Information Sciences Letters

Volume 8
Issue 1 *Jan. 2019*

Article 2

2019

Self-adaptive DNA-based Steganography Using Neural Networks

Marghny H. Mohammed
Department of Computer Science, Assiut University, Assiut, Egypt, bosina.hussein@gmail.com

Botheina H. Ali
Department of Information System, Assiut University, Assiut, Egypt, bosina.hussein@gmail.com

Ahmed I. Taloba
Department of Information System, Assiut University, Assiut, Egypt, bosina.hussein@gmail.com

Inf. Sci. Lett. **8**, No. 1, 15-23 (2019)

Information Sciences Letters
An International Journal

http://dx.doi.org/10.18576/isl/080102

Self-adaptive DNA-based Steganography Using Neural Networks

Marghny H. Mohammed[1], *Botheina H. Ali*[2,*] *and Ahmed I. Taloba*[2]

[1] Department of Computer Science, Assiut University, Assiut, Egypt
[2] Department of Information System, Assiut University, Assiut, Egypt

Received: 2 Nov. 2018, Revised: 12 Dec. 2018, Accepted: 22 Dec. 2018
Published online: 1 Jan. 2019

Abstract: Steganography is the science of concealing the secret information within a digital cover-object such as the files text, image, video and etc. Recently, the deoxyribonucleic acid (DNA) sequences are used as a cover-object for data hiding. In this work, an effective algorithm called the self-adaptive DNABS (DNA-based steganography) is proposed. This algorithm is applied for data hiding without changing the function or the type of the original DNA protein. It is implemented using a DNA-based steganography and a Neural Network (backpropagation) algorithm to achieve a lower cracking probability than other techniques. The performance of the algorithm is analyzed and tested by measuring four parameters: the embedding capacity, data payload, cracking probability and the bits per nucleotide (bpn).

Keywords: Neural Network, DNA-based steganography, DNA digital coding rule, Cracking Probability.

1 Introduction

Due to the need for transferring all different data types in a safe communication channel via the internet, transmitting the secret data through a covert communication channel becomes the only challenging problem. Information security can be developed using a several approaches including steganography and cryptography.

The process of hiding the existence of the secret data is called steganography. The general model of steganography is defined as the process in which the secret data are embedded inside the digital cover media such as image, video or audio files. Cryptography is the process that uses a certain algorithm, called an encryption algorithm, to convert the secret data (plaintext) to another data (ciphertext) using a device called the secret key in order to change the meaning of the plaintext. The reversing process of cryptography is called decryption [1], [2].

Digital watermarking is the process of embedding a signal, such as a signature or a trademark, into digital media. It is used as an application for steganography. It marks the data of the digital media as a watermark and it is the main branch of Copyright marking [3].

In this work, the stego-object sent via the covert channel is the Neural Network's final weights after embedding the secret data in the DNA sequence.

There are two implementation steps in the proposed algorithm. The first step includes embedding the secret data in a DNA sequence. In the second step, the use of the Neural Network algorithm increases the proposed technique's security.

1.1 DNA Steganography

Biotechnology is increasingly applied on everything in our life. The use of DNA as a cover-object instead of digital media to hide messages becomes more interesting and secure option. Digital media can be deformed and can be noticed by the naked eye. The hiding capacity of other digital media (image, video, or audio) is relatively low while DNA has a high hiding capacity. Accordingly, DNA-based steganography is applied for overcoming the hiding capacity problem [4]. The DNA structure is a dual helix shape which is made up of four nucleotides (A (adenine), C (cytosine), G (guanine), and T (thymine)), as shown in **Fig. 1**. Hence, DNA can be viewed as a sequence of nucleotides: ACGTATATTCACTTTA…etc.

* Corresponding author e-mail: bosina.hussein@gmail.com

The available DNA sequences for public use are about 163 million on different web sites (like the National Center for Biotechnology Information (NCBI gene bank) and the European Bioinformatics Institute (EBI gene bank). The proposed algorithm uses the NCBI DNA bank as shown in the experimental results [4],[5],[6],[7].

Fig. 1: DNA Structure [4]

1.2 Neural Network in Steganography

The Neural Network (NN) is an information learning model. It consists of many layers in its architecture where each layer consists of huge number of neurons connected to solve several problems. The input, the hidden, and the output layer are the three layers that made up this model as shown in **Fig. 2** [3]. The network has been learned to do a particular function by specifying the weights among layers in the network. Similar to the human brain, NN learns by example. It can be used in many applications of steganography. NN can be easily applied on embedding and extracting the data in the steganography system.

