International Journal of Electrical and Computer Engineering (IJECE)
Vol. 12, No. 3, June 2022, pp. 2756~2764
ISSN: 2088-8708, DOI: 10.11591/ijece.v12i3.pp2756-2764

Data protection based neural cryptography and deoxyribonucleic acid

Sahar Adill Kadum, Ali Yakoob Al-Sultan, Najlaa Adnan Hadie
Department of Computer Science, College of Science for Women, Babylon University, Hillah, Iraq

Article Info

Article history:

Received Nov 20, 2020
Revised Jan 7, 2022
Accepted Jan 19, 2022

Keywords:

Bio-molecular
DNA computing
DNA sequence
Hebbian network
Neural cryptography

ABSTRACT

The need to a robust and effective methods for secure data transferring makes the more credible. Two disciplines for data encryption presented in this paper: machine learning and deoxyribonucleic acid (DNA) to achieve the above goal and following common goals: prevent unauthorized access and eavesdropper. They used as powerful tool in cryptography. This paper grounded first on a two modified Hebbian neural network (MHNN) as a machine learning tool for message encryption in an unsupervised method. These two modified Hebbian neural nets classified as a: learning neural net (LNN) for generating optimal key ciphering and ciphering neural net CNN) for coding the plaintext using the LNN keys. The second granulation using DNA nucleated to increase data confusion and compression. Exploiting the DNA computing operations to upgrade data transmission security over the open nets. The results approved that the method is effective in protect the transferring data in a secure manner in less time

Corresponding Author:

Sahar Adill Kadum
Department of Computer Science, College of Science for Women, Babylon University
Hillah, Iraq
Email: dr.sahar.adill@gmail.com

1. INTRODUCTION

Security concerns the safety of the network and the transmission of data. Data security is the most critical component of the secure transmission of data over the network. Today, however, achieving maximum data protection is a difficult problem for data communications over open networks. There is more than one technology used to protect data, such as encryption and information concealment. Encryption codes a message that cannot be read by an accidental individual. Whereas, hiding information is a technique used to conceal a hidden message so that it cannot be identified by accidental users [1], [2]. Three goal purposes from using cryptography: first, to achieve confidentiality. Second, authentic the sender and thirdly, the integrity. The modern cryptography concentrated on Kirchhoff principle: assume that the attacker known all or some details of encryption method usage, and the working procedure, except key data piece. Generally, the cryptography can be classified to two classes: symmetric and asymmetric systems. The symmetry system in turn classified into two ciphering types: block cipher system and stream cipher system [3] as shown in Figure 1.

Effective coding techniques are needed to achieve an effective data protection because encryption method can be down and not robust enough to offer successful data protection. Hence, unauthorized user or intruders can access the information for various harmful purposes. This leads to the need for adopting additional supported techniques for cryptography methods to overcome any falls or gaps resulted from weakness methods such as neural network encryption module concept and deoxyribonucleic acid (DNA) computing. They incorporated with cryptography and information hiding as a new hope for a robust

cryptographic techniques and unbreakable algorithms [2]. Neural networks have several properties, the most important of them is the generalization capability and parallel implementation. The special property of neural networks is confusion caused from non-linearity structure of the net, this confusion property is preferable for cipher design. Many types of artificial neural networks (ANNs) have built each ANN has own style design and method [4], one of these types is the Hebbian neural network (HNN). It is a computational tool inspired by biological nervous system. HNN is a network composed of arrays of artificial neurons linked together with various connection weights, its rule is a rule of learning that explains how the neuronal connection is affected by neuronal activities (synaptic plasticity). Therefore, it provides an algorithm for modifying the neuronal network relationship weight. The Hebb rule offers a simplified model based on physiology to simulate synaptic plasticity's activity-dependent characteristics and has been widely used in the artificial neural network field [5]–[9].

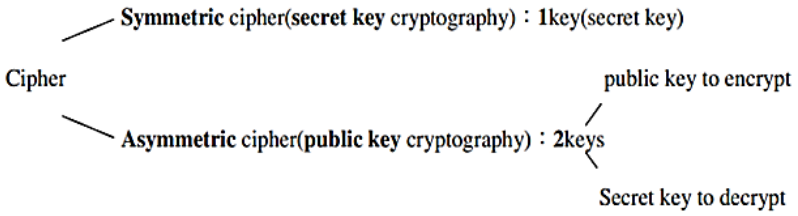

Figure 1. Encryption classes [3]

While, DNA computing is an emerging computing branch that utilizes DNA biochemistry. In this sphere, research and development concerns theory, experiments, and DNA computing applications. The originally field began with Len Adleman's demonstration of a computer application in 1994, it has now been extended to many other avenues, such as the advancement of storage technologies, nanoscale imaging modalities. synthetic controllers and reaction networks [10], [11]. Massive parallelism is the promise of DNA computing: with a given setup and enough DNA through parallel search once can theoretically solve huge problems. This could be much faster than a traditional computer, for which large quantities of hardware will be needed for massive parallelism, not just more DNA [10], [12].

