parameters can be easily selected from the menu screen given in Figure 4. These both user friendly screens belonging to the developed software help the users to design the neural modules for encryption and decryption processes.

The ANN parameters (weights, number of neurons and layers, type of activation function) are used as a secret key for the cryptosystem presented in this work. To understand the encrypted message from these parameters is really difficult for a cryptanalyst. Disordering these neural parameters can give unexpected results which help to protect the message from opponents.



| INPUTS | BINARY REPRESENTATION OF INPUTS | BINARY REPRESENTATION OF OUTPUTS | OUTPUTS |
|---|---|---|---|
|  | 00100000 | 00100101 | % |
| ! | 00100001 | 00100001 | ! |
| " | 00100010 | 11111100 | ü |
| # | 00100011 | 11011110 | Þ |
| $ | 00100100 | 11010000 | Ð |
| % | 00100101 | 01111100 | \| |
| & | 00100110 | 01111000 | x |
| ' | 00100111 | 01110100 | t |
| ( | 00101000 | 01110000 | p |
| ) | 00101001 | 01101100 | l |
| * | 00101010 | 01101000 | h |
| + | 00101011 | 01100100 | d |
| , | 00101100 | 01100000 | ` |
| - | 00101101 | 01011100 | \ |
| . | 00101110 | 01011000 | X |
| / | 00101111 | 01010100 | T |
| 0 | 00110000 | 01010000 | P |
| 1 | 00110001 | 01001100 | L |
| 2 | 00110010 | 01001000 | H |

Figure 3. screenshot for the developed software.

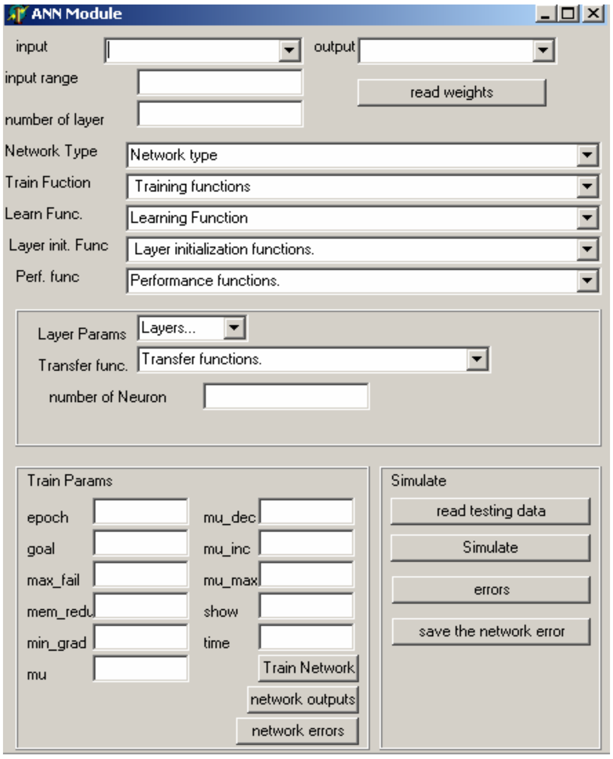


Figure 4. ANN module for the developed software.

## 4. CONCLUSIONS

In this study a new approach based on ANNs for data security has been successfully introduced. Data security is achieved with the help of neural encryption and decryption modules. 95 possibilities with 8-bit representation also increase the security. The frequency traps of the encrypted binary message are even more difficult to achieve with the help of the extra encryption module.

Finally, with the available software, the neural solution for data security can be achieved easily, effectively, friendly and securely. This system can be used for personal information security. Possible hardware applicability of the neural cryptosystem might be another important features of the approach presented in this work.

## REFERENCES

1. Menezes, A. J., P. C. Oorschot and S. A. Vanstone, Hanbook of Applied Cryptography, CRC Press, Ontorio, Canada, 2002.
2. Seberry, J. and J. Pieprzyk, Cryptography: An Introduction to Computer Security, Prentice-Hall, 1989.
3. Stinson, D., Cryptography; Theory and Practice, CRC Press Inc., 2nd Edition, 1996.
4. Koblitz, N., A Course in Number Theory and Cryptography, Springer, 1994, 2nd Edition, 1994.
5. http://www.nsa.gov/about_nsa/faqs_ internet. html, The National Security Agency,
6. Haykin, S., Neural Networks: A Comprehensive Foundation, Macmillan College Publishing Company, New York, USA, 1994.
7. Cin, İ. , Using artificial neural networks for encryption (in Turkish), MSc. Thesis, Osman Gazi University, Eskişehir, Turkey, 1996.
8. Kanter,I, W. Kinzel and E. Kanter, Secure exchange of information by synchorization of neural networks, Europhys. lett , 57, 141-147, 2002.
9. Kinzel, W. and I. Kanter, Interacting Neural Networks and Cryptography, arXiv:cond-mat/0203011, 1, 1, 1-9, 2002.
10. Klimov, A., A. Mityaguine and A. Shamir, Analysis of neural cryptography, AsiaCrypt 2002, 1-5 Dec. 2002, Queenstown, New Zeland, Submitted for publication.
11. Tanrıverdi, H., Artificial neural networks for cryptography (in Turkish), MSc. Thesis, METU, Ankara, Turkey, 1993.
12. Sağıroğlu Ş., N. Demirayak ve T. Baydar, Artificial neural networks for encryption and decryption (in Turkish): A new approach (in Turkish), GAP IV. Engineering Congress (with International participants), 6-8 June 2002, Şanlıurfa, Turkey, 1, 527-531, 2002.
13. Sağıroğlu Ş., N. Demirayak ve T. Baydar, Artificial neural networks for cryptography (in Turkish), Bilişim Kurultayı, pp.37-39, 3-6 Sept 2002.

14. Sağıroğlu, Ş., N. Demirayak and T. Baydar, Encryption and decryption of all characters in writing communication with ANNs. (in Turkish): ELECO '02, 8-22 December 2002, Bursa, Turkey, ISBN:975-395-566-9, Vol.Electric-Computer, 351-354, 2002.

15. Pham, D. T. and S. Sagiroglu, Three methods of training multi-layer perceptrons to model a robot sensor. Int. J. of Robotica, 13, 531-538, 1995