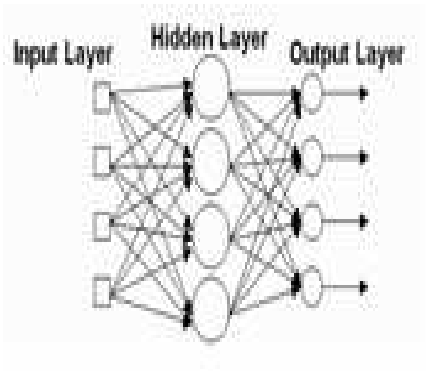

Fig. 2: The Neural Network Architecture Model [8]

The rest of this paper is divided into eight sections. Section 2, introduces the motivation & objectives of this work. Section 3, presents the problem formulation of the work. Section 4, presents the background related to this work. Section 5, explains the proposed algorithm. Section 6, analysis the performance of the proposed algorithm. The experimental results are presented in section 7. Finally, section 8 summarizes the conclusion.

2 Motivations & Objectives

The main motivation of DNA-based steganography is to increase the hiding capacity than other digital media. Due to the ability of DNA to store huge amounts of data, it is used as a cover-object for data hiding. DNA, because of its huge data hiding capacity and its high redundancy and randomness, is used now in most steganography applications. It has many characteristics which make it an excellent steganography medium.

The general objective of this work is to enhance the DNA-based steganography scheme, considering a high data hiding capacity, a zero payload, a low cracking probability, a high robustness, a low execution time, and a high security. This paper also aims at preserving the DNA sequence's function and structure. So, the system should be strong against the unauthorized access.

3 Problem Formulation

Steganography using DNA is safer than the regular cryptography in transferring secret messages. A huge number of DNA strands is in one single microdot. It is very difficult to recognize the strand containing the hidden message without the reference sequences. Hiding the reference DNA sequence is one of the most critical considerations if someone wants to solve the problem of security. However, DNA steganography could be decoded; measures should be considered to surmount the security deficiency. In this work, the NN as a learning model is implemented by the DNA sequence as an input and the locations of the embedded secret message as an output as shown **Fig. 3**. The network model is trained using a set of datasets of reference DNA sequences as inputs. A set of Stego-DNA sequences as outputs and a set of values as initial weights are used for training to generate the final weights. After training the model, the generated weights are used as a ciphertext. The suggested algorithm results in the highest embedding capacity and the lowest cracking probability as shown in results.

4 Background

The following studies are some of the most recent related works.

Inf. Sci. Lett. **8**, No. 1, 15-23 (2019) / www.naturalspublishing.com/Journals.asp

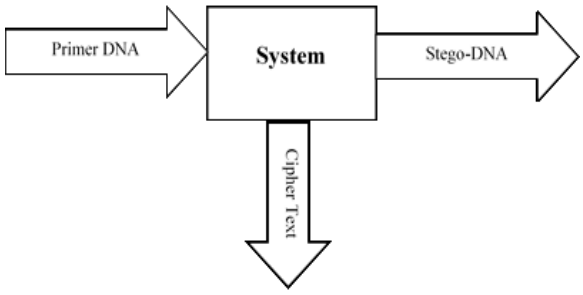

Fig. 3: The System with the Input and the Output.

Shiu et al. [9] proposed three methods of hiding data using DNA-based steganography. These methods are: the insertion based, the complementary pair based, and the substitution based method. In the insertion based method, the DNA sequence and the secret data are divided into a number of blocks after converting them to binary. At the beginning of each block of the reference sequence, one bit of the secret data is inserted. In the complementary pair based method, the secret data is divided into a number of segments in order to select the longest complementary pairs. Then, at the beginning of each complementary pair, one segment of the secret data is inserted. In the substitution-based method, each reference DNA nucleotide base is substituted with one bit of the secret data according to the particular complementary rules: (AC), (CG), (GT) or (TA). The insertion method achieves the highest performance and the lowest cracking probability.

Abbasy et al. [10] proposed a data hiding technique by converting the secret data to a DNA sequence and then hiding it in a DNA sequence by selecting the locations of each two complementary pairs. The result of this idea is the locations list of the hidden data. The main advantage of this algorithm is that the data is hidden without increasing the size of the DNA sequence and no alterations. However, it is an un-blind technique and if the size of the embedded data increases, the obtained locations list increases also.

Khalifa and Atito [11] proposed a powerful embedding capacity algorithm using DNA-based steganography. The algorithm is implemented in two steps. In the first step, the secret data are encrypted by the Playfair cipher based on DNA. In the second step, the encrypted data are embedded in a DNA sequence based on the substitution method using the complementary pair using two-by-two rules.

Guo et al. [12] proposed a new algorithm for DNA-based steganography. The secret data are hidden by substituting the repeated characters in a DNA sequence using the mapping between the complementary pairs rule and two bits of the secret data.

