International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Intelligent Secure System

Noor Dhia Kadhm Al-Shakarchy

Computer Science Department, College of Science, Karbala University, Karbala, Iraq

Abstract: ***Artificial intelligent is the science and study how the computer can perform the activities that the human bring can do. The outputs of AI system are either reports or actions, these outputs depend on the information and knowledge that are stored in the computer which represent the problem that the system wants to solve it. The Artificial Intelligent is used in the cryptography in many application such as cryptanalysis of some kind of ciphering methods for example the cryptanalyzing of the convention cipher systems the simple substation cipher systems depend on the cipher text only attack. In this research we construct the encryption and decryption algorithms depending on the artificial intelligent concept such that computational Linguistics . These algorithms convert the original (plaintext) message to ciphertext message by generation natural language. That's mean, the ciphertext message stay natural language also. We constructed four algorithm depending on the degree of security, each one presented kind of security by selecting the method to generating the text. These algorithm gives good randomness and be able to hide the properties of natural language, by using inner keys. These keys changed dynamically during text generation to produced ciphertext. These algorithms gives the safety to ciphertext from attack, that's because, the output text is natural and prevent the intruder curiosity to cryptanalysis ciphertext. In addition to these algorithm (except first one; base text generation algorithm) roubest against the statistical cryptanalysis.***

Keywords: Encryption; Decryption, plaintext; ciphertext, secure system.

1. Introduction

Artificial intelligent is the science and study how the computer can perform the activities that the human bring can do. The outputs of AI system are either reports or actions; these outputs depend on the information and knowledge that are stored in the computer which represents the problem that the system wants to solve it [1]. The Artificial Intelligent is used in the cryptography in many application such as cryptanalysis of some kind of ciphering methods for example the cryptanalyzing of the convention cipher systems the simple substation cipher systems depend on the cipher text only attack.

The heart of an Expert System ES is its croups of knowledge that accumulates during system building, the knowledge is explicit and organized to simplify decision making. Acknowledge base contains general problem solving knowledge as well as specific problem domain knowledge. General knowledge (or Meta knowledge) is about how to solve problems, or knowledge about how to interact with the user, and it is mostly build into the way the inference engine operates [1, 2].

Specific domain knowledge consists of the symbolic descriptions that characterize the definitional and empirical relationships in the domain, and heuristics (called rules of thumb) that limit the search for solution. We can think of the knowledge base in terms of a mapping between the objects and relations in a problem domain and the computational objects and relations in a program [8].

This proposed system employed the concept of Natural Language Processing NLP, tries to make the computer able to understand commands written in standard human languages, also called Computational Linguistics; in cipher system. When the output of any original cipher system to be un-understanding message, which can be make the suspicion to any one indeed the intruder this is ciphered message. The output of the proposed system is natural language in which to go away the suspicion. The proposed intelligent secure system deals with two parts. First part the knowledge base of the system whom used to generate the natural sentence as output. The second part the properties of cipher system and how can provide the confusion and diffusion to the system which can hide the natural properties of the language. The proposed system tested and evaluated using many criterions and obtained good results.

2. Aim of Research

Encryption and decryption algorithms are presented in this research based on the concept of artificial intelligent text generation. The encryption algorithm provides a secure manner by prevent doubt that the text is encrypted, good randomness, the **privacy** which cannot anyone except the exact receiver reads the message, **data integrity** which guarantee that no change and manipulation in data during transmission process, **Authentication** which provides the Verification of the person that you want to read the sent message and finally Non repudiation which makes the person whose message sent to him Unable to denial that he's the right person the message sent to him.