Several research papers have been prepared in the field of cryptography dealing with neural nets and DNA with various encryption methods and techniques, such as Kulkarni presents a new proposal schedule for XOR design is considered. they replace the normal XOR function with neural network XOR function in the design process, the motivation for this is ability of neural networks to perform complex mapping function to generate key ciphering [13]. Jagtap *et al.* [14] presents a cryptography based on artificial neural network. The used artificial neural network (ANN) has many features such as learning, generation, less data requirement, fast computation, and simple deployment. Mohammed [15] proposed a symmetric cryptography coupled with a multilevel chaotic neural network (NN), the encryption algorithm process the data as a block. The proposed algorithm proved the efficiency of execution time in encrypt/decrypt long messages by short time and small memory, the system uses secret keys with array of keys (weights) that change at each iteration. Roy *et al.* [16] for text messages using a time-varying delayed hopfield neural network and a cryptographic posterior DNA, a new encryption model was proposed. To produce a binary sequence that is later transferred to a permutation function, the chaotic neural network used here is used and the first level encryption key is generated. Dixit *et al* [17] proposed a DNA based cryptography to achieve cryptographic strength using least significant bits (LSB) method. Applying binary index compression technique to reduces data up to 50% to improve payload capacity. Output is a sequences of DNA nucleotides. Basu *et al.* [18] presents a system of encryption based on the central dogma of molecular biology for encryption/decryption algorithms. The bidirectional associative memory neural network (BAMNN) has been used for key generation in order to saving memory space by restoring and regenerating key sets in a recurrent process. Namasudra *et al*. [19] proposed a DNA based data encryption scheme for cloud computing era. Generating key of 1024-bit is depend on DNA computing, user's attributes and media access control (MAC) address, and decimal encoding rule, American standard code for information interchange (ASCII values), DNA bases and complementary rule are to protect against many security attacks. Li *et al.* [20] proposed a new method of data protection and user authentication using DNA QR coding to be very fast, reliable and less predictable. Any message is translated to a DNA sequence and converted to a QR code called a DNA QR code. Singh and Naidu [21] proposes a method to authenticate the user using DNA sequence at first level and secondly secure the data using DNA sequence and Armstrong number. Malathi *et al*. [22] proposed a modified DNA insertion

algorithm due to its low cracking probability. Confidential information, such as confidential messages and document images, is concealed inside the DNA sequence. Measurement output is performed using cracking chance, bits per nucleotide (BPN), payload and power. Indrasena *et al*. [23] presents a bio-inspired cryptographic DNA system. The scheme is simulating the genetic encoding procedures (transcription and translation), i.e., the central dogma of molecular biology (CMDB) using BAMNN and whale optimization algorithm (WOA) as a highest fit weight vector. Volna *et al*. [24] presents new direction of cryptography based neural networks, where the cryptographic scheme is generated automatically. The proposal evolving neural network architecture called Spectrum-diverse unified neuro evolution architecture to achieve automatic encryption and decryption subsequently using adversarial training. The main purpose behind this paper is to apply a new encryption method based on two modified HNN for encryption phase and DNA computing operations for another coding type and compression to increase the concept of data confusion and confidential secrecy. The paper organized according to the following topics: proposed method, modified Hebbian neural network (MHNN) structure, research method, results, and conclusion

2. PROPOSED METHOD

The proposed system adopt hybrid securing method for transferring data in an open net by implying two contribution methods; The first one, adopt a new ciphering method using two MHNN. The second, exploit the DNA computing operations to compress the data before transferring process and increase the probability of cracking. Figure 2 illustrates the general structure of the proposed system.