Taur et al. [13] proposed a data hiding algorithm in which the secret data is embedded in a DNA using the substitution based method where each nucleotide base is substituted by another base based on two bits of the secret data in a lookup table. With this algorithm, the increase of the embedding capacity is due to hiding two secret bits in a nucleotide base instead of one bit, but it doesn't keep the functionality of the DNA sequence after hiding the data.

Chakraborty and Bandyopadhyay [14] proposed an algorithm for embedding data in an image and then embedding the resultant image in a DNA sequence using the DNA complementary rule and the magic number sequence(as indexes of complementary rule bases).

Mitras and Abo [15] proposed an algorithm for hiding the secret data in a DNA sequence using the complementary rule of DNA after the encryption of secret data using an encryption algorithm called RSA.

Khalifa [5] proposed a new hybrid system using DNA-based steganography and cryptography. This system is mainly divided into two steps. In the first step, the secret data are encrypted by a secret key. In the second step, the secret key is embedded in the DNA by replacing every codon's LSB (least significant bit) into either purine nucleotide (A & G) or pyrimidine nucleotide (T & C).

Saranya et al. [16] proposed a new hybrid algorithm using DNA-based steganography and cryptography. A Genetic algorithm (GA) is applied to select the best reference DNA sequence for hiding the data. In the first step, a number of reference DNA sequences were generated using the chaotic function and the DNA encoding rules. In the second step, the implementation of GA is done to obtain the best reference DNA sequence.

Hamed et al. [22] proposed a data embedding algorithm. Through the first step of this algorithm, the secret data are converted into DNA sequence using a binary coding rule called the generic N-bits. Through the second step of this algorithm, the resultant DNA sequence is encrypted using the RSA algorithm and embedded in a DNA sequence at random locations generated by a random real number obtained from the atmospheric noise.

Khalifa and Hamad [17] succeeded in using the silent mutations in embedding the secret data inside a DNA sequence (DNA mutations do not change the function of the DNA) by replacing each codon's LSB into either purine nucleotide (A & G) or pyrimidine nucleotide (T & C). In the process of retrieving the data, the reference DNA sequence is not used. Accordingly, their method is considered as a blind algorithm.

Marwan et al. [18] proposed a secured technique for hiding the data by encrypting them. The encrypted data is hidden in a DNA sequence using the substitution based method [18].

Vijayakumar et al. [19] proposed a powerful algorithm for hiding an image after converting it to a DNA sequence inside another image using LSB technique.

5 The Proposed Algorithm

The proposed algorithm has been implemented in three steps. In the first step, the secret message is converted to a

DNA sequence by the digital DNA coding rule. In the second step, the resultant DNA sequence is hidden in a DNA sequence where the sender has to keep it secret with the receiver. Each base from the resultant DNA sequence is hidden in its position in the DNA sequence and extracts all positions as a position list. In the third step, a BackPropagation algorithm as a Neural Networks model is used whereas the DNA sequence is assigned as the input and the position list is the target of the network. After accomplishing the training of the network, the network's final weights are considered as a ciphertext which is sent to the receiver. The summarization of the proposed algorithm is explained in **Fig. 4**.

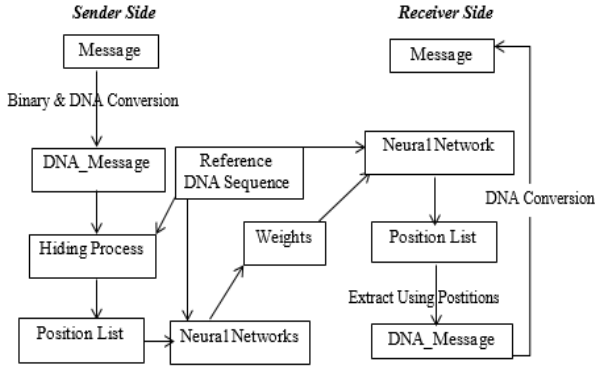

Fig. 4: The Proposed Algorithm.

5.1 The Embedding Process

The embedding process is illustrated using the following steps:
Input: the secret message and the reference DNA sequence.
Output: the weights set of the NN algorithm.
Step 1: select a reference DNA sequence from the NCBI database for embedding the secret message.
Step 2: convert the secret message to its equivalent value (8–bit binary value for each character) using the ASCII conversion then, the obtained binary values are converted to DNA sequence by mapping the resultant sequence to a DNA sequence using the encoding rule of DNA shown in table 1.
Step 3: hide the resultant DNA message in their positions in the DNA sequence and store them in a position list that refers to the DNA message locations.
Step 4: convert the position list values to their equivalent binary values (8, 16, 32 or 64 bits) while the number of bits depending on the secret message's size.
Step 5: use The NN (Back propagation) algorithm for training as shown in **Fig. 5** where the reference DNA sequence is the input to the NN and the binary position list is the target. Then, the NN is trained to reach the output that is close to the target. After accomplishing the training, the neural network's final weights will be transferred to the receiver.