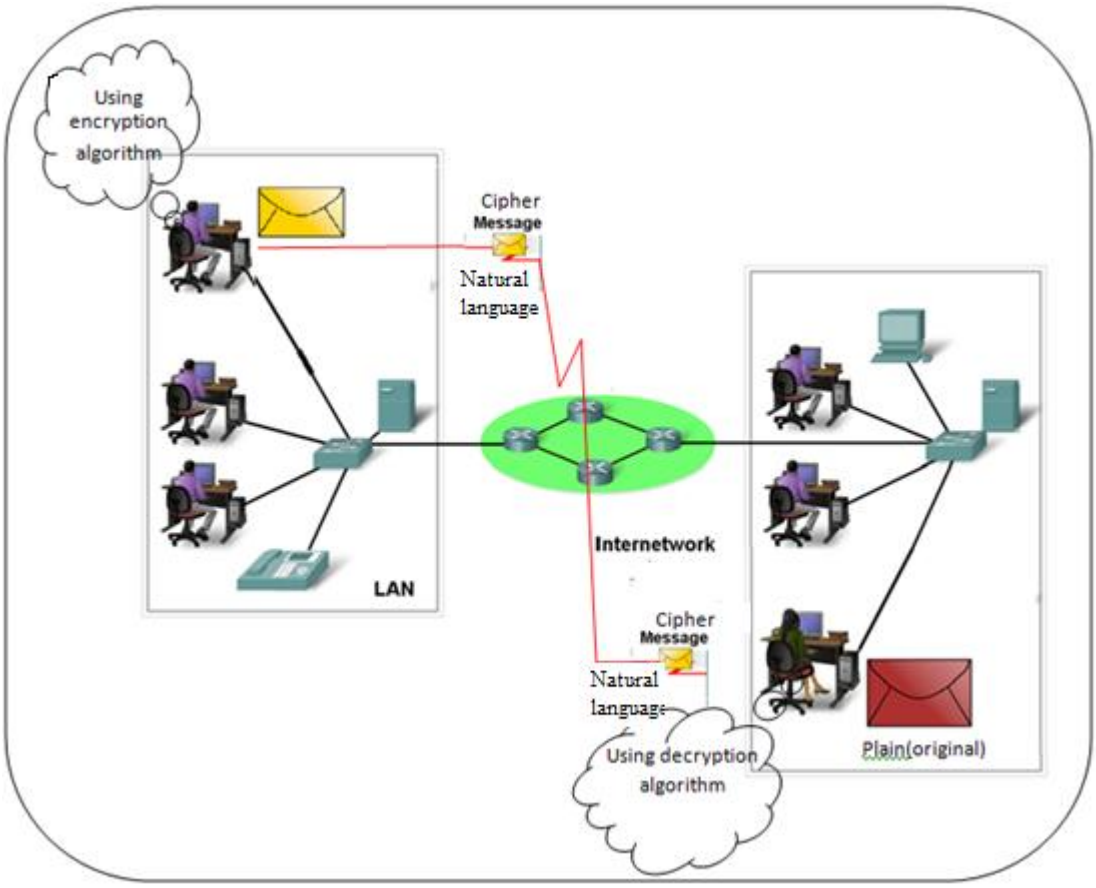

Figure 1: Proposed system architecture

3. Literature Survey

The proposed system presented in this paper deals with two issues; constructed cipher system which protect the data against attack by generating natural text (formal text). The random generation of text constrained by the rules of a text grammar, is of limited interest to workers in artificial intelligence since it is oriented more toward theoretical Linguistics then toward function Natural Language Processing system the objective of implementing a generation system of this sort is to test the descriptive adequacy of the test grammar or as illustrated by the following tow systems.

Victor Yngve (1961) [10]was one of the first researchers to attempt English text generation, the work was seen as preliminary to full program for machine translation Yngve used a generative Context Free Grammar and a random number generator to produce "grammatical" sentences. The system randomly selected one production from among those that were applicable to each point in the generation process ,starting from these production that "produced" <sentence>and finally randomly selecting wards to fill in the<Noun>,<Verb>,and other like positions .

Joyce Friedmans (1969, 1971) system [11, 12] was designed to test the effectives of transformational grammars. It operated by generating phrase markers (derivation tree) and by performing transformations of them until a surface structure was generated. The generation was random, but the user could specify an input phrase marker and semantic restrictions between various terminals in order to test specific rules for grammatical validity.

4. The Knowledge base for Ciphering Method:

In any knowledge based system there are may be three levels of knowledge [3, 7]:

The First Level (decision knowledge) is used to make a decision about the validity of the plaintext after it is cryptanalyzed. This is made by using the concept of natural language processing.

Many AI professionals believe that the most important task that AI can solve is NLP. This means that once a computer can understand and speak a human language, then there would no longer be a need for most tasks to be programmed by software engineers. This level will contain the knowledge that is used in the natural language. For example:

Part of the cipher text was cryptanalyzed and there are three characters in the end of the word that is unknown then depending on the language grammar. The program will recognize that the word is (objective). Then it get the knowledge about the adjective, and it replace these three characters with the knowledge about the adjective, and this process will repeated over the verbs, adverbs…..
The proposed system used the basic grammar to the English sentence such that:
Sentence → NP VP | Aux NP VP | VP
NP → Pronoun | Proper-Noun | Det Nominal

Nominal → Noun | Adjective | Nominal Noun | Nominal PP
VP → Verb | Verb NP | Verb Adverb NP | Verb Adverb | VP PP
Det → the | a | that | this
Noun → book | flight | meal | money
Verb → book | include | prefer
Pronoun → I | he | she | me
Proper-Noun → Houston | NWA
Aux → does
PP → Prep NP
Prep → from | to | on | near | through

The Second Level of knowledge base that contains knowledge that is so important that decision cannot be made without it. This knowledge of will contain not only the syntax of natural language but also additional consideration must be kept in the knowledge base such as:

1. 12 characters cannot be double.
2. 24 characters cannot be a word with length one.
3. 5 characters cannot be found in the end of the word.
4. 14 characters cannot be in the beginning of words with length two.
5. 16 characters cannot be in the end of words of length two.
6. There is 286 words of two letters cannot be used.
7. There is 163 words of two letters can be used.
8. There is 477 words of three letters usual.