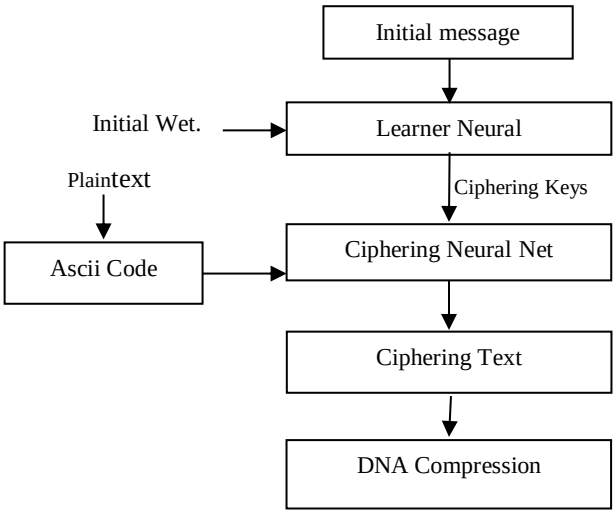

Figure 2. The proposal general structure

2.1. Modified Hebbian neural network (MHNN)

The suggested system designated by adopts the Hebbian neural network infrastructure [5]–[9], but with several modifications has been done on HNN to the behalf of the proposed system. These modifications include: i) the number of input nodes and output nodes are equally; ii) the learning process is an unsupervised approach; iii) it is a single layer (no hidden layer), this topology provides an ability of simple implementation and fast learning speed compared to other networks with hidden layers; and iv) the propagation of the signals in the network will be in one direction only (feedforward), therefore each neuron will depend on the directed input signals only. There is no activation function as in the conventional Hebbian neural network to overcome the problem of selecting which one of the activation functions that gives the better results in learning process.

The modified neural network has X vector (X: X1, X2, …, Xm) as an input of plaintext in ASCII form, while the vector (Y:Y1, Y2, …, Ym) is the output of encryption process (ciphertext). The weight (W_{ij}) plays two roles: the first one, it acts as a ciphered key of length (m^2), such that, if the message length is 15 characters, then there are 15*15=225 different weights (key length). The second role, it represents the strength of connection between the (i^{th}) input node and the (j^{th}) output node also represent. Figure 3 shows the MHNN structure.

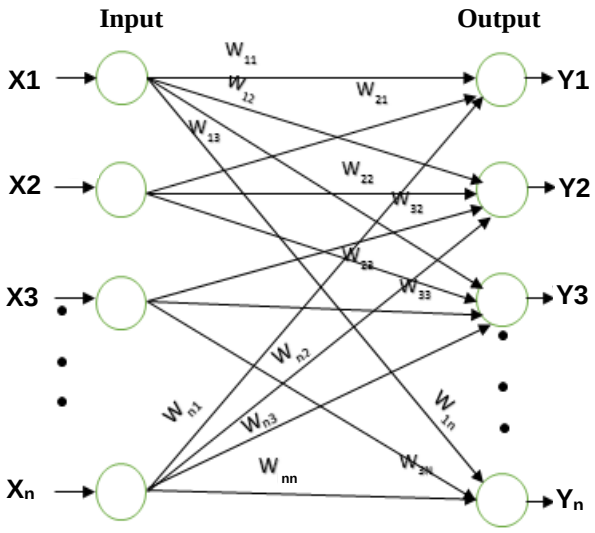

Figure 3. Architecture of MHNN

3. RESEARCH METHOD

The proposal system deals with two sites: sender and receiver, each site includes a main phase all cooperate to achieve the proposal goals of security. These phases are: ciphering phase implemented at sender site.to encrypt the plain text in somehow, so that the encryption text could not read to anyone except the intended person. Another phase is the deciphering implemented at receiver site, its role is an inverse of ciphering phase

3.1. Sender site

At this site, the sender will prepare several stages for the ciphering phase. Each stage has its own function, the stages cooperated between them to result the encryption text before transferred to the receiver site. The stages of ciphering phase are: Pre-processing stage, Ciphering stage, and DNA compression stage as shown in Figure 4.

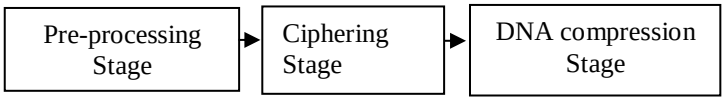

Figure 4. Ciphering phase

3.1.1. Pre-processing stage

This stage includes the following steps: i) prepare an initial (weight, message) and ii) assign each character in plaintext message and initial message to its ASCII code.

3.1.2. Ciphering stage

To cipher the plaintext message using MHNN; two types of Hebbian nets will be used, these nets are the Learner net and Ciphering net. For each net type has a specific function. The learner net function is to prepare an optimal weights to the ciphered net as an input to this net, the following steps will conduct:

a. Learner neural net (LNN)

During the learning process, the following steps will be prepared:

- Input initial message as an input for LNN. The message must have the same length of the original plaintext message.
- Normalize the initial weights to avoid the problem of overflow in the weight values. The normalization process is applied to all initial weights to find optimal weights (ciphered keys), the normalization process implemented by the following formula:

$$NW_{IJ}=\frac{Wij}{\sum_{i=1}^{m} \sum_{j=1}^{m} Wij} \quad (1)$$

b. Ciphering neural net (CNN)