Table 1: The DNA Digital Coding Rule [5] [24].

DNA Base	Decimal	Binary
A	0	00
C	1	01
G	2	10
T	3	11

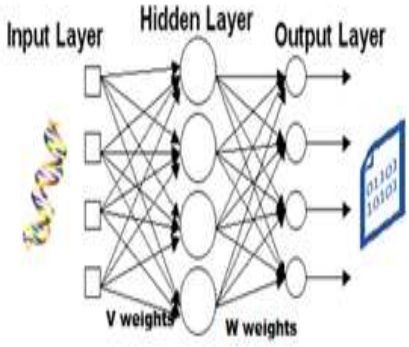

Fig. 5: The Neural Networks Model.

5.1.1 Example for the Embedding Process

Step 1: take in mind that: **reference DNA sequence =** "ACGTACCAACACAGTT", **secret message =** "FCI".
Step 2: convert the ASCII of each character in the secret message to the binary value:

F 70 01000110
C 67 01000011
I 73 01001001

binary message = 01000110 01000011 01001001.
Step 3: convert the binary message to DNA message using the DNA encoding:
DNA message = "CACGCAATCAGC".
Step 4: embed the resultant DNA message in inverse ordering in the reference DNA sequence and store their positions:
Position list = [5 4 7 3 10 8 9 1 11 12 14 15].
Step 5: convert the position list and the reference DNA sequence into their binary equivalent as follows:

Inf. Sci. Lett. **8**, No. 1, 15-23 (2019) / www.naturalspublishing.com/Journals.asp

Binary positions list = [00000101 00000100 00000111 00000011 00001010 00001000 00001001 00000001 00001011 00001100 00001110 00001111].
Binary DNA sequence = [00 01 10 11 00 01 01 00 00 01 00 01 00 10 11 11].
Step 6: train the network using BackPropagation algorithm by assigning the input with the binary DNA sequence and the target with the binary position list. After accomplishing the training, the final weights (the two matrices **V** and **W**) are transferred to the receiver.

5.2 The Extraction Process

This process is achieved by inversing the steps of embedding algorithm:
Step 1: assign the selected reference DNA sequence as input to the network and the given weights layers' neurons in the network. Accordingly, the binary positions list is obtained.
Step 2: convert the binary position list to its decimal values (for each 8–bit).
Step 3: extract the stockticker DNA message from the reference stockticker DNA sequence using the resultant position list.
Step 4: convert the resultant DNA message to its binary equivalent.
Step 5: convert the binary DNA message (for each 8–bit) to its ASCII values and consequently the secret message can be obtained.

5.2.1 Example of the Extraction Process

Step 1: assign the received two matrices as the network's weights **V** (between the input and the hidden layer) and **W** (between the hidden and the output layer) and the binary reference DNA sequence as the input. After accomplishing training the network, the final output can be acquired.
Step 2: round the obtained output to get the binary position list.
Step 3: convert the binary positions list to its decimal values:
Position list = [5 4 7 3 10 8 9 1 11 12 14 15].
Step 4: extract the DNA message using the resultant position list:
DNA message = "CACGCAATCAGC".
Step 5: convert the DNA message to its binary values.
Binary DNA Message = [00 01 10 11 00 01 01 00 00 01 00 01 00 10 11 11].
Step 6: convert the DNA message to its ASCII values to retrieve the secret message which is **"FCI"**.

6 Performance Analysis

The security of any DNA-based steganography system can be measured and analyzed using the parameters: the cracking probability, the embedding capacity, the payload and the bpn [4].

6.1 Cracking Probability

The cracking probability is defined as the total probability of predicting the hidden message. it is also the probability of enemy's success in detecting and retrieving the hidden message for any steganography system. It can be calculated by the totality of some factors in DNA-based steganography algorithm [4, 20].

