

New text steganography method using the arabic letters dots

Ahlam R. Khekan¹, Hiba Mohammed Wajeh Majeed², Omer F. Ahmed Adeb³

^{1,2}University of Information and Communication Technology, Baghdad, Iraq

³Razi University, Iran

Article Info

Article history:

Received Jul 18, 2020

Revised Sep 27, 2020

Accepted Dec 4, 2020

Keywords:

Characters dots

Huffman coding

Stego-cover

Text steganography

ABSTRACT

With the increasing technological and electronic development, methods have been developed to hide important information using text steganography as a new technology, since it is not noticeable and easy to send and receive. The use of the Arabic language is one of the new methods used to hide data. In this work, we preview our method that depends to use the part of Arabic language properties to embed the secret English message in to cover text to create text steganography. More than half of the Arabic characters contain dots. Several characters have upper dots and others have lower dots. Some have one dot others have two dots. Few have even three dots. In this new idea, we will use the dots of characters to embed the English secret message. First, we will compress the secret message by using the 5-Bit Encoding (T-5BE) to make the cover text able to embed more bits of the secret message by 37.5%. Then we start using the Arabic semantic dictionary to correct the hiding path and enhancement the stego-cover text to eliminate errors caused by switching words. In this research, we were able to extract experimental results that show that the proposed model achieves high masking accuracy in addition to the storage capacity of the cover text.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Ahlam Rashid Kharbat Khekan

University of Information and Communication Technology

Baghdad, Iraq

Email: ahlamr.khekan@uoitc.edu.iq

1. INTRODUCTION

The revelation of computer and electronic technology expanded in very fast these twenty years that make a lot of information and data in dangers because the technology thief takes any chance to still the information and data to use it for illegal profit [1, 2]. When the people want to transfer data must have the security so they start to protect the information by using the cipher and coding [3, 4]. The data and information, the steganography is the artwork of hidden data and information in other cover and every cover have the different type there is image-cover, audio-cover, video-cover, and text-cover [5]. The most difficult and tricky is the text steganography it's used from the long-time even before the computer and technology are found and they use it to send the message between the spy's it protects the sender and receiver in the same time [6]. In the usually use written on wax and hide the message under it, or the chemical material to write the secret on the animal leather without making any possible visual for the secret message and when the technology arrives the text steganography change to electronic [7, 8].

Arabic language property: The Arabic language very rich with properties first it's the cursive characters it means the characters had different shape when being in the beginning or middle or the end of the word like (ععع) it's the same character with three different shapes [9], second every character had another special small shape upper or lower the famous shapes about (7) kind like (ع, ع, ع, ع, ع, ع, ع), third all the characters in the middle can be starched in operation called 'Kashida' like (ع—ع) all these properties we can

use it to embedded one bit or more [10], fourth the Arabic language unique than another national language because the Arabic language had (28) characters (15) of them contains a dots as shown the Table 1 we can use the dots to hide a lot of bits in every word [11, 12].

Table 1. The Arabic character properties

Letters properties	Letters with number of dots
Letters without dots	ا, ح, د, ر, س, ص, ط, ع, ك, گ, ل, م, و, ه
Letters with one dot	ب, ج, خ, ذ, ز, ض, ظ, غ, ف, ن
Letters with tow dots	ت, ث, ي
Letters with three dots	ش, پ, ز, چ

The letters of letters change from one text to another in Table 2. To know the correct frequencies for each of the Arabic letters, the Holy Qur'an frequencies were found and considered the main and approved frequencies because the Holy Qur'an is considered a reference for the Arabic language and most of the book uses the Arabic language rules from the Holy Quran [13], and every Arabic letter had a unique frequency Figure 1.

Table 2. Letters repeated on Holy Qur'an

No.	Ch.	Fr.	No.	Ch.	Fr.
1	أ	48940	15	ض	1882
2	ل	33922	16	ط	1264
3	ن	26925	17	ظ	842
4	ث	2115	18	ع	9470
5	ح	3322	19	ع	1229
6	ح	4138	20	ف	9813
7	ح	2503	21	ق	8099
8	د	5798	22	ك	8022
9	ذ	4934	23	ل	33922
10	ر	2206	24	م	28922
11	ز	1680	25	ن	1700
12	ع	5799	26	ه	26925
13	ث	2115	27	و	25506
14	ص	2780	28	ي	5717