For example if we use a predicate logic to represent the characters that does not come double such as:
Double (xx) double (nn)
Double (yy) ….
These predicates represent the characters that cannot come in the form of double, and so on for the other consideration.

The Third Level of knowledge is the meta- knowledge. An expert system not only should know things but also should know what it knows. It should be able not only to solve programs but also to explain how it solved the problems and why it made certain decision in the cryptanalysis of cipher text the meta- knowledge will be as follows [8]:

There are some words in English always repeated. Their frequencies are very high such as (the, are, is…) and there are also some letters always repeated in words like (e, t, h …).

These things are additional knowledge to the other knowledge that we have.

The proposed cryptography system dependent on the sequence of the Capital English alphabetic that we give each character a Code which points to its position in the alphabetic. And dependent on this code in process of construct a characters and generating a word. These codes for each character we display below:

The character	A	B	C	D	E	F	G	H	I	J	K	L	M	N
its code	0	1	2	3	4	5	6	7	8	9	10	11	12	13

The character	O	P	Q	R	S	T	U	V	W	X	Y	Z
its code	14	15	16	17	18	19	20	21	22	23	24	25

5. Text Generation

Computer text generation is the process of constructing text (phrases, sentences, and paragraph) in a natural language in sense; it is the opposite of natural language understanding by machine. Although few coherent principles has been emerged to this problem, the approaches have varied widely. Attempts as generated text have been made with two general research goals [4]:-

1) Generating random sentence to text a grammar or grammatical theory.
2) Converting information from internal representation into a natural language.

This research presented the first generation according to the knowledge base grammar.

5.1 Discourses in Language Generation[4, 6]

The majority of natural language researches deal with the structure of the sentences separated .in terms of considering the theoretical discussion of these issues some of these researches consider the connectivity of sentences and coherence of discourse, as in English Language. It is worthily mentioning that this issue had not been taking in the vided Arabic. Reference as a main topic relating with Arabic computational linguistics consider the language generation as an output from data base systems. In spoken or written language the focusing phenomena occurs of many of discourse. In a discourse a particular element that speaker usually centers his attention is called discourse element.

5.2 Focusing As a Tool for Generation

The development of discourse focusing has been taking in natural language processing specially in the interpretation of discourse; some of them are application in the production and the generation of natural language. There are two kinds of focusing Global Focus describe the effective of speakers centre of attention on the other hand, immediate focus refer to how the speaker over to consecutive sentences shifts or remains constant his centre of attention .in order to take advantage of focusing in generation natural language from D.B system.

6. Proposed Intelligent Secure System

Secure system including any method that keep the system and its data secure or we including the techniques enable to save that system. we may use the capability of AI to make the system more secure and produce intelligent secure system. First we speak about the way that we followed it when we implement the algorithms.

In this project we design and constructed an intelligent secure system. This intelligent security presented with cryptography during encryption and decryption algorithms. Text generation employs in encryption algorithm so that the ciphertext obtained being natural language. That's gives the necessary protection to hide the fact that the text is encrypted. That's prevent the breakers curiosity to attempt to cryptanalysis the

text. In addition to, these algorithms most robust against statistical cryptanalysis because them hiding the natural language properties and good randomness.

6.1 The Type of Algorithms

We presented four algorithms with different steps. All these depending on text generation presented in simple manner with algorithm 1. Now we display the algorithms we implemented it as following:-

6.1.1 Algorithm (1)

- **Encryption:**
  The algorithm is depending on the idea of partitions process for the plaintext in to separate characters then take each character and generating a word which must be start with this character .we still repeat this process until the end of the plaintext. This algorithm considered week system but we presented as base and step of other algorithms.
- **Decipher:**
  When we want, to Decipher this algorithm we reverse the previously process .we take each ward in the cipher text and. separate the first character from it then we repeated this process for all the wards after that we collected these characters each other's and obtain the plaintext.

Figure-1 illustrate the algorithm-1 Encryption and Decryption block diagram.

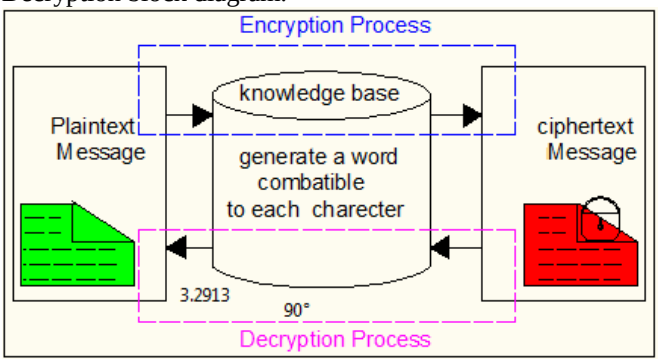

Figure 1: Algorithm-1 Encryption and Decryption block diagram

a) Practical Example:-

If the plaintext be = {IM Happy) then we cipher it as following:-

- Take the next char =1, generate word start with (I) → IS.
- Take the next char =M, generate word start with (M) → more.
- Take the next char =H, generate word start with (H) → High.
- Take the next char =A, generate word start with (A) → And.
- Take the next char =P, generate word start with (P) → Play.
- Take the next char =P, generate word start with (P) → Party.
- Take the next char =Y, generate word start with (Y) → Yet.