The cracking probability of the proposed algorithm is calculated using the following factors:

Factor 1: the data hiding direction: the sender must agree with the receiver on the direction of data hiding. Without this direction, the receiver cannot retrieve the correct secret message. There are two directions: the forward and the backward direction so, the probability to predict the direction of hiding data in reference DNA sequence is:

$$\frac{1}{2}. \tag{1}$$

Factor 2: the used DNA sequence: the obtainable number of DNA sequences on the public databases specifically NCBI is nearly 163 million possible reference DNA sequences, so the probability of this factor is:

$$\frac{1}{1.63 \times 10^8}. \tag{2}$$

Factor 3: the digital DNA coding rule: digital coding rules for A, C, G, T are: 00, 01, 10, 11, where A can be mapped to 00, 01, 10, 11. If A is mapped to 00 then, C can have the remaining three possibilities rules which are 01, 10 or 1, where G can have the remaining two possibilities rules and T can have the remaining one. Then, the totality of digital coding rule is 4*3*2*1, so the probability for this factor is:

$$\frac{1}{24}. \tag{3}$$

Factor 4: the network's final weights (the two matrices). The size of the matrix V is $m \times n$ and the size of the matrix W is $n \times p$. So, the possibilities of the two matrices are: $m \times n \times R$ and $n \times p \times R$, respectively; where R is the set of all real numbers. Therefore, their probabilities are $\frac{1}{m \times n \times R}$ and $\frac{1}{n \times p \times R}$. The probability of predicting the two matrices together is:

$$\frac{1}{m \times n \times R} \times \frac{1}{n \times p \times R}. \tag{4}$$

Therefore, the cracking probability of the proposed algorithm is:

$$\frac{1}{2} \times \frac{1}{1.63 \times 10^8} \times \frac{1}{24} \times \frac{1}{m \times n^2 \times p \times R^2}. \tag{5}$$

The cracking probability for other existing techniques is larger than the cracking probability of the proposed algorithm as shown in table 2. As the cracking probability close to zero, it means that the probability of retrieving the hidden message is very low. This algorithm provides high security for the secret message.

Table 2: The Cracking Probability of the Existing Techniques and the Proposed Algorithm.

Method	Cracking Probability
Insertion based method [9]	$\frac{1}{1.63\times10^8}\times\frac{1}{24}\times\frac{1}{(n-1)}\times\frac{1}{(2^m-1)}\times\frac{1}{2^{c-1}}$
Complementary based method [9], [15]	$\frac{1}{1.63\times10^8}\times\frac{1}{24^2}$ *or* $\frac{1}{3^n}$
Insertion based method [4]	$\frac{1}{1.63\times10^8}\times\frac{1}{24}\times\frac{1}{(n-1)}\times\frac{1}{(2^m-1)}\times\frac{1}{2^{c-1}}\times\frac{1}{2^{bm}}$
Substitution based method [9], [21]	$\frac{1}{1.63\times10^8}\times\frac{1}{6}$
Substitution based Complementary method [11]	$\frac{1}{1.63\times10^8}\times\frac{1}{16}\times\frac{1}{24}$
Substitution based Complementary method [12]	$\frac{1}{1.63\times10^8}\times\frac{1}{6}\times\frac{1}{24}$
Substitution based method [22]	$\frac{1}{1.63\times10^8}\times\frac{1}{16!}\times\frac{1}{4}$
Insertion based method [23]	$(\frac{1}{1.63\times10^8})^2\times\frac{1}{24^2}\times p_{64}^{64}$
Generic Complementary Base Substitution [24]	$\frac{1}{(2^{\lvert S\rvert}-1)^2}\times\frac{1}{6}\times\frac{1}{24}$
The Proposed algorithm	$\frac{1}{2}\times\frac{1}{1.63\times10^8}\times\frac{1}{24}\times\frac{1}{m\times n^2\times p\times R^2}$

6.2 Capacity

Capacity is the total amount of secret data that can be embedded in a DNA sequence (**S**) [9]. The embedding capacity of the proposed algorithm = $|S|$, where $|S|$ is the length of **S** after hiding the secret data.

6.3 Payload

Payload is the increasing in the length of the DNA sequence [9]. The proposed algorithm's payload is 0.

6.4 BPN

BPN is the number of the secret bits that can be embedded inside the reference DNA sequence in each nucleotide [9]. The proposed algorithm's bpn is 2.

A comparison between the existing DNA data hiding techniques and the technique of the proposed algorithm is shown in table 3.

Table 3: A Comparison between the Existing DNA Data Hiding Techniques and the technique of the Proposed Algorithm.

DNA Data Hiding Method	Capacity	Payload	BPN
Insertion	$\lvert S\rvert+\frac{\lvert M\rvert}{2}$	$\frac{\lvert M\rvert}{2}$	$\frac{\lvert M\rvert}{\lvert S\rvert+\frac{\lvert M\rvert}{2}}$
Complementary pair	$\lvert S\rvert+\lvert M\rvert(K+3\frac{1}{2})$	$\lvert M\rvert(K+3\frac{1}{2})$	$\frac{\lvert M\rvert}{\lvert S\rvert+\lvert M\rvert(K+3\frac{1}{2})}$
Substitution	$\lvert S\rvert$	0	$\frac{\lvert M\rvert}{\lvert S\rvert}$
The Proposed algorithm	$\lvert S\rvert$	0	2