The CNN net is used as a ciphering model for coding the plaintext as an output following the steps:

- Convert the plaintext to ASCII code as one of the inputs to CNN
- The optimal weights (ciphered keys) resulted from LNN will be the initial weights to CNN
- The ciphered message (Y_j) is found by using formula (2):

$$Y_j=\sum_{i=1}^{m}(NWij * Xi)$$
$$j = 1, 2, 3, 4, \ldots, m \quad (2)$$

- The weight (ciphered keys) is calculated by formula (3):

$$\Delta W_{ij}=\zeta * Y_j * [X_i - \sum_{k=1}^{i}(NWij * Yk)]$$
$$i, j == 1, 2, 3, 4, \ldots., m \quad (3)$$

where ζ is learning factor (positive and <1), ΔWij are the updated weight. X_i is plain message (ASCII Code), and Y_j is ciphering message (ASCII Code).

3.1.3. DNA compression stage

The DNA is the abbreviation of deoxyribonucleic acid, it represents the whole life forms. DNA is made of nucleotides and is a type of macromolecule that is biological. There are four groups of bases corresponding to four types of nucleotides, they are adenine (A) and thymine (T) or cytosine (C) and guanine (G). This DNA characterized by massive storage, parallelism, and hard cracking. These traits encouraged researchers to move towards incorporating DNA into scientific research with many different directions that include most of the sciences discipline [10]–[26].

a. DNA binary coding

The four chemical bases that make up DNA sequence A, C, G, and T bases, where biologically A is connected to T, and C is connected to G. These synthesis rules can be modified in binary arithmetic by changing input such as assuming that T is related to C or T is related to G. Using a binary encoding rule to translate a hidden message into DNA rules. For each rule such as (A), the corresponding binary formulas can be 00, 01, 10, or 11. for each DNA base. The encoding of DNA and its random properties make it an ideal candidate for both coding and encoding. As a result, converting DNA into the binary form will result in 4!=24 different encoding methods On DNA bases, logical operations such as addition, subtraction, XOR, AND, OR, and NOT are possible [27]–[29]. To compress/decompress the resulted ciphered text from the previous stage, a DNA computing operation will used such as DNA Addition and DNA Subtraction operations using the following steps:

- Convert the ciphered message to DNA coding using Table 1.
- Compress the converted DNA coding message by apply DNA addition operation using Table 2.
- Decode the compressed DNA message
- Send the decoded message to the receiver

3.2. Receiver site

Retrieving the original plaintext message, the receiver has to decrypt the receiving ciphered message using deciphering phase. The receiver will follow a reverse sender steps with some exceptions such as using a subtraction DNA computing operation as in Table 3 instead of addition to decompress the received message. The deciphering phase consist of the stages bellow in order to get the original plaintext, these stages illustrated in Figure 5.

Table 1. DNA coding

Base Name	Binary Coding
A	00
C	01
G	10
T	11

Table 2. DNA addition [30]

+	T	A	C	G
T	C	G	T	A
A	G	C	A	T
C	T	A	C	G
G	A	T	G	C

Table 3. DNA Subtraction [30]

-	T	A	C	G
T	C	G	T	A
A	A	C	G	T
C	T	A	C	G
G	G	T	A	C

Figure 5. Receiver site phases

3.2.1. DNA decompression stage

- Code the compressed message to DNA using Table 1.
- Decompress the converted DNA message by apply DNA subtraction operation using Table 3.
- Convert the DNA message to binary form using Table 1.
- Convert the binary message to ASCII form.

3.2.2. Decryption stage

- Neural network based decryption module is used to retrieve the original message (plain text).
- To carry out the decryption process, using (4), where the sender and receiver sharing the final optimal weights (ciphered keys):

$$X_j = \sum_{i=1}^{m} ((IW)ij * Yi)$$
$$j=1, 2, 3, 4, ..., m \quad (4)$$

where *Yi* are the input in ASCII form of the i^{th} element in ciphered message, *IWij* is the inverse weight that represents the connection between the i^{th} element in ciphertext and j^{th} element in plaintext. *Xj* is the output of j^{th} element in the plaintext.

Finally, convert each ASCII value to its represented character to retrieve the original message. To illustrate the proposed system thought, the example bellow taken to explain the proposal work steps included the main phases:

A text message "Text Encryption Using Modified Hebbian Neural Net" is to be encrypted using MHNN", with ζ=0.5, and initial text "This text is used as one of initial inputs", as an input to the learner net.

During the cipher phase: Convert the two messages to their ASCII code:

"Text Encryption Using Modify Hebbian Neural Net" is to be encrypted using MHN

84 101 120 116 **32** 69 110 99 114 121 112 116 105 11 110 **32** 85 115 105 110 103 **32** 77 111 100 105 102 121 **32** 72 101 98 98 97 105 110 **32** 78 101 117 114 97 108 **32** 78 101 115.

This text is used as a one of the initial inputs (initial message input to LNN):

84 104 105 115 **32** 116 101 120 116 **32** 105 115 **32** 117 115 101 100 **32** 97 115 **32** 97 **32** 111 110 101 **32** 111 102 **32** 116 104 101 **32** 105 110 105 116 105 97 108 **32** 105 110 112 117 116 115.

The result of ciphering is:

1145 1160 1142 1065 1118 1030 1076 1271 1117 1054 1250 1272 1166 1097 1212 1066 1097 1022 996 1230 1255 1202 1201 1170 1165 1141 1212 1181 1315 1020 1281 1201 1166 1256 1259 1134 1121 1060 1076 1197 1070 1155 1013 1143 1224 1051 1241.

Each number consist of 4 digits, these numbers will be separated in two parts, if the number consist of 3 digits it will completed to 4 digits by adding 0 in left side as a in number 996:

11 45, 11 60, 11 42, 10 65, 11 18, 10 30, 10 76, 12 71, 11 17, 10 54, 12 50, 12 72, 11 66, 10 97, 12 12, 10 66, 10 97, 10 22, **0**9 96, 12 30, 12 55, 12 02, 12 01, 11 70, 11 65, 11 41, 12 12, 11 81, 13 15, 10 20, 12 81, 12 01, 11 66, 12 56, 12 59, 11 34, 11 21, 10 60, 10 76, 11 97, 10 70, 11 55, 10 13, 11 43, 12 24, 10 51, 12 41.

Each part will be converted to a binary form of 8 bits, if one part is results less than 4 bits then embeds it with 0's on left side to be 8 bits:

00001011 **00**101100, **0000**1011 **00**111100, **0000**1011 **00**101010, **0000**1010 **0**1000001, **0000**1011 **000**10010, **0000**1010 **000**11110, **0000**1010 **0**1001100, **0000**1100 **0**1000111, **0000**1011 **000**10001, **0000**1010 **00**110110, **0000**1100 **00**110010, **0000**1100 **0**1001000, **0000**1011 **0**1000010, **0000**1010 **0**1100001, **0000**1100 **0000**1100, **0000**1010 **0**1000010, **0000**1010 **0**1100001, **0000**1010 **000**10110, **0000**1001 **0**1100000, **0000**1100 **000**11110, **0000**1100 **00**110111, **0000**1100 00000010, **0000**1100 00000001, **0000**1011 **0**1000110, **0000**1011 **0**1000001, **0000**1011 **00**101001, **0000**1100 **0000**1100, **0000**1011 **0**1010001, **0000**1101 **0000**1111, **0000**1010 **000**10100, **0000**1100 **0**1010001, **0000**1100 00000001, **0000**1011 **0**1000010, **0000**1100 **00**111000, **0000**1100 **00**111011, **0000**1011 **00**100010, **0000**1011 **000**10101, **0000**1010 **00**111100, **0000**1010 **0**1001100, **0000**1011 **0**1100001, **0000**1010 **0**1000110, **0000**1011 **00**110111, **0000**1010 **0000**1101, **0000**1011 **00**101011, **0000**1100 **000**11000, **0000**1010 **00**110011, **0000**1100 **00**101001.

The binary bits will be coded to DNA bases according to Table 1:

AAGT AGTA, AAGT ATTA, AAGT AGGG, AAGG CAAC, AAGT ACAG, AAGG ACTG, AAGG CATA, AATA CACT, AAGT ACAC, AAGG ATCG, AATA ATAG, AATA CAGA, AAGT CAAG, AAGG CGAC, AATA, AATA, AAGG CAAG, AAGG CGAC, AAGG ACCG, AAGC CGAA, AATA ACTG, AATA ATCT, AATA AAAG, AATA AAAC, AAGT CACG, AAGT CAAC, AAGT AGGC, AATA, AATA, AAGT CCAC, AATC AATT, AACC ACCA, AATA CCAC, AATA AAAC, AAGT CAAG, AATA ATGA, AATA ATGT, AAGT AGAG, AACCT ACCC, AAGG ATTA, AAGG CATA, AAGT CGAC, AAGG CACG, AAGT ATCT, AAGG AATC, AAGT AGGT, AATA ACGA, AAGG ATAT, AATA AGGC.