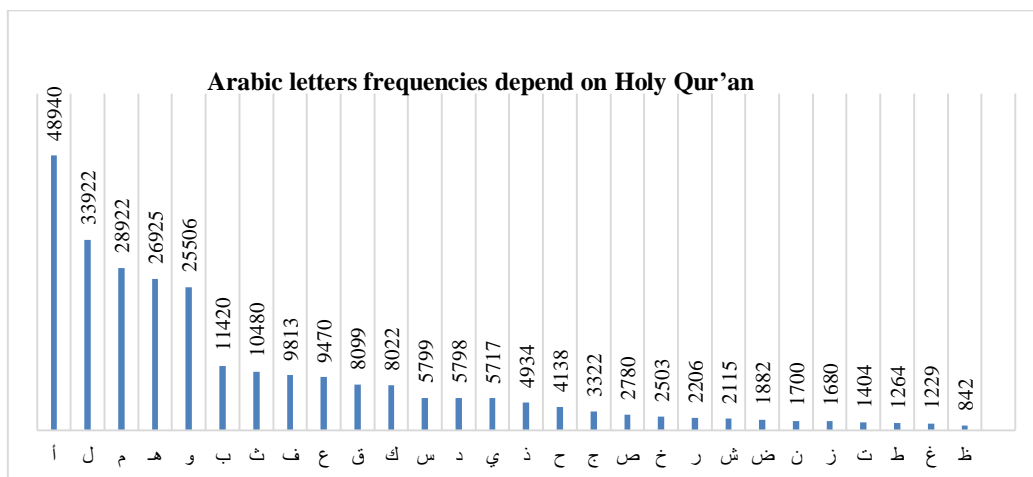


Figure 1. The Arabic letters frequencies

2. RELATED WORK

2.1. Shifting letters dot [14]

In 2006 start to use the Irani language to hide the secret message by changing the character dot upper to hide the bit (1) and lower to hide bit (0) for make stego-cover and convert it PDF file before sending it or print it, as shown in Figure 2.

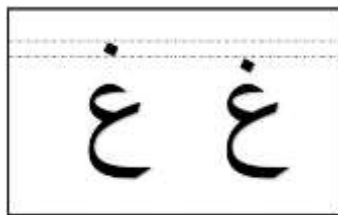


Figure 2. Arabic Character Vertical displacement of the points for the Arabic letter AEEN

2.2. Similar letters [15]

In 2009 use the similarity between the Arabic and Irani letters to hide the secret bit the Arabic and the Irani language have the similar tow letters (س,ي) it's the same shape but different (ASCCI) code when to put one of these letters in Arabic mood its hide bit with value (1) and in Irani mood its hide bit with value (0).

2.3. Kashida extensions [16, 17]

In 2010 manage the Arabic text steganography by using the property of the letter to embedded the secret message if the Arabic word used kashida (عـ) it is mean holding (0) and if it is clear (ع) that's mean embedding (1) and so on for all the cover text to hold the unread bits to make stego-text, or use one Kashida to hide (0) and tow Kashida to hide (1) until the secret message finish put (111111) to make the receiver understand all the secret.

2.4. Whitespaces [18]

In 2016 utilize the space between the word to hide a bit, first must calculate all the cover text words then decide how bits can be embedded in this cover, put space between the words mean it is hidden (1) else if there is no space mean holding (0).

2.5. linguistic and synonymy [19]

Linguistic steganography systems can be defined as methods of implementing practical tools for linguistic steganography. They are pertinent to the natural language processing (NLP) and fields of computational linguistics. As there are only a few implementations of linguistic steganography, there are no established criteria for linguistic steganography systems. However, the behind is the basic obligatory features.

2.6. Zero white space (ZWS) [20]

Is a non-printable or observable character used in computerized typesetting to show the locations of words and their location in word processing systems when using scripts that do not use clear spaces, or after special characters such as italics that do not follow a visible distance and can be After that there is a line break. It is also used in languages that do not have a visible distance between words, such as Japanese. It is naturally neither visible nor easy to observe, in the year 2018 (S. Tyagi, R. K. Dwivedi, and A. K. Saxena) made an algorithm to combine the use of PDF files with the effectiveness of (ZWS) to hide the secret text, where the algorithm converts the letters of the secret text to ASCII code and after you confirm the account The sum of the words of the plain text and matching its size with the size of the data to be hidden, the values of the secret text are stored inside a binary array to reduce the size of the secret text and then the process of concealing the bits by adding (ZWS) between the words for each bit holds a value (1) of the secret text to be hidden.