7 The Experimental Results

MATLAB R2015 framework has been used in implementing the proposed algorithm. Various sizes of secret messages have been used with different reference DNA sequences [25]. The implementation is done using the backpropagation algorithm. The total number of the NN's layers is three: the input, the hidden with 2 nodes, and the output. The assigned error rate for the network is 0.01. The network takes the reference DNA sequence after converting it to binary using the DNA digital encoding rule as an input. The positions of the DNA message after converting them to binary (8–bit or greater because the position value may be greater than 2∧15; we face this problem in testing so, in many cases, we convert to 16–bit binary for each byte) are used as a target. The NN is trained first by random weights then, updating to minimize the error between the target and the output from the NN until it reached 0.01. Training the NN takes long time to generate the final weights.

In the extraction process, the NN's weights are the two matrices (weights) and the target is the reference DNA sequence. After training the NN, the positions of the DNA message will be obtained. After that, the DNA message will be extracted and then, converted to binary and to its equivalent ASCII values. And finally, the secret message is extracted.

Although this technique provides higher security, it takes long execution time. The training process to generate the weights consumes much time of implementation. These weights act as a cipher text (see **Fig. 6, 7**) which is required for retrieving the secret message. The mapping between the long length reference DNA and the target of neural network takes very long time. Table 4 shows the performance parameters of hiding secret messages inside different reference DNA sequences.

In tables 5, 6, 7, 8 and 9 show the result of embedding different sizes of secret messages in different DNA sequences.

Table 4: The performance of embedding a 10KB secret message in eight DNA sequences.

Sequence	Number of DNA Bases	Capacity	Payload	BPN
AC153526	200117	200117	0	2
AC166252	149884	149884	0	2
AC167221	204841	204841	0	2
AC168874	206488	206488	0	2
AC168897	200203	200203	0	2
AC168901	191456	191456	0	2
AC168907	194226	194226	0	2
AC168908	218028	218028	0	2

Inf. Sci. Lett. **8**, No. 1, 15-23 (2019) / www.naturalspublishing.com/Journals.asp

0.708332774910562460	0.415814517396961290
0.508859416759238180	0.959551017585233090
0.131295348725484760	0.975344346318841590
0.049915067654274337	0.412589066584703730
0.045229507765932708	0.912733971275537010
0.698536307775319830	0.913809615268872280
0.358082758572417780	0.681637498601099350
0.768786893431183450	0.481235295561402540
0.884450445117748060	0.437134178115053840
0.612282745300774980	0.943037933116204760
0.983492898199001100	0.774221580802839870
0.760322199977203960	0.022958678474423566
0.545117635273021930	0.555431471911249750
0.755691100222127600	0.015286464097268193
0.211960224441027470	0.101100899527702630
0.797918834574476880	0.975693143220841110
0.999022189068106910	0.835367565417884820
0.346489347215008460	0.275867252400328170
0.153805727447376640	0.371630906947853610
0.361206009505408930	0.608731469967516150
0.842835531110129210	0.688421126991560550
0.615935683660779000	0.173168808639698300
0.405190424330047930	0.315030680688586310
0.004077608457992654	0.493189851456978130
0.608087164415097580	0.028058140110706509
0.223310279522402010	0.566434475063857470

Fig. 6: The last 26 rows of the V matrix are presented as a result of hiding 10KB secret message inside the reference DNA sequence "AC167221" using the neural network algorithm.

```
Columns 783889 through 783900

-2.5199   -2.2473    2.8408   -2.5443   -2.7379   -2.6755    2.3714   -2.4079   -2.4565   -2.2308   -2.2753    2.7769
-2.6508   -2.9233    2.3304   -2.6262   -2.4329   -2.4951    2.7998   -2.7628   -2.7140   -2.9398   -2.8955    2.3948

Columns 783901 through 783912

 2.4335    2.6768   -2.6525   -2.6300   -2.7497    2.5704    2.9578   -2.7819   -2.4233   -2.3380    2.5982   -2.8067
 2.7378    2.4954   -2.5182   -2.5408   -2.4211    2.6007    2.2134   -2.3887   -2.7473   -2.8327    2.5733   -2.3689

Columns 783913 through 783924

 2.5655    2.5304    2.7178   -2.7802    2.9617   -2.8142    2. 2102   2.3460    2.755   -2.2347   -2.5902   -2.2293
 2.6064    2.6414    2.4537   -2.3905    2.2094   -2.3566    2.9612    2.8253    2.4663   -2.9359   -2.5806   -2.9415

Columns 783925 through 783936

 2.4005    2.4186   -2.1756   -2.5682   -2.8924    2.7133   -2.1919    2.6585    2.7759    2.5142   -2.6747    2.6724
 2.7712    2.7529   -2.9950   -2.6023   -2.2782    2.4577   -2.9788    2.5132    2.3951    2.6573   -2.4959    2.4991

Columns 783937 through 783940

 2.4898    2.5059   -2.7730    2.7024
 2.6821    2.6662   -2.3975    2.4695
```

Fig. 7: the last 51 columns of the W matrix are presented as a result of hiding 10 KB secret message inside the reference DNA sequence "AC167221" using the neural network algorithm.