The compress the message, adding DNA Computing operations are used, using Table 2:

CTTG, CGAG, CTCA, ACTG, CATA, CAAC, ACAT, ACTG, CATT, CGGC, CGGT, ACAC, ACTA, ATTG, CCCC, ACTC, ATTG, CAGC, ATTC, CACC, CGTG, CCGT, CCGA, ACGA, ACTT, CTCT, CCCC, AATT, CCCT, CACA, AAGA, CCGA, ACTA, CGAC, CGAG, CTTA, CACT, CGAT, ACAT, ATTT, ACGC, CGGC, CCAG, CTCC, CAAC, CGTA, CTAA.

Convert the binary bits to decimal and transfer the decimal numbers the receiver:

126, 98, 130, 30, 72, 66, 19, 30, 79, 105, 106, 17, 44, 62, 85, 29, 62, 73, 69, 110, 91, 88, 24, 31, 112, 85, 15, 87, 68, 8, 88, 28, 97, 98, 124, 71, 99, 19, 63, 25, 105, 82, 117, 65, 108, 112.

4. RESULTS AND DISCUSSION

This paper introduced a modified cipher system that use the concepts of neural network and DNA Computing to provide robust security. In this work, modernization of the HNN with no activation function so the secret message will be encrypted by unsupervised neural network method. The results in Table 4 and Figure 6 demonstrate the encryption/decryption processes are performed with a reasonable amount of time in recovering the original message and achieved the goal of cannot break the transmitted ciphertext by the intruder because the ciphertext is ciphered using a very long key.

Table 4. Encryption and decryption time

Characters/Message	Encryption Time (ms)	Decryption Time (ms)
47	0.000011417	0.00001100
400	0.000131223	0.0002112016
1000	6.113423341	0.117343320
8445	10.51000012	0.151867112
125634	13.50230144	0.198740113

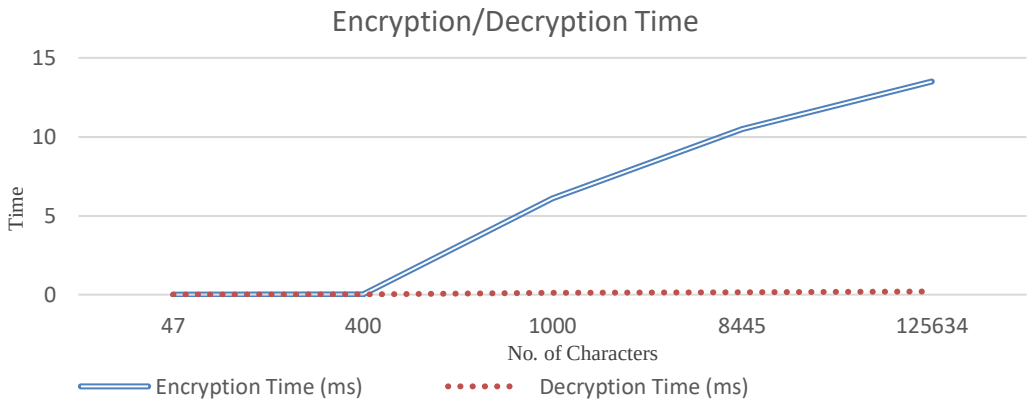

Figure 6. Encryption and decryption time

4.1. DNA reference sequence

There are 163 million DNA reference sequences. The probability of the attacker making a guess is (1/24). The sender not restricted in choosing any binary coding form for nucleotide bases. Hence, A can be (00, 01, 10, 11) or T can be (00, 01, 10, 11) and so on for reset bases. Therefore, the number of binary coding rules are 4×3×2×1=24. So, the chance of the attacker making the right guess is (1/24). As a consequence, the likelihood of an attacker making a correct and accurate guess is ((1/24)*(1/24)*(1/(163*10^6))).

4.1.1. Randomness

A random process is one whose results are not understood. Intuitively, this is why randomness is vital to our work because it offers a way to generate knowledge that an enemy cannot learn or predict. In our work, we get different keys for each iteration. As a consequence, the attacker cannot predict the key.

4.1.2. Data compression

This section explain the result of using DNA compression rate for the amount of time it takes either for encryption or decryption according to the numbers of characters that constitute the message (plain text) as shown in Table 5 compared with research [31].