This cipher text = {is more high and play part yet}.

6.1.2. Algorithm (2)

This algorithm can be illustrate and explain by figure-2 block diagram below:

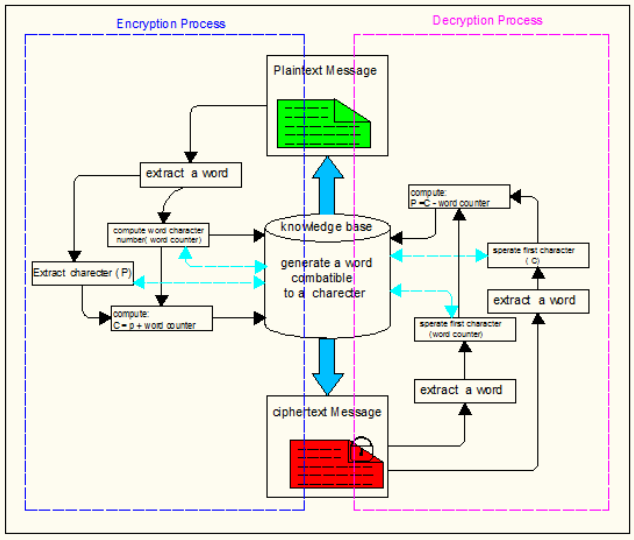

Figure 2: Algorithm-2 Encryption and Decryption block diagram

a) Encryption:

The main encryption process can be presented in the following steps:

Step1: Read or loaded plaintext message (original message).
Step2: Extract a word from plaintext.
Step3: Compute the number of characters in this word (w-counter)
Step4: Generate a word which must be start with character have the code equal to that w-counter by using the Knowledge base system.
Step5: Capture generated word in ciphertext file.
Step6: Extract a character from a extracted word (P).
Step7: Applying following equation on each character in this word after getting its code from knowledge base:

Code (new char-c) = code (old char-p)+ the number(w-counter)

Step8: Check the range of the (C) if C > alphabetic range then compute the following equation:

Code (new char-c') = code (new char-c) mod alphabet range.

Step9: Independent to the value of (C') in the process of generating the word by using the Knowledge base system.
Step10: Capture generated word in ciphertext file.
Step11: Repeat the steps (6-10) to all characters in an extracted word.
Step12: Repeat the steps (2-11)) to all words in ciphertext.

b)Decryption:

To de- cipher a text which cipher by using the algorithm-2 we followed the steps below:

Step1: Read or loaded ciphertext file (ciphertext message).
Step2: Extract a word from ciphertext.
Step3: Extract first character from a word.
Step4: compute character code from Knowledge base system, which represent w-word.
Step5: Extract next word from ciphertext.
Step6: Extract a character from a extracted word (C).
Step7: Applying following equation on each character in this word after getting its code from knowledge base:

Code (old char-p) = code (cipher char-C) — W-counter

Step8: Check the range of the (P) , if the value calculated of P is negative value then compute the following equation:

Code (old char-p') = alphabetic range - Code (old char-p)

Step9: Decrement W-counter

Step10: Independent to the value of (P or P') in the process of generating the plain character by using the Knowledge base system.

Step11: Capture a character in Plaintext file.

Step12: Repeat the steps (5-11) until W-word = 0.

Step13: Repeat the steps (2-12)) to all in ciphertext.

c) Practical Example:-

If we want to cipher the plaintext {IM HAPPY}, then we do the following:-

1- Take the next word in this text = TM.
2- Count the number of characters in this word, W count= 2.
3- Generate word which must be start with character have code=2 code (2) = B, generating word =Be.
4- Implement the following equation for all the characters in this Word:-

Code (new char-cl-) code (I) + W count = 9+2 = 11
Code (11) = K, generating word = Keep.
Code (new char-c2-) code (M) + W count = 13+2 15
Code (15) = 0, generating word = Open.

5- Now, we repeat the steps 1,2,3,4, for all words in the text. Take the new next word HAPPY, count the number of character on it =5, code (5) =E, generating word=Easy, then we apply the equation:

Code (new char-c1-) = code (A) + W count = 1+5 =6, 6 <=26
Code (6) = F, generating word Five.
Code (new char-c3-) = code (P) + W count = 16+5 =21, 21<=26
Code (21) U, generating word = Until.
Code (new char-c4-) = code (P) + W count = 16+5 =21, 21<=26
Code (21) = U, generating word = University.
Code (new char-c5-) = code (Y) + W count 25+5 =30, 30<=60, then false
Code (new char-c5-) = 30 mod 26 =4
Code (4) = D , generating word = Done.
The cipher = {Be Keep Open Easy Five Until, University}.

6.1.3. Algorithm (3)

In this algorithm we depend on the use of key. That key can be represented and used in different ways, such that to abound a diffusion manner to cipher system. In the practical proposed algorithm the key determined to special value and when the counter is being equal to a special value (we determine it), we re- initialize it. When want to cipher a text we dealing with this text as a collection of characters, we separate each character from the text and take its code and applying the Encryption algorithm. Figure-3 bellow illustrates the block diagram and process steps of this algorithm:

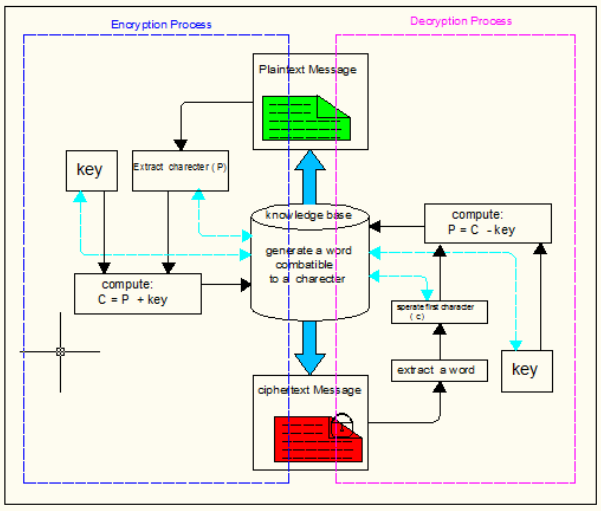

Figure 3: Algorithm-3 Encryption and Decryption block diagram.

a) Encryption:

The main encryption process can be presented in the following steps:

Step1: Read or loaded plaintext message (original message).
Step2: Read key and initialized counter to 1.
Step3: Extract a character from a plaintext (P).
Step4: Applying following equation on a character (P) after getting its code from knowledge base:

Code (new char-c) = code (old char-p)+ the value of counter

Step5: Increment counter
Step6: Check the range of the (C) if C > alphabetic range then compute the following equation:

Code (new char-c') = code (new char-c) mod alphabet range.

Step7: Independent to the value of (C') in the process of generating the word by using the Knowledge base system.
Step8: Capture generated word in ciphertext file.
Step9: re-initialize the counter to 1 when the value of counter equal to the key, otherwise skip this step.
Step10: Repeat the steps (3-9)) to all characters in plaintext.

b) Decryption:

In the process of deciphering this algorithm we followed the steps below:-

Step1: Read or loaded ciphertext file (ciphertext message).
Step2: read key. And initialized counter to 1.
Step3: Extract a word from a ciphertext.
Step4: Extract first character from an extracted word (P).
Step5: Applying following equation on a character (P) after getting its code from knowledge base:

Code (old char-p)= Code (cipher char-C) - the valuae of counter

Step6: Increment counter
Step7: Check the range of the (P) , if the value calculated of P is negative value then compute the following equation:

Code (old char-p') = alphabetic range - Code (old char-p)

Step8: Independent to the value of (P or P') in the process of generating the plain character by using the Knowledge base system.
Step9: Capture getting character in plaintext file.
Step10: re-initialize the counter to 1 when the value of counter equal to the key, otherwise skip this step.

Step11: Repeat the steps (3-10)) to all characters in ciphertext.

6.1.4. Algorithm (4)

This algorithm deals with sentence, such that takes each sentence in the text. Figure-4 , the block diagram of this algorithm represented process steps to both encryption and decryption algorithms.

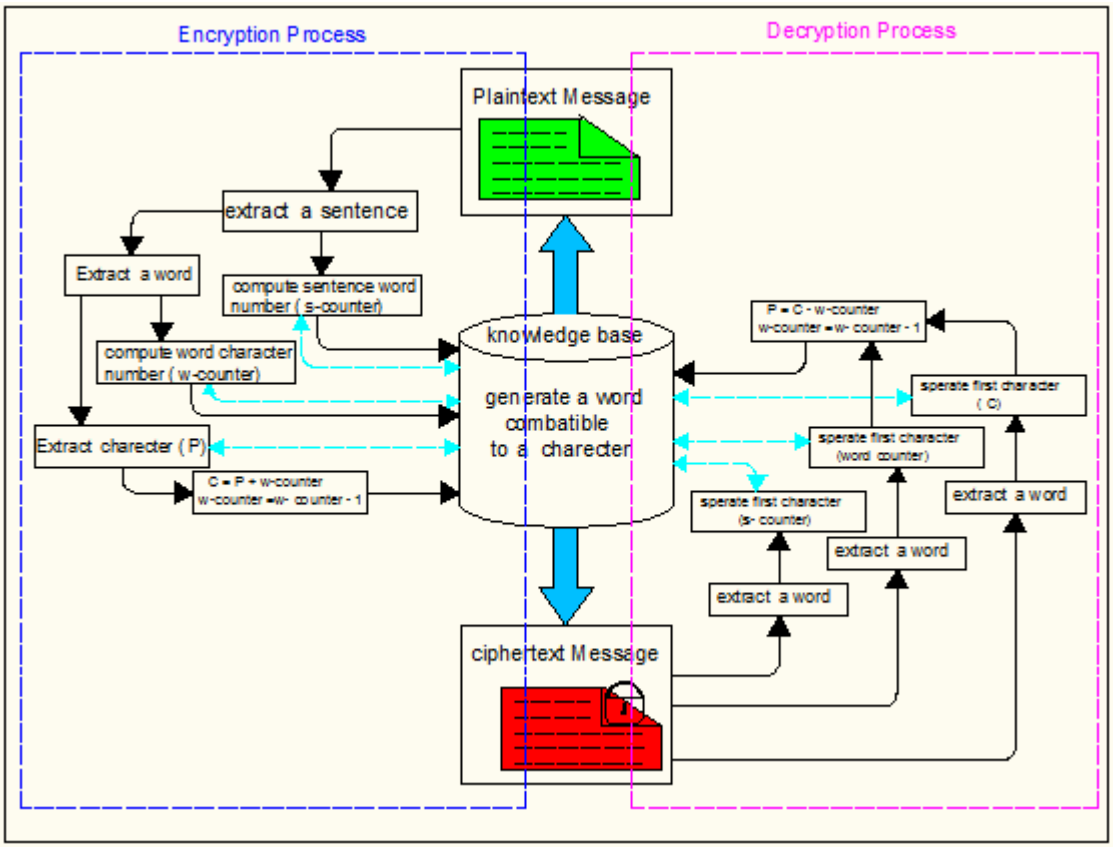

Figure 4: Algorithm-4 Encryption and Decryption block diagram

a)Encryption:

The main encryption process can be presented in the following steps:

Step1: Read or loaded plaintext message (original message).
Step2: Extract a sentence from plaintext.
Step3: Compute the number of word in this sentence (S-counter)
Step4: Generate a word which must be start with character have the code equal to that S-counter by using the Knowledge base system.
Step5: Capture generated word in ciphertext file.
Step6: Extract a word from an extracted sentence.
Step7: Compute the number of characters in extracted word (W- counter)
Step8: Generate a word which must be start with character have the code equal to that W-counter by using the Knowledge base system.
Step9: Capture generated word in ciphertext file.
Step10: Extract a character from an extracted word (P).
Step11: Applying following equation on each character in this word after getting its code from knowledge base:

Code (new char-c) = code (old char-p)+ (w- counter) + (S-counter)

Step12: Check the range of the (C) if C > alphabetic range then compute the following equation:

Code (new char-c') = code (new char-c) mod alphabet range.

Step13: Independent to the value of (C') in the process of generating the word by using the Knowledge base system.
Step14: decrement W-count.
Step15: Capture generated word in ciphertext file.
Step16: Repeat the steps (10-15) to all characters in an extracted word.
Step17: Decrement S-counter.
Step18: Repeat the steps (6-17)) to all words in an extracted sentence.
Step19: Repeat the steps (2-18)) to all sentences in plaintext.

b)Decryption:

In the process of deciphering this algorithm we followed the steps below:-

Step1: Read or loaded ciphertext file (ciphertext message).
Step2: Extract a word from ciphertexrt.
Step3: Separate first character from extracted word and getting its code from knowledge base to represent S- counter.
Step4: Extract next word from ciphertexrt.
Step5: Separate first character from extracted word and getting its code from knowledge base to represent W-counter.
Step6: Extract next word from ciphertext.
Step7: Separate first character (P) from extracted word and getting its code from knowledge base to represent code of (P).
Step8: Applying following equation:

Code (original char-P) = code (ciphered char-C) - (w-counter) - (S- counter)

Step9: Check the range of the (P) , if the value calculated of P is negative value then compute the following equation:
Code (old char-p') = alphabetic range - Code (old char-p)
Step10: Independent to the value of (P or P') in the process of generating the plain character by using the Knowledge base system.
Step11: Capture getting character in plaintext file.
Step12: decrement W-count
Step13: Repeat the steps (6-12) until W-count =0.
Step14: Repeat the steps (4-13)) until S-count =0.
Step15: Repeat the steps (2-14)) to all ciphertext.

7. Experimental Analysis and Results

Any general cipher system teds to provide the diffusion and confusion manner; which gives the strongest ciphertext in same time the hardest breaking and cryptanalysis. The diffusion can be provides in proposed system by modifying the actual plain character codes according to special keys. The confusion provide by adding many words in cipher text represent the word and sentence counter. The main material of proposed system is the NLP, who's used to generate the output ciphertext. Proposed system cipher space must be natural plotting to prevent suspicion to any one for truth text [5].

The main interface represents the first step in system. In this interface we can input the message and choose the algorithms wanted to used in encryption and decryption as well as inner choosing the process (encryption or decryption) the following figures (5,6,7,8) represent the resulted interface to each algorithm

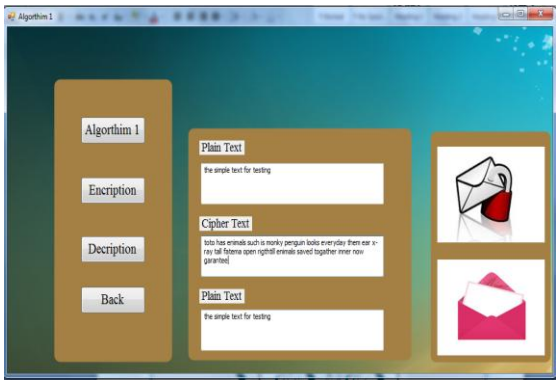

Figure 5: Algorithm-1 Encryption and Decryption Results

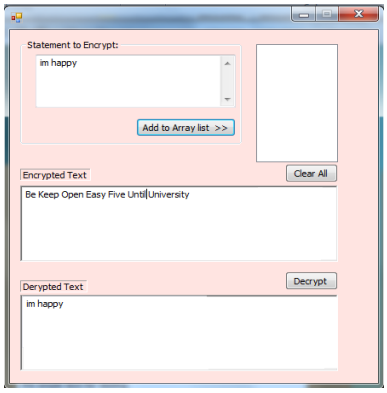

Figure 6: Algorithm-2 Encryption and Decryption Results

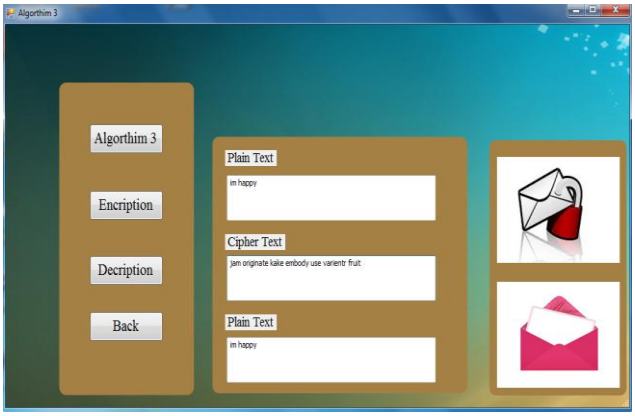

Figure 7: Algorithm-3 Encryption and Decryption Results

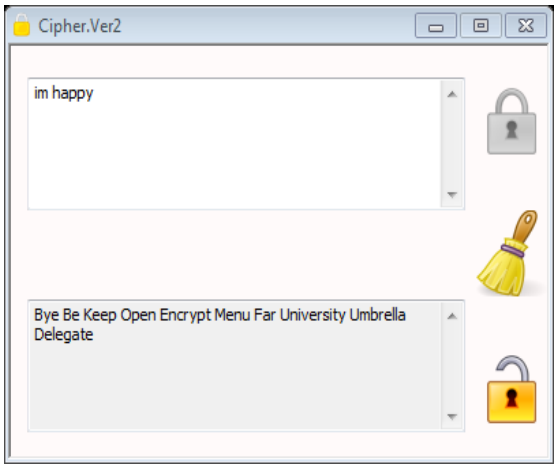

Figure 8: Algorithm-4 Encryption and Decryption Results

The proposed system different from general cipher systems in a basic concept, which is the outputs (ciphertext) must be legally (natural language). Therefore we can't use the same tested and evaluation methods to these proposed algorithms. To evaluate the proposed system and give them algorithms the strength we determine the following points (these points deal with all algorithms except algorithm-1):

- **Plaintext-Ciphertext Correlation Test**

It is a mathematical tool for finding repeating patterns; In statistics, the autocorrelation of a random process describes the correlation between values of the process at different times, as a function of the two times or of the time lag [9]; as evaluation proposed system after converting plaintext and ciphertext to binary sequences, then deals with it as sequence of binary bits. The obtained result of autocorrelation function can be determined during the following table-1:

	Algorithm1	Algorithm2	Algorithm3	Algorithm4
Autocorrelation	- 0.0173913	-0.1217391	- 0.07826087	- 0.173913

- The cryptanalyst can't predict the length of the plaintext because the plaintext and ciphertext don't have the same length.
- The ciphertext of proposed system keeping the frequencies characteristic of the natural language. The figure-9 shows the plaintext and ciphertext plotting.

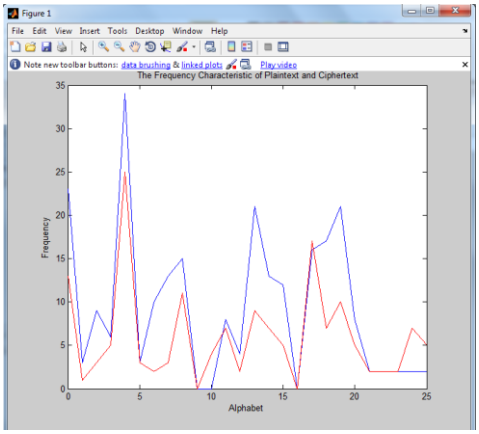

Figure 9: The Plaintext- Ciphertext Frequency Characteristic

➢ The proposed algorithms provide protection against cryptanalysis. If a cryptanalyst intercept part of the sequence, and have no information on how to predict what comes next.

8. Conclusion

In this paper we present encryption and decryption algorithms which used with text data depending on the concept of artificial intelligent. The proposed algorithms provide the following aims:

- **The privacy:** which cannot anyone except the exact receiver reads the original message.
- **Data integrity:** which guarantee that no changed and manipulated doing in cipher message data during transmission.
- **Authentication:** which provides the Verification of the person that you want to read the sent message.
- **Non repudiation:** which makes the person whose ciphertext message sent to him Unable to denial that he's the right person the message sent to him.
- **The cover security:** which prevents suspicion to anyone about this message is encrypted. Because the ciphertext is natural language also.
- There are two reasons for separating the knowledge based from the inference engine:-

1) If similar kinds of things are grouped together, it will be easier to find rules that do particular kinds of work in the system, so that they can be modified, deleted, or added to as necessary.
2) If all the knowledge rules are kept separate from the inference rules, it will make it easier to use the design for new expert system. An expert system with knowledge rules removed is called a "shell". They are programs that have been constructed by using an existing system in which the knowledge rules were removed, and new knowledge rules were then installed that were tailored to a new problem.

9. Suggestion

These algorithms provide good security using the artificial intelligent concepts. Therefore the working and developing on this work very visible. We can suggest some future works:

- Applying scrambling manner as a pre-process to original message before algorithms to increase the randomness and permutation to the ciphertext included the system confusion manner.
- Implication semantic manner to these algorithms.
- Applying syntax process as post-process to the ciphertext depending on the original production rule to the origin language.

References

[1] Robert Keller, "Expert System Technology: Development and Application", January 1987.
[2] George, F-luger and William A. Stubblefied, "Artificial intelligent and the design of Expert System", 1989.
[3] Spyros G. Tzafestas, "Knowledge- based system Diagnosis, supervision and Control", 1989.
[4] Michael Heilman, "Automatic Factual Question Generation from Text", ph. D thesis, School of Computer Science, Carnegie Mellon University, 2011.
[5] Anonymous, "Maximum Security", 1st Edition, Sams.net Publishing, 1997.
[6] Ronan Collobert, JasonWeston, , L´eon Bottou, Michael Karlen, Koray Kavukcuoglu, Pavel Kuksa, " Natural Language Processing (Almost) from Scratch" , Journal of Machine Learning Research 12 (2011) 2493-2537, 2011.
[7] K.P. Valavanis, G.N. Saridis' "Intelligent Robotic Systems: Theory, Design and Applications", Kluwer, 1992.
[8] K. Brunnstein, S. Fischer-Huebner, M. Swimmer, "Concepts of an Expert System for Virus Detection", Information Security, Elsevier Science Publishers B.V. (North-Holland), IFIP, 1991.
[9] David, "A Simple Autocorrelation Algorithm for Determining Grain Size from Digital Images of Sediment", journal of sedimentary research, vol. 74, no. 1, 2004.
[10] Victor H. Yngve, " Random generation of English Sentences", First international conference on machine translation of languages and applied language analysis, pages 66-80, Teddington, England, 1961.
[11] Joyce Friedman, " Application of Computer System for Transformational Grammar", Preprint No. 14, in International Conference on Computational Linguistics, Stockholm, 1969.
[12] Joyce Friedman, et al., " A Computer Model of Transformational Grammar (Mathematical Linguistics and Automatic Language Processing Nr.9), New York, London, Amsterdam, 1971.

Author Profile

Awarded her B.Sc, and M.Sc, at University of Technology, Department of Computer Science and information systems-information systems in 2000 and 2003 respectively. She is a lecturer at Karbala University, Collage of Science, Computer Department. Here research interests include: Object Modeling, Image processing such as Segmentation and Steganography, Data Security, Artificial intelligent, artificial intelligent applications and information systems.