Table 5: The results of embedding a 2K byte secret message in eight DNA sequences with the number of output neurons=131136 and the number of epochs=2597.

Sequence	Number of DNA Bases	Number of Input Neurons
AC153526	200117	400234
AC166252	149884	299768
AC167221	204841	409682
AC168874	206488	412976
AC168897	200203	400406
AC168901	191456	382912
AC168907	194226	388452
AC168908	218028	436056

Table 6: The results of embedding a 3.07K byte secret message in eight DNA sequences with the number of output neurons=239400 and the number of epochs=3944.

Sequence	Number of DNA Bases	Number of Input Neurons
AC153526	200117	400234
AC166252	149884	299768
AC167221	204841	409682
AC168874	206488	412976
AC168897	200203	400406
AC168901	191456	382912
AC168907	194226	388452
AC168908	218028	436056

Table 7: The results of embedding a 10K byte secret message in eight DNA sequences with the number of output neurons=783940 and the number of epochs=80207.

Sequence	Number of DNA Bases	Number of Input Neurons
AC153526	200117	400234
AC166252	149884	299768
AC167221	204841	409682
AC168874	206488	412976
AC168897	200203	400406
AC168901	191456	382912
AC168907	194226	388452
AC168908	218028	436056

Table 8: The results of embedding a 20K byte secret message in eight DNA sequences with the number of output neurons=1563168 and the number of epochs=158862.

Sequence	Number of DNA Bases	Number of Input Neurons
AC153526	200117	400234
AC166252	149884	299768
AC167221	204841	409682
AC168874	206488	412976
AC168897	200203	400406
AC168901	191456	382912
AC168907	194226	388452
AC168908	218028	436056

Table 9: The results of embedding a 30K byte secret message in eight DNA sequences with the number of output neurons=2346044 and the number of epochs=237714.

Sequence	Number of DNA Bases	Number of Input Neurons
AC153526	200117	400234
AC166252	149884	299768
AC167221	204841	409682
AC168874	206488	412976
AC168897	200203	400406
AC168901	191456	382912
AC168907	194226	388452
AC168908	218028	436056

8 Conclusion

In this work, a powerful DNA-based steganography system is developed. According to the proposed algorithm, the secret messages are converted to DNA sequences using the DNA digital coding rule. Then, these DNA messages are embedded in DNA sequences with the opposite direction (from the end to the start) of the DNA sequence. After that, their positions are taken as the NN's target and the DNA sequence as the network's input.

Then, apply the training to generate the final weights which will be sent to the receiver to retrieve the secret message. The proposed algorithm doesn't disturb or change the function of the DNA sequence and it achieves higher capacity, zero payload, 2 bpn and lower cracking probability which are required. The proposed algorithm is not a blind one because the DNA sequence is obligatory in the extraction process. The only challenge of our technique is the long time for training the NN algorithm.

References

[1] M.M.Sadek, A.S. Khalifa, and M.G. Mostafa, Video steganography: a comprehensive review. Multimedia tools and applications**74**, 7063-7094 (2015).

[2] V. Nagaraj, V. Vijayalakshmi, and G. Zayaraz, Overview of digital steganography methods and its applications. Int. J. Adv. Sci. Technol**60**, 45-58 (2013).

[3] S. Husien and H. Badi, Artificial neural network for steganography. Neural Computing and Applications **26**, 111-116 (2015).

[4] P.Malathi, M. Manoaj, R. Manoj, V.Raghavan, and R. E. Vinodhini, Highly Improved DNA Based Steganography. Procedia Computer Science **115**, 651-659(2017).

[5] A. Khalifa, LSBase: A key encapsulation scheme to improve hybrid crypto-systems using DNA steganography. in Computer Engineering & Systems (ICCES), 8th International Conference on. 2013. IEEE **1**05-110 (2013).

[6] M.R.N. Torkaman, N.S. Kazazi, and A. Rouddini, Innovative approach to improve hybrid cryptography by using DNA steganography. International Journal of New Computer Architectures and their Applications (IJNCAA), **2**, 224-235 (2012).

[7] S. Namasudra, P. Roy, P. Vijayakumar, S. Audithan, and B. Balusamy, Time efficient secure DNA based access control model for cloud computing environment. Future Generation Computer Systems **73**, 90-105 (2017).