Table 5. Compression between [31] and the proposal using standard compression algorithms

Data Set	LZFG	HUFF	RLE	SF	AC	Proposal
Bib	2.90	8.16	5.26	5.56	5.23	2.33
Book1	3.62	8.17	4.57	4.83	4.55	2.55
Book2	3.05	8.17	4.83	5.08	4.78	2.66
News	3.44	7.98	5.24	5.41	5.19	2.54
Paper1	3.03	8.12	5.09	5.34	4.98	2.51
Paper2	3.16	8.14	4.68	4.94	4.63	2.65
progc	2.89	8.10	5.33	5.47	5.23	2.20

5. CONCLUSION

It is a challenge to maintain big data of an enormous population and protect this data. In this paper, adopted new encryption method using hybrid techniques such as machine learning and DNA computing operations have an important role in implementing secure environment. Using two modified HNN as a ciphering model. The ciphering process based on the plain message and the number of epochs that based on the used parameters. Hence, determine the actual parameters like initial weight, learner function for encryption and decryption methods is hard for the intruder to guess. On other hand, visibility is the main attribute of the DNA sequences, so finding the hidden message from a DNA sequence is difficult. Although, using DNA computing operations for data compression increasing the confusion image and decrease the transferring time compared with the algorithms shown in Table 5. In the future, one can use the proposed model to transmit multimedia data such as images, audio and videos.

REFERENCES

[1] O. G. Abood and S. K. Guirguis, "DNA computing and its application to information and data security field: a survey," *International Journal of Academic Engineering Research (IJAER)*, vol. 3, no. 1, pp. 1–5, 2019.

[2] A. Gehani, T. LaBean, and J. Reif, "DNA-based cryptography," in *DNA Based Computers V*, American Mathematical Society, 2000, pp. 233–249.

[3] M. Barakat, C. Eder, and T. Hanke, "An introduction to cryptography," *mathematik*, 2018. https://www.mathematik.uni-kl.de/~ederc/download/Cryptography.pdf (accessed Sep. 20, 2018).

[4] A. Yayik and Y. Kutlu, "Neural network based cryptography," *Neural Network World*, vol. 24, no. 2, pp. 177–192, 2014, doi: 10.14311/NNW.2014.24.011.

[5] "Hebb, D. O. The organization of behavior: A neuropsychological theory. New York: John Wiley and Sons, Inc., 1949. 335 p. $4.00," *Science Education*, vol. 34, no. 5, pp. 336–337, Dec. 1950, doi: 10.1002/sce.37303405110.

[6] P. Mishra, "What is Hebbian Learning?," *Data Driven Investor*. 2019.

[7] G. Amato, F. Carrara, F. Falchi, C. Gennaro, and G. Lagani, "Hebbian learning meets deep convolutional neural networks," in *Lecture Notes in Computer Science*, Springer International Publishing, 2019, pp. 324–334.

[8] G. Lagani, G. Amato, F. Falchi, and C. Gennaro, "Training convolutional neural networks with Hebbian principal component analysis," *Computer Vision and Pattern Recognition*, Dec. 2020.

[9] A. Magotra and J. Kim, "Improvement of Heterogeneous transfer learning efficiency by using Hebbian learning principle," *Applied Sciences*, vol. 10, no. 16, p. 5631, Aug. 2020, doi: 10.3390/app10165631.

[10] S. Gangadharan and K. Raman, "The art of molecular computing: whence and whither," *Biomolecules*, Feb. 2021.

[11] M. N. Sadiku, A. E. Shadare, and S. M. Musa, "DNA computing made simple," *Journal of Scientific and Engineering Research*, vol. 3, no. 2, pp. 116–118, 2016.

[12] S. G. Abels and E. F. Khisamutdinov, "Nucleic acid computing and its potential to transform silicon-based technology," *DNA and RNA Nanotechnology*, vol. 2, no. 1, Jan. 2015, doi: 10.1515/rnan-2015-0003.

[13] S. S. Kulkarni, "Implementation of SDES and DES using neural network," *International Journal of Innovations in Engineering and Technology (IJIET)*, vol. 1, no. 2, pp. 34–49, 2012.

[14] S. D. Jagtap, P. BalaRamudu, and M. K. Singh, "Cryptography based on artificial neural network," *International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)*, vol. 4, no. 11, pp. 2785–2789, 2015.

[15] G. S. Mohammed, "Text encryption algorithm based on chaotic neural network and random key generator," *Text Encryption Algorithm Based on Chaotic Neural Network and Random Key Generator*, 2016.

[16] S. S. Roy, S. A. Shahriyar, M. Asaf-Uddowla, K. M. R. Alam, and Y. Morimoto, "A novel encryption model for text messages using delayed chaotic neural network and DNA cryptography," in *2017 20th International Conference of Computer and Information Technology (ICCIT)*, Dec. 2017, pp. 1–6, doi: 10.1109/ICCITECHN.2017.8281796.

[17] P. Dixit, M. C. Trivedi, A. K. Gupta, V. K. Yadav, Vineet, and K. Singh, "Video steganography using concept of DNA sequence

and index compression technique," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 8, no. 5, pp. 408–417, 2019.
[18] S. Basu, M. Karuppiah, M. Nasipuri, A. K. Halder, and N. Radhakrishnan, "Bio-inspired cryptosystem with DNA cryptography and neural networks," *Journal of Systems Architecture*, vol. 94, pp. 24–31, Mar. 2019, doi: 10.1016/j.sysarc.2019.02.005.
[19] S. Namasudra, D. Devi, S. Kadry, R. Sundarasekar, and A. Shanthini, "Towards DNA based data security in the cloud computing environment," *Computer Communications*, vol. 151, pp. 539–547, Feb. 2020, doi: 10.1016/j.comcom.2019.12.041.
[20] H. Li *et al.*, "The specific DNA barcodes based on chloroplast genes for species identification of Orchidaceae plants," *Scientific Reports*, vol. 11, no. 1, p. 1424, Dec. 2021, doi: 10.1038/s41598-021-81087-w.
[21] S. P. Singh and M. E. Naidu, "A Novel method to secure data using DNA sequence and Armstrong Number," *Asian Journal For Convergence In Technology (AJCT)*, vol. 3, no. 3, 2017.
[22] P. Malathi, M. Manoaj, R. Manoj, R. Vaikunth, and R. E. Vinodhini, "Highly improved DNA based steganography," *Procedia Computer Science*, vol. 115, pp. 651–659, 2017, doi: 10.1016/j.procs.2017.09.151.
[23] M. Indrasena Reddy, A. P. Siva Kumar, and K. Subba Reddy, "A secured cryptographic system based on DNA and a hybrid key generation approach," *Biosystems*, vol. 197, Nov. 2020, doi: 10.1016/j.biosystems.2020.104207.
[24] E. Volna, M. Kotyrba, V. Kocian, and M. Janosek, "Cryptography based on neural network," in *ECMS 2012 Proceedings edited by: K. G. Troitzsch, M. Moehring, U. Lotzmann*, May 2012, pp. 386–391, doi: 10.7148/2012-0386-0391.
[25] H. Al-Mahdi, M. Alruily, O. R.Shahin, and K. Alkhaldi, "Design and analysis of DNA encryption and decryption technique based on asymmetric cryptography system," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 2, 2019, doi: 10.14569/IJACSA.2019.0100264.
[26] M. Gao, A. Krissanaprasit, A. Miles, L. C. Hsiao, and T. H. LaBean, "Mechanical and electrical properties of DNA hydrogel-based composites containing self-assembled three-dimensional nanocircuits," *Applied Sciences*, vol. 11, no. 5, Mar. 2021, doi: 10.3390/app11052245.
[27] B. R, D. S, and P. K, "Secure data transmission using DNA sequencing," *IOSR Journal of Computer Engineering*, vol. 16, no. 2, pp. 19–22, 2014, doi: 10.9790/0661-16221922.
[28] E. I.AbdEl-Latif and M. I. Moussa, "Chaotic information-hiding algorithm based on DNA," *International Journal of Computer Applications*, vol. 122, no. 10, pp. 38–42, Jul. 2015, doi: 10.5120/21740-4949.
[29] Adithya B. and Santhi G., "DNA computing using cryptographic and steganographic strategies," in *Data Integrity and Quality*, IntechOpen, 2021.
[30] Z. Ahmad, H. G. Umar, C. Li, and L. Chen, "A DNA-based security solution using aggregated chaos cross and cubic map," *International Arab Journal of Information Technology*, vol. 13, pp. 873–879, 2016.
[31] K. Sailunaz, M. R. A. Kotwal, and M. N. Huda, "Data compression considering text files," *International Journal of Computer Applications*, vol. 90, no. 11, pp. 27–32, Mar. 2014, doi: 10.5120/15765-4456.

BIOGRAPHIES OF AUTHORS

Sahar Adill Kadum received the Ph.D. degrees in computer science from higher institute of informatics and computer science. Currently, she is an Associate Professor at the Department of computer science at Babylon University–collage science for women. Her research interests include information security, bioinformatics security. Image security. She can be contacted at email: dr.sahar.adill@gmail.com, wsci.sahar.adil@uobabylon.edu.iq.

Ali Yakoob Al-Sultan received a PhD in Artificial Intelligent from University of Babylon Currently, he is an lecture at the Department of computer science at Babylon University–collage science for women. His research interests include Artificial Intelligent applications, machine learning, and deep learning. He can be contacted at email: wsci.ali.yakoob@uobabylon.edu.iq.

Najlaa Adnan Hadie received a BSc and high diploma in mathematics and operation research from University of Babylon Currently, she is an MSc student at the Department of Mathematic science at Babylon University. Her research interests in operation research and computation research. She can be contacted at email: hudinajlaa96@gmail.com.