2.7. Different worlds language [21]

The world speaks more than 7000 languages for understanding and communication as the Indi language alone has more than 400 languages 22 of which are considered an official language, both (Vidhya P.M. and Varghese Paul) in 2015 have benefited from multilingualism to hide a secret message within a plain text to create a cover text using the local language (Malayalam). English letters and symbols are used as a coding system called (ASCII code) consisting of 8 bits while the rest of the languages use a coding system called (UNICODE) consisting of 16 bits, it creates Alphabet index matrices (AIM)for Indi language and (AIM) for English language, the secret text is used and converted to the language table English and then compare it to the financial language table to produce a text in the financial language that contains the secret text, as this algorithm is considered to have a high storage capacity compared to a few security due to errors appearing due to the difference in the two languages.

2.8. Homophone words [22]

One of the most important and oldest methods used to conceal similar words with writing and meaning is one of the most important and oldest methods used to hide confidential information as there are words that have the same word pronunciation but write differently and contain another meaning that is completely different from the original meaning, in 2016 (Shih-Yuhuang and Ping-Shenghuang) use the chat rooms to send encrypted texts in a simple way using Chinese language. The Chinese language being the largest number of letters and words contains approximately 54,648 letters and the simple dictionary contains 106 characters, so there is a great possibility to find words that are similar to pronunciation and different way to writing, the algorithm is based on the division of similar words in terms of different Writing and meaning to the first two groups are given the value of (1) and the second value (0) and send messages through social media, where the bits of the secret text are hidden inside the words and sent, the receiver opens the similar words and extracts the secret text a second time. This method is successful and effective by hiding secret texts.

2.9. Justification the text [23]

For typographic and aesthetic arrangement, the "justification" feature is used where the distances and words between the words and letters are extended, lengthened or compressed in order to align both the right and left sides of the text. In 2019 (B. Khosravi, B. Khosravi, B. Khosravi, and K. Nazarkardeh) made an algorithm to hide the secret text within the plain text by taking advantage of the "justification" feature, where the work of the algorithm is by using the Huffman method to compress the secret text to exploit the available space and the compressed text is divided into blocks and entered into the logical circle XOR and the result The blocks are divided into (4) bits and the resulting secret text is hidden by adding one blank in the case that the bit to be hidden holds a value (0), and two spaces in the case of the bit that holds a value (1), to produce a well-arranged text from the right ends The left, it is converted to PDF and sent without raising anything Cook about the content of the message.

3. COMPRESSION ALGORITHM

In this research, we suggest a way to compress the size of the secret message letters and symbols for the English language using Text 5 Bit Encoding (T-5BE) technology. For the purpose of coding letters in memory, we need 8 bits memory for each letter. The suggested idea is to reduce the number of bits per letter or symbol from 8 to 5 bits, this method works mainly with 79 letters and symbols for the English language in addition to the printable space that in the normal position on the keyboard where creating a table to represent 79 letters and symbols, each letter and symbol of them can be represented 5 bits instead of 8 bits and the value of 79 combinations of them (26) for letters The uppercase and (26) for lowercase letters and (26) for numbers and symbols plus space are divided Into three groups as in Table 3.

Table 3. The T-5BE

No.	Binary bit	5	Set -1	Set-2	Set-3	No.	Binary bit	5	Set -1	Set-2	Set-3
1	01000	A	a	0	17	10011	R	r	(
2	10000	B	b	1	18	11001	Q	q	!		
3	00011	C	c	2	19	11010	S	s	@		
4	00101	D	d	3	20	11100	T	t	#		
5	01001	E	e	4	21	10101	U	u	\$		
6	01010	F	f	5	22	10110	V	v	%		
7	00110	G	g	6	23	01111	W	w	&		
8	01100	H	h	7	24	10111	X	x	?		
9	10001	I	i	8	25	11011	Y	y	{		
10	10010	J	j	9	26	11101	Z	z	}		
11	11000	K	k	/	27	00000	Space	Space	space		
12	10100	L	l	*	28	00001	Set-1				
13	00111	M	m	+	29	00010		Set-2			
14	01011	N	n	-	30	00100			Set-3		
15	01101	O	o	=	31	11110	F.T.U.	F.T.U.	F.T.U.		
16	01110	P	p)	32	11111	END	END	END		

4. THE COMPRESSION ALGORITHM WORKING [24]

The secret message is read, one letter is taken after a letter, each letter enters the algorithm where it searches for its presence in one of the three groups as a capital letter or a small letter or a symbol or a number when knowing the group is written before writing the value of the letter where the totals are as follows

(00001) For the first group, (00010) for the second group, and (00100) or the third group, after which the value of the letter or symbol is written according to its location from the group, and in the case of the end of the secret text, the code (11111) is written to indicate the end of the secret text as showing in Figure 3.