[8] R. Khare, R. Mishra, and I. Arya. Video Steganography Using LSB Technique by Neural Network. in Computational Intelligence and Communication Networks (CICN), International Conference on. 2014. IEEE **8**98-902 (2014).

[9] H. J. Shiu, K. L. Ng, J. F. Fang, R. C. Lee and C. H. Huang, Data hiding methods based upon DNA sequences. Information Sciences **180**, 2196-2208 (2010).

[10] M. R. Abbasy, P. Nikfard, A. Ordi, and M. R. N. Torkaman, DNA base data hiding algorithm. International Journal of New Computer Architectures and their Applications (IJNCAA) **2**, 183-192 (2012).

[11] A. Khalifa and A. Atito. High-capacity DNA-based steganography. in Informatics and Systems (INFOS), 8th International Conference on. 2012. IEEE **7**6-80 (2012).

[12] C. Guo, C.-C. Chang and Z.-H. Wang, A new data hiding scheme based on DNA sequence. Int. J. Innov. Comput. Inf. Control **8**, 139-149 (2012).

[13] J. S. Taur, H. Y. Lin, H. L. Lee and C. W. Tao, Data hiding in DNA sequences based on table lookup substitution. International Journal of Innovative Computing, Information and Control **8**, 6585-6598 (2012).

[14] S. Chakraborty and S.K. Bandyopadhyay, Two stages data-image steganography using dna sequence. International Journal of Engineering Research and Development**2**, 69-72 (2012).

[15] Mitras, B.A. and A. Abo, Proposed steganography approach using DNA properties. International Journal of Information Technology and Business Management**14**, 96-102 (2013).

[16] M. Saranya, A.K. Mohan, and K. Anusudha. A composite image cipher using DNA sequence and genetic algorithm. in Contemporary Computing and Informatics (IC3I), International Conference on. 2014. IEEE **1**022-1026 (2014).

[17] A. Khalifa and S. Hamad, Hiding Secret Information in DNA Sequences Using Silent Mutations. British Journal of Mathematics & Computer Science **11**, 1-11 (2015).

[18] S. Marwan, A. Shawish, and K. Nagaty. An Enhanced DNA-based Steganography Technique with a Higher Hiding Capacity. in Bioinformatics.**1**50-157 (2015).

[19] P. Vijayakumar, V. Vijayalakshmi, and G. Zayaraz, An improved level of security for dna steganography using hyperelliptic curve cryptography. Wireless Personal Communications **25**, 967-973 (1980).

[20] G. Hamed, M. Marey, S. El-Sayed and F. Tolba, DNA based steganography: survey and analysis for parameters optimization, in Applications of intelligent optimization in biology and medicine **4**7-89 (2016).

[21] M. R. N. Torkaman, P. Nikfard, N. S. Kazazi, M. R. Abbasy and S. F. Tabatabaiee, Improving hybrid cryptosystems with DNA steganography, in Digital Enterprise and Information Systems**4**2-52 (2011).

[22] G. Hamed, M. Marey, S. A. El-Sayed and M. F. Tolba, Hybrid technique for steganography-based on DNA with n-bits binary coding rule. in Soft Computing and Pattern Recognition (SoCPaR), 7th International Conference of. 2015. IEEE **9**5-102 (2015).

[23] F.E. Ibrahim, H. Abdalkader, and M. Moussa, Enhancing the Security of Data Hiding Using Double DNA Sequences. in Industry Academia Collaboration Conference (IAC)**6**-8 (2015).

[24] A. Khalifa, A. Elhadad, and S. Hamad, Secure blind data hiding into pseudo DNA sequences using playfair ciphering and generic complementary substitution. Appl. Math. Inf. Sci**49**, 1221-1242 (2016).

[25] Information, N.C.f.B.; Available from: https://www.ncbi.nlm.nih.gov/.

Marghany Hassan Mohamed Mohamed is a Professor of Computer Science at Faculty of Computers and Information, Assiut University. He occupied the position of Vice Dean for Education and Student Affairs from 3-1-2012 until now at Faculty of Computers and Information, Assiut University. He received the PhD degree in Computer Science from Kyushu University, Japan.

Inf. Sci. Lett. **8**, No. 1, 15-23 (2019) / www.naturalspublishing.com/Journals.asp

Botheina Hussein Ali is a master student at Faculty of Computers and Information, Assiut University. She studies information system at Faculty of Computers and Information and graduated from it in 2012.

Ahmed Ibrahim Taloba Mohamed is a Lecturer of Information Systems at Faculty of Computers and Information, Assiut University. He occupied the position of Director of Quality Assurance Unit at Faculty of Computers and Information, Assiut University. He received the PhD degree in 2015 from Faculty of Computers and Information, Assiut University.