A Novel Approach for Arabic Text Steganography Based on the "BloodGroup" Text Hiding Method

Suhad Malalla
Department of Computer Science
University of Technology
Baghdad, Iraq
suhad_malalla@yahoo.com

Farah R. Shareef
Department of Computer Science
University of Technology
Baghdad, Iraq
sabahaliraq2014@gmail.com

***Abstract*—Steganography is the science of hiding certain messages (data) in groups of irrelevant data possibly of other form. The purpose of steganography is covert communication to hide the existence of a message from an intermediary. Text Steganography is the process of embedding secret message (text) in another text (cover text) so that the existence of secret message cannot be detected by a third party. This paper presents a novel approach for text steganography using the Blood Group (BG) method based on the behavior of blood group. Experimentally it is found that the proposed method got good results in capacity, hiding capacity, time complexity, robustness, visibility, and similarity which shows its superiority as compared to most several existing methods.**

Keywords-Information Security; Steganography; Hiding Message; Text Steganography; Arabic Stego

I. INTRODUCTION

Exchanging hidden information is an important domain of information security which includes different methods such as cryptography, steganography, etc. [1]. Steganography is a new-fangled approach of securing messages from outside interference/attacks in an exceptional way. This technique provides a way that anticipated recipient recognizes the subsistence of the message(s). This is just obscuring a file, message, image, or video within another. Steganography is a way of covering a message surrounded by an extra message, so that nobody can have a notion of its presence. Messages can be perceived by the pre-determined addressee only. In steganography the information is concealed in the cover media so no person observes the presence of the secret information. The working of steganography has been implemented on different medias like text, video, images, and sounds [2].

Fundamentally, stenographic techniques need to find insignificant bits of medium or cover files. Thus, any kind of modifications to those insignificant bits shouldn't damage the integrity of the cover medium. Nevertheless, undetectability might be generally implemented by adding invisible modifications to the cover file. For text steganography method, the stego text fidelity is generally utilized to evaluate and calculate the undetectability of the steganography method used. Nonetheless, fidelity describes the possibility of persons to discover variations between the stego and cover text. Nonetheless, the cover text integrity is not conserved with steganography because some parts of the cover file needs to be modified or changed to be able to hide the secret message and obtain the stego file [3]. Text stenography is not usually utilized since the files of text a have tiny amount of unnecessary data.

The aim of this paper is to describe a new method for text steganography through text and proposed a novel steganography technique for Arabic language called "Blood Group" stenographic method.

II. PREVIOUS WORKS

The first proposal in the Arabic text steganography was done in [4]. Their schema is dependent on concealing binary values within Persian or Arabic scripts by using a characteristic of coding strategy. This approach is dependent upon the points inherited within the Persian, Urdu and Arabic letters. The points' location on the letters conceal certain data. The hidden data length (i.e. the Secret Object) is considered as a binary using the first several bits. The middle text, (i.e. the Cover Object) is scanned in such a way when a pointed letter is detected. The position of the point is shifted up slightly when the hidden value is 1otherwise, the location remains unchanged. The advantage of this method is the large amount information will be concealed in text due to the large number of points in letters for both Persian and Arabic.

Authors in [5] proposed a new steganography approach to conceal hidden secret data within Arabic text cover media. The suggested method employs Arabic language diacritics that are utilized for vowel sounds and it's located in several religious documents. About 8 of these symbols are found in Arabic. They discovered that the "Fatha" symbol has been used in Arabic text more than the other 7 symbols. Thus, they used the "Fatha" symbol to present 1 and the other symbols to present 0. The benefit of this method is the big capacity due to each Arabic character is relevant for a diacritic. The downside is that concealing some diacritics can get the reader's attention. In [6] the reverse "Fatha" was used to conceal data in the cover text rather than the regular "Fatha". This was not as easily noticed by third parties which is a benefit for this method. The downside of this method is the requirement for a new font to apply the "inverse Fatha" as it isn't a standard diacritic.

In [7], authors proposed a new method by using "Kashida". Several algorithms were developed and performed in a stego method named MSCUKAT (Maximizing Steganography Capacity Using "Kashida" in Arabic Text). The enhancements with this attempt involve maximizing the capacity of cover media to conceal more secret information, minimizing the file size that rise after hidden the secret and improving the security of the encoded cover media. It was shown that this method was superior over similar previous. In [8], eight different techniques that deal with using Arabic natural language to hide secret message were considered. The authors of this paper have also published a previous similar study in [9].

III. THE PROPOSED TECHNIQUE

A. *Architecture*

The architecture of system is same as in [9], which is organized with two portions: the sender side that encrypt the secret message, compress it and then embed it and the receiver side that decodes the stego message and by using the same encrypted key.

B. *Stego Module*

The hiding process is used by the sender to hide the secret message into the cover text. This process involves select the input file, which represents the encrypted compressed secret message, and a group of sub processes as represented in (1). The proposed method is based on employing two stego options of "Kashida" and change the Unicode of the letter based on the behaviour of the blood group (ABO).

CoverText+SecretInformation = StegoText (1)

Let Group A connect dotted Arabic characters like (بـ,تـ,ثـ) except characters of the AB group, Group B connect not-dotted Arabic characters like (ـح,ـر,ـس) except characters of AB group, and Group AB separate Arabic characters like (د ,ر,س,أ). The steps of blood group apply algorithm is as following.

1. Assume that P is previous character and C is current character.
2. If P is (A group) and C is (A group) or P is (B group) and C is (B group) then insert Kashida and return, for example:
   - Example 1*:* Case (P is (A group) and C is (A group)): if the word is (شجرة), where (P=ش, C=ج), the result after applying kashida on P, will be (شـجرة).
   - Example 2: Case (P is (B group) and C is (B group)): if the word is (سماح), where (P=س, C=م), the result after applying kashida on P, will be (سـماح).
3. If C is AB group:
   - If P is (A group or B group), replace C from ISO-8859-6(Arabic) code to Medium type, for example:
     - Example 3: Case (P is (A group) and C is (AB group)): if the word is (غرفة), where (P=غ , C=ر),the result after applying Medium type on C letter, will be (0xFEAE).
     - Example 4: Case (P is (B group) and C is (AB group)): if the word is (سد), where (P=س , C=د),the result after applying Medium type on C letter, will be (0xFEAA).
   - If P is AB group or AB` group, replace C from ISO-8859-6 (Arabic) code to isolated type, for example:
     - Example 5: Case (P is (AB group) and C is (AB group)): if the word is (در), where (P=د , C=ر), the result after applying isolated type on C letter, will be (0xFEAE).

Figure 1 shows the blood group behavior and Table I the mapping from AB to AB` for isolated characters.

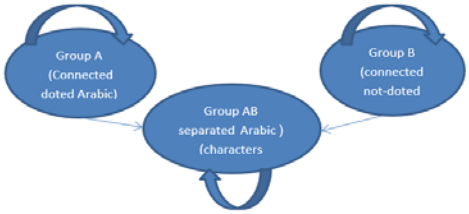

Fig. 1. Behavior of blood group Stego method.

TABLE I. MAPPING FROM AB TO AB` FOR ISOLATED CHARACTERS

№	Unicode	Character	Isolated Type	Medium Type
1	0x0621	ء	0xFE80	0xFE80
2	0x0622	آ	0xFE81	0xFE82
3	0x0623	أ	0xFE83	0xFE84
4	0x0624	ؤ	0xFE85	0xFE86
5	0x0625	إ	0xFE87	0xFE88
6	0x0627	ا	0xFE8D	0xFE8E
7	0x0629	ة	0xFE93	0xFE93
8	0x062F	د	0xFEA9	0xFEAA
9	0x0630	ذ	0xFEAB	0xFEAC
10	0x0631	ر	0xFEAD	0xFEAE
11	0x0632	ز	0xFEAF	0xFEB0
12	0x0648	و	0xFEED	0xFEEE

The Stego algorithm of this method is described in follows:

1) *Embedding Algorithm*

Input: Encrypt Compressed Secret message (ECSM) , Arabic cover.
Output: Arabic Stego Text.

Step 1: Start.
Step 2: Read two characters (P-previous, C-current).
Step 3: Call for blood group A (dotted letters), blood group B (non- dotted letters) and blood group AB (isolated letters).
Step 4: Check:

- If (ESM/ECSM-bit=1) and (previous letter is from A or B) and (current letter is from A or B) Then:

i. Remove Kashida from Arabic cover text (if exist).
ii. Check Arabic Letters Conditions:
- Arabic Code Zone.
- Not Start or End of a word.
- Not between ل and ا.
- Kashida table.

iii. Put "Kashida" after the previous letter.

- If (ESM/ECSM-bit=1) and (previous letter is from A) and (current letter is from AB) Then change the Unicode to medium type for the current letter.
- If (ESM/ECSM-bit=1) and (previous letter is from B) and (current letter is from AB) Then change the Unicode to medium type for the current letter.
- If (ESM/ECSM-bit=1) and (previous letter is from AB) and (current letter is from AB) Then change the Unicode to isolated type for the current letter.

Step 5: Combine the results and return.
Step 6: Output Arabic stego text.
Step 7: End.

2) Extracting Algorithm

Input: Arabic Stego Text
Output: Encrypted Compressed Secret message (ECSM)

Step 1: Start.
Step 2: Convert stego-cover to UTF-8.
Step 3: Check:
- If (previous letter is from A) and (code of Kashida is exist) and (current letter is from A) then return 1.
- If (previous letter is from B) and (code of Kashida is exist) and (current letter is from B) then return 1.
- If (previous letter is from A) and (code of Unicode for current letter is exist) and (current letter is from AB) then return 1.
- If (previous letter is from B) and (code of Unicode for current letter is exist) and (current letter is from AB) then return 1.
- If (previous letter is from AB) and (code of Unicode for current letter is exist) and (current letter is from AB) then return 1.
- Otherwise, return 0.

Step 4: Gather each 8-bits to get byte and then convert it to string.
Step 5: End

IV. RESULTS

This part shows the experiments results used to evaluate the performance of the proposed system. The system has been utilized in C#. The tests were run in a workstation laptop (Dell) with the following specifications: CPU 1.8 GHz core i3, RAM , 4GB DDR3, OS Windows 8 64bit, Visual studio 2013

A. Capacity

At first we calculated the secret messages sizes before and after encryption and compression ratio to determine the changes and the gain from using compression to reduce the message sizes that will be improving the hiding process by reducing cover capacity needed. As shown in Table II, the results show that compression with (gzip) is very useful with large secret message that is be practically efficient.

B. Hiding capacity

The Hiding capacity (in bits/Bytes) need to hide two secret messages with fixed cover. The cover capacity needs for Blood Group stego method is shown in Table III.

C. Time Complexity

In order to calculate the actual time needed to embed with the Blood Group method a timer function was used. Table IV shows the embedding time needed for this stego method for 2 secret messages with a fixed cover message.

D. Robustness, Visibility, & Similarity

Robustness is the resistance of the steganography technique against modifying or destroying the secret message. Results are shown in Table V.

V. CONCLUSION

A hybrid method that combines both cryptography and steganography is considered in this paper. A novel algorithm called "BloodGroup" is proposed. The new scheme demonstrated rather good results in terms of capacity, speed, robustness, printing, copying and pasting, font changing, similarity, visibility and security but poor results in OCR which should be attributed to the poor level of the available Arabic OCR software. Further improvement is achieved when the GZIP compression technique is also employed.

TABLE II. SECRET MESSAGE CAPACITY WITH OR WITHOUT ENCRYPTION AND COMPRESSION

Secret Message Language	Secret Message (length)	Before Encrypt+ Compression		After Encryption(AE)		After Encrypt+ Compression (AEC)		Comp. Ratio*
		No. of 1's	No. of char.	No. of 1's	No. of char.	No. of 1's	No. of char.	
Arabic	1340	8871	19152	9659	19736	4261	8472	57.0%
English	3212	11395	25720	13114	26264	6713	13592	48.2%

*Comp. ratio=(size before comp. (AE)-size after comp. (AEC))/size before comp. (AE)

TABLE III. HIDING CAPACITY FOR 2 SECRET MESSAGES WITH FIXED COVER (LENGTH=258107).

Length of cover (real used) With (Encrypt + Compression)					
Real used of cover	Secret (S1)	Hiding capacity (bits / bytes)	Real used of cover	Secret (S2)	Hiding capacity (bits / bytes)
37572	8472 bits	22.5	60330	13592 bits	22.5

*(Hiding capacity= secret (bits) / real used of cover (bytes)

TABLE IV. EMBEDDING TIME NEEDED FOR THREE DIFFERENT SECRET MESSAGE WITH FIXED COVER

Embedding time (seconds)			
Secret Message (S1)		Secret Message (S2)	
Encryption	Encryption + Compression	Encryption	Encryption + Compression
0.486	**0.509**	**0.535**	**0.547**

TABLE V. ROBUSTNESS, SIMILARITY AND VISIBILITY

Robustness					Similarity	Visibility
Printing	OCR	Copying & pasting	Font changing	Retyping		
√	X	√	√	X	0.926	Hard to notes

REFERENCES

[1] M. Shirali-Shahreza, M. H. Shirali-Shahreza,An Improved Version of Persian/Arabic Text Steganography Using "La" Word, 2nd Malaysia Conference on Photonics. NCTT-MCP 2008 and 6th National Conference on Telecommunication Technologies 2008 and 2008, August 26-28, 2008

[2] C. Cachin, "An Information-Theoretic Model for Steganography", 2nd Information Hiding Workshop, Vol. 1525, pp. 306-318, 1998

[3] R. J. Anderson, F. A. P. Petitcolas, "On the Limits of Steganography", IEEE Journal on Selected Areas in Communications, Vol. 16, pp. 474-481, 1998

[4] H. Shirali-Shahreza, M. Shirali-Shahreza, "A New Approach to Persian/Arabic Text Steganography", 5th IEEE/ACIS International Conference on Computer and Information Science (ICIS-COMSAR 06), pp. 310- 315, 2006

[5] A. Mohammed, A. Sameh, E. Abdul Rahman, G. Adnan, "Arabic Diacritics Based Steganography", IEEE International Conference on Signal Processing and Communications (ICSPC'07), pp. 756–759, Dubai, UAE, 2007

[6] A. M. Jibran, K. Kamran, K. Hameedullah, "Evaluation of Steganography for Urdu/Arabic Text", Journal of Theoretical and Applied Information Technology, Vol. 4, No. 3, pp. 232–237, 2008

[7] G. Adnan, A. N. Ahmed, "High Capacity Steganography Tool for Arabic Text Using 'Kashida", The ISC International Journal of Information Security, Vol. 2, No. 2, pp. 107–118, 2010

[8] A. A. Maisa, A framework to design and implementation of linguistic Steganography system, PHD thesis, Department of Computer Science, University of Technology, 2016

[9] S. Malalla, F. R. Shareef, "Improving Hiding Security of Arabic Text Steganography by Hybrid AES Cryptography and Text Steganography", Journal of Engineering Research and Application, Vol. 6, No. 6, pp. 60-69, 2016