The Compression algorithm technology [25]:

Input: secret text

Step 1: take the next character

Step 2: if the character from the same last group go to step 4

Step 3: print the group value

Step 4: print the character value

Step 5: if not end of the file go to step 1

Step 6: print (11111)

Step 7: END

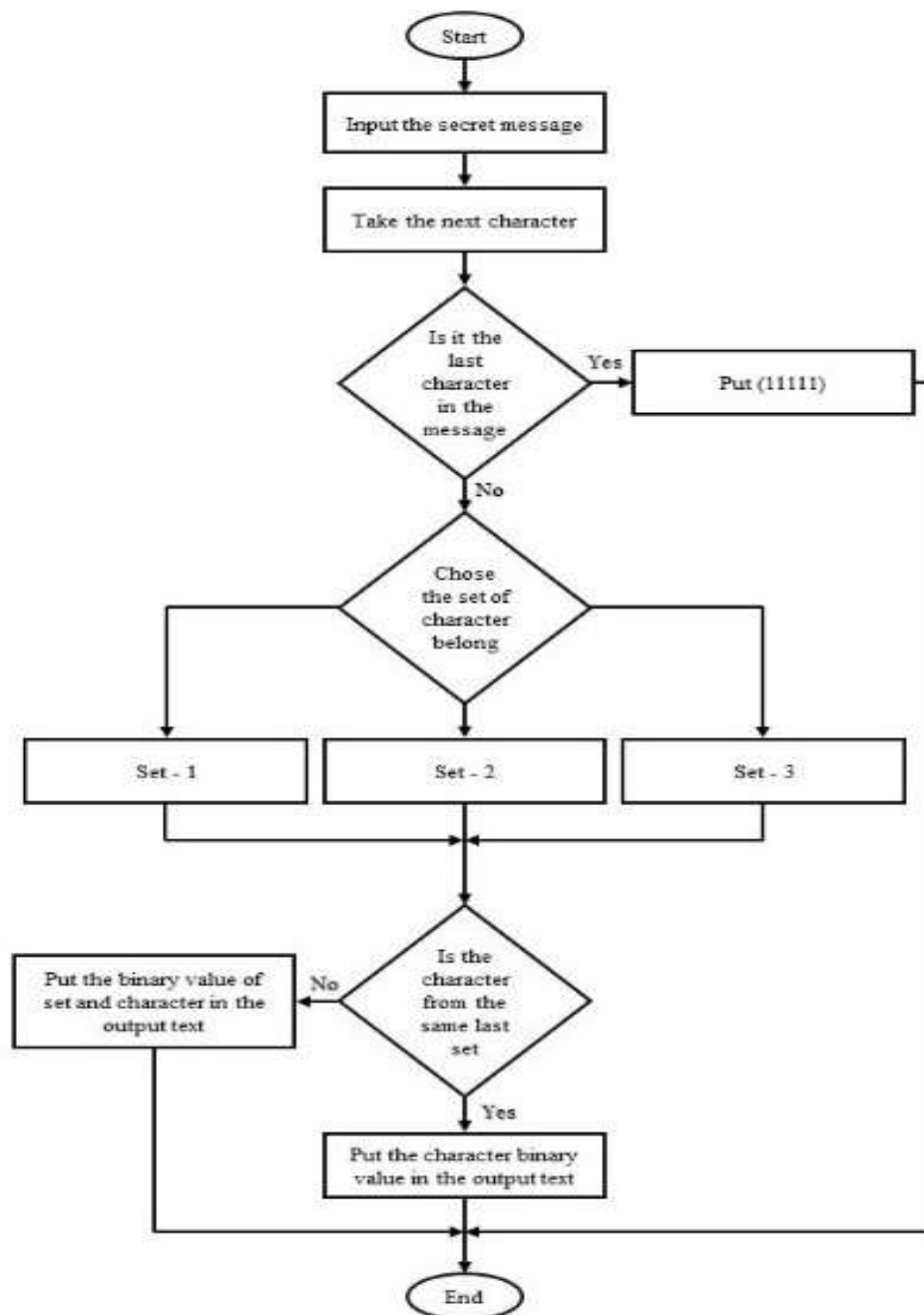


Figure 3. Flowchart of T-5BE algorithm

Example 1:

The secrete message “Text Steganography” in 5-Bit

$$\begin{aligned}
 \text{Text} &= \overbrace{00001}^{\text{set 1}} \overbrace{11100}^T \overbrace{00010}^{\text{set 2}} \overbrace{01001}^e \overbrace{10111}^x \overbrace{11100}^t \\
 &\quad \text{space} \\
 \text{Space} &= \overbrace{00000} \\
 \text{Steganography} &= \overbrace{00001}^{\text{set 1}} \overbrace{11010}^S \overbrace{00010}^{\text{set 2}} \overbrace{11100}^t \\
 &\quad \overbrace{01001}^e \overbrace{00110}^g \overbrace{01000}^a \overbrace{01011}^n \\
 &\quad \overbrace{01101}^o \overbrace{00110}^g \overbrace{10011}^r \overbrace{01000}^a \\
 &\quad \overbrace{01110}^p \overbrace{01100}^h \overbrace{11011}^y \\
 \text{END} &= \overbrace{11111}^{\text{END}}
 \end{aligned}$$

The output:

“00001111000010010011011111100000000001110100001011100010010011001000010110110100110100110011010000111001100110111111”

The secrete message “Text Steganography” in 8-Bit

$$\begin{aligned}
 \text{Text} &= \overbrace{01010100}^T \overbrace{01100101}^e \overbrace{01111000}^x \overbrace{01110100}^t \\
 &\quad \text{space} \\
 \text{Space} &= \overbrace{00100000} \\
 \text{Steganography} &= \overbrace{01010011}^S \overbrace{01110100}^t \overbrace{01100101}^e \\
 &\quad \overbrace{01100111}^g \overbrace{01100001}^a \overbrace{01101110}^n \\
 &\quad \overbrace{01101111}^o \overbrace{01100111}^g \overbrace{01110010}^r \\
 &\quad \overbrace{01100001}^a \overbrace{01110000}^p \overbrace{01101000}^h \\
 &\quad \overbrace{01111001}^y \\
 \text{END} &= \overbrace{01000101}^E \overbrace{01001110}^N \overbrace{01000100}^D
 \end{aligned}$$

The output:

"01010100011001010111100001110100001000000101001101110100011001010110011101100001011011100111011000010110111001110110011101110011011100110011001100001011100000110100001111001010001010100111001000100"

Text Steganography in 5Bit= 18 character +2*set1+2*set2+end= 115 bit

Text Steganography in 8Bit= 18 character + END = 168 bit

The compression percentages = 37.5%

5. THE PROPOSED TECHNIQUE

In our research, we tried to find a method of concealment that was easy to hide and transport, and at the same time, it was difficult to penetrate an unauthorized person. Where we took advantage of the dots property of the Arabic letters and the ability of the thesaurus to the Arabic language to hide many secret texts inside the plain text and send it easily with social media or e-mail without raising suspicion in the content of the message or raising suspicion as it carries a secret text hidden inside it.

As for hiding the secret text inside the plain text using Arabic alphabet points is one of the new methods in this field, and the possibility of concealing is done by converting the secret text to be hidden to the binary system and then inserting it with the compression algorithm T-5BE which is useful to reduce the size of the secret text by 37% which leads To increase the possibility of concealment about the plain text and after this process, the work of the concealment algorithm begins using the regular text to convert it to the cover of the text, and this algorithm works by taking a word after a word from the plain text and comparing it with the bit to be hidden where the sum of the litter dotes of the word is the basis of the process so if the number of litters dots in the word is equal to an odd number, which means that the word contains a secret bit

with a value of (0). and if the number of letter dots in the word is equal to an even number, which means that the word contains a secret bit with a value of (1). Contrasting to the amount of the secret bit to be hidden, the synonym will be resorted to where the word is replaced by another word that has the same meaning, but with some letter dots equal to the value of the bit to be hidden and in the case of not being able to find a word close to the meaning and containing the same number of dots for its letters to equal the value of the bit to be hidden, it is neutralized The word, that is, to find a word that does not contain its letters and continue with the algorithm until all the last bits of the secret text to be hidden are finished. If the secret text to be hidden may be over, five words containing letter points with an even number of letters will be placed to the last of the secret message to make the receiver understand the recipient with the end of the secret text as showing in Figure 4.

The embedding algorithm technology:

Input: secret text, plan text.

Output: Cover text

Step 1: Convert the secret message using T-5BE.

Step 2: Calculate the plan text words and compare them with secret message length.

Step 3: Check the secret bit if it's the end of the secret message print (11111) in cover text.

Step 4: Compare the hiding bit and the sum dot of the word cover.

Step 5: If there unmatching between the bit and the sum dot of the word cover use the Arabic Synonyms Dictionary to change the matching word and print it in the cover text.

Step 6: END

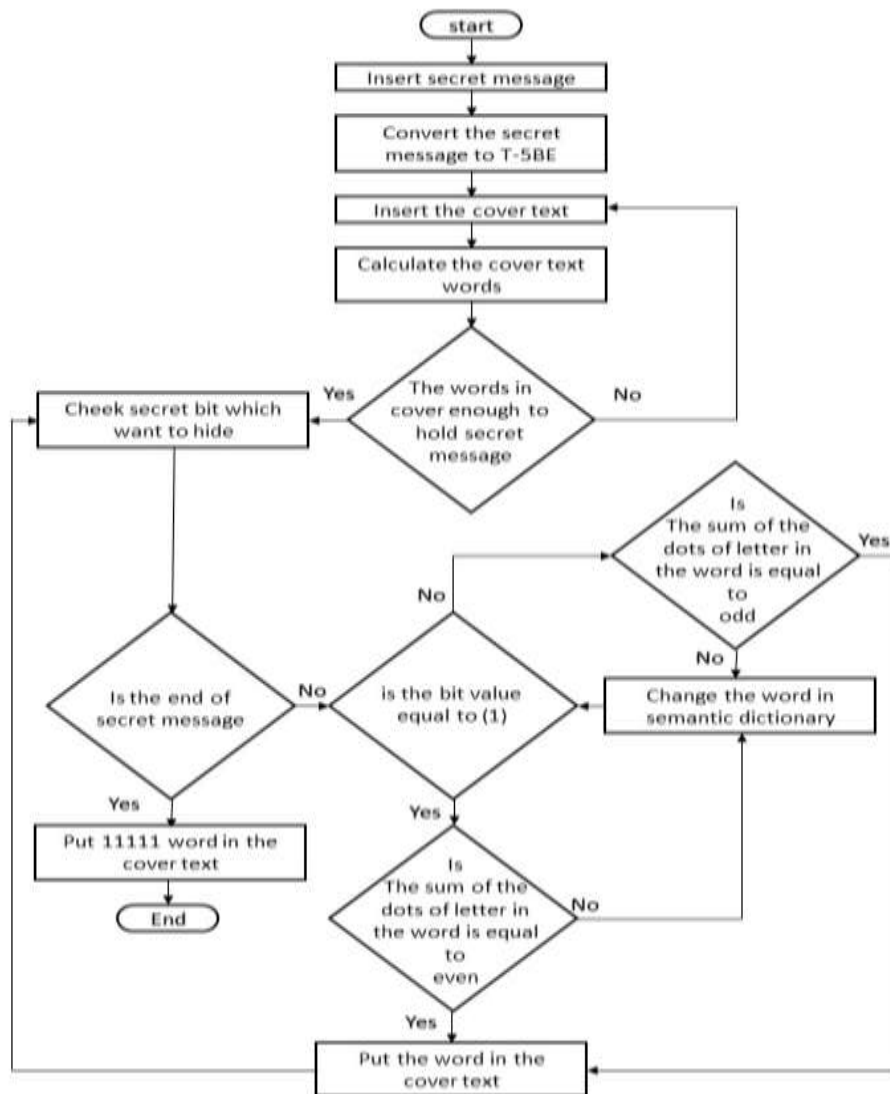


Figure 4. Hiding secret message Flowchart

The inversion hiding algorithm is as important as the concealment algorithm, as this algorithm relies on the cover text to extract the secret text hidden in it. Where this algorithm relies on pulling a word after a word from the cover text and then calculating the number of letters points in each word and then storing them in a matrix consisting of five cells where each of the matrices represents the value of one bit and then compare these five bits stored in the matrix in the table Special T-5BE To extract the existing character and insert it into a new text file, the algorithm returns the same previous steps to obtaining the matrix that contains five bits with a value (11111) where it represents the end of the hidden secret text inside the text cover as shown in Figure 5.

The Extracting algorithm technology;

Input: Cover Text.

Input: Secret Message

Step 1: make a matrix 1*5 called Matrix.

Step 2: Matrix = 0.

Step 3: Read the Cover Text word after Word.

Step 4: Increase Matrix (1).

Step 5: If the Matrix not equal (5) then step 7.

Step 6: If the sum of the dots of letter in the word is equal to odd put (1) in the Matrix else put (0) in the Matrix, Step 3.

Step 7: If the sum of the dots of letter in the word is equal to odd put (1) in the Matrix else put (0) in the Matrix.

Step 8: Convert the Matrix value to a letter from T-5BE.

Step 9: If the Matrix is not equal to (11111) then print the secret letter, Step 3.

Step 10: END



Figure 5. Extracting secret message Flowchart

6. EXPERIMENT RESULTS AND COMPARISONS

At this stage, a random text that speaks about the ancient history of Baghdad was chosen as a regular text and inserted into the algorithm to hide a secret text. The secret text was made up of eight words. The total number of its characters is 53 characters. It was successfully hidden without the text having any defects or doubts, as this algorithm showed The full ability to hide and the possibility of high work with nappy on the frequencies of letters according to what is present without producing distortions in the text lead to the possibility of detecting it by looking or electronic methods.

This example of hiding a secret message in Arabic plain text

“This is an example of embedded English secrete text”

8 words, 41 character, 7 space, 4 set, Total = 53 symbol, 106 bit with the set = 530 bits

<i>set1</i>	<i>T</i>	<i>set2</i>	<i>h</i>	<i>i</i>	<i>s</i>	<i>space</i>	<i>i</i>	<i>s</i>	<i>space</i>	<i>e</i>	<i>x</i>	<i>a</i>	<i>m</i>
00001	11100	00010	01100	10001	11010	00000	10001	11010	00000	01001	10111	01000	00111
<i>p</i>	<i>l</i>	<i>e</i>	<i>space</i>	<i>t</i>	<i>o</i>	<i>space</i>	<i>e</i>	<i>m</i>	<i>b</i>	<i>e</i>	<i>d</i>	<i>d</i>	<i>e</i>
01110	10100	01001	00000	11100	01101	00000	01001	00111	10000	01001	00101	00101	01001
<i>d</i>	<i>set1</i>	<i>E</i>	<i>set2</i>	<i>n</i>	<i>g</i>	<i>l</i>	<i>i</i>	<i>s</i>	<i>h</i>	<i>space</i>	<i>s</i>	<i>e</i>	<i>c</i>
00101	00001	01001	00010	01011	00110	10100	10001	11010	01100	00000	11010	01001	00011
<i>r</i>	<i>e</i>	<i>t</i>	<i>space</i>	<i>e</i>	<i>x</i>	<i>t</i>	<i>END</i>						
10011	01001	11100	00000	11100	01001	10111	11100	11111					

اظهرت الخرائط الموجودة أن بغداد العاصمة اثناء الازمان الماضية موطناً للعيش والازدهار حيث تعود إلى الحقبة الاشورية اثناء انتشائها كونها عاصمه للخلافة. سبب جعل هذا المكان لموقعه و أهميته توسطه مدناً انسانية و حضارية من الناحية الجغرافية والاستراتيجية حيث تمتد من شمال البلاد لتصل حتى مواقع الجنوب. يعد هذا المكان الجغرافي و التاريخي جعل مدينة مثل بغداد تتبوأ هذا الموقع المتوسط و المميز بين تلك الثقافات الامر الذي حقق بيئة جيدة لتكوين و خلق فكر لتطوير حياة الانسان و انجازاته العصرية، منذ العصور التي سبقت انتشار الفتح الاسلامي وما بعده، كانت المنطقة التي عرفت في بغداد بالعهد العباسي محاطه بها مدن كثيرة لها اهميتها العلمية و الموقعية الاستراتيجية و الجغرافية حيث تقع بغداد شمال بابل المدينة الاثرية، كما انها توجد على بعد ما مجموعه بعض من الالف الامتار اتجاه شمال غرب موقع قطيسفون التي تذكر الان باسم (المدائن)، حيث بناها من قبل الاغريق و ورثها الفرس و التي مازالت مستمرة كمركز للبلاد حتى اصبح محلها بغداد عندما حلة القرن الثامن بنيت في العصر العباسي بغداد كمدينة و اطلق اسم الزوراء و مدينة السلام و كانت ذات يوم عاصمه الدنيا و مقر الحكم العباسي حيث بناها في ذلك التاريخ الخليفة العباسي ابو جعفر المنصور و اقام لها ابواب اربعة الاولى كانت تسمى باب الشام و الثانية كان اسمها باب البصرة ثم الثالثة كان اسمها الكوفة و الرابعة باب خراسان او باب الدولة ، وكان الخليفة المنصور اختار لها الموقع بحنكة كون هذه البقعة من الارض على نهر دجلة تزخر بغداد بالعديد الزاخر من المعالم التاريخية القديمة و الحضارية مثل المدرسة المستنصرية و المساجد و كذلك القصور القديمة و العديد من الاماكن و المقامات العالية الروحانية اهمها مقام الامامين موسى الكاظم و مقام النعمان و مقام الجبلاني و جامع القشلة و جامع العباسي المعروف باسم جامع القصر قديماً أو الخليفة ، حيث ان ابو جعفر المنصور امر ببناء بغداد على رقعة مرتفعة و عالية بالجانب الغربي لجران تيار مياه نهر دجلة و التقائه بنهاية جريان تيار مياه نهر ارفيل فيه والتي بدأ العمل في بنائها بتاريخ الثالث والعشرين من تموز من عام سبعمئة وأثنان وستين ميلادية.

REFERENCE

- [1] R. Din, S. Utama, and A. Mustapha, "Evaluation review on effectiveness and security performances of text steganography technique," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 11, no. 2, pp. 747–754, 2018, doi: 10.11591/ijeecs.v11.i2.pp747-754.
- [2] A. Majumder, S. Changder, and N. C. Debnath, "A new text steganography method based on Sudoku puzzle generation," *Lect. Notes Electr. Eng.*, vol. 605, pp. 961–972, 2020, doi: 10.1007/978-3-030-30577-2_85.
- [3] A. Koluguri, S. Gouse, and P. B. Reddy, "Text steganography methods and its tools," *Int. J. Adv. Sci. Tech. Res.*, vol. 2, no. 4, pp. 888–902, 2014.
- [4] A. M. Hamdan and A. Hamarshah, "AH4S: an algorithm of text in text steganography using the structure of omega network," *Secur. Commun. Networks*, vol. 9, no. 18, pp. 6004–6016, 2016, doi: 10.1002/sec.1752.
- [5] S. F. Lu, O. Farooq, and H. Ali, "New steganography method using litter manipulations frequency," *DEStech Trans. Comput. Sci. Eng.*, no. icitia, pp. 224–229, 2017, doi: 10.12783/dtcse/icitia2017/13247.
- [6] M. Shirali-Shahreza and S. Shirali-Shahreza, "High capacity Persian/Arabic text steganography," *Journal of Applied Sciences*, vol. 8, no. 22, pp. 4173–4179, 2008, doi: 10.3923/jas.2008.4173.4179.
- [7] K. Wang and Q. Gao, "A coverless plain text steganography based on character features," *IEEE Access*, vol. 7, pp. 95665–95676, 2019, doi: 10.1109/ACCESS.2019.2929123.
- [8] S. Utama, R. Din, and M. Mahmuddin, "Critical analysis on steganography technique in text domain," *Proc. 5th Int. Cryptol. Inf. Secur. Conf. 2016, Cryptol. 2016*, 2016, pp. 150–157.
- [9] R. Kumar and H. Singh, "Recent trends in text steganography with experimental study," *Handb. Comput. Networks Cyber Secur.*, pp. 849–872, 2020, doi: 10.1007/978-3-030-22277-2_34.
- [10] H. Alshahrani and G. Weir, "Hybrid Arabic text steganography," *Int. J. Comput. Inf. Technol.*, vol. 6, no. 6, pp. 329–338, 2017.
- [11] S. M. A. Al-Nofaie and A. A. A. Gutub, "Utilizing pseudo-spaces to improve Arabic text steganography for multimedia data communications," *Multimed. Tools Appl.*, vol. 79, no. 1–2, pp. 19–67, 2020, doi: 10.1007/s11042-019-08025-x.

- [12] A. Gutub and K. Alaseri, "Hiding Shares of Counting-Based Secret Sharing via Arabic Text Steganography for Personal Usage," *Arab. J. Sci. Eng.*, 2019, doi: 10.1007/s13369-019-04010-6.
- [13] K. M. S. Foda, A. Fahmy, K. Shehata, and H. Saleh, "A Qur'anic Code for Representing the Holy Qur'an (Rasm Al-'Uthmani)," *Proc. - 2013 Taibah Univ. Int. Conf. Adv. Inf. Technol. Holy Quran Its Sci. NOORIC 2013*, 2015, pp. 304–309, doi: 10.1109/NOORIC.2013.67.
- [14] M. H. Shirali-Shahreza and M. Shirali-Shahreza, "A new approach to Persian/Arabic text steganography," *Proc. - 5th IEEE/ACIS Int. Conf. Comput. Info. Sci., ICIS 2006. conjunction with 1st IEEE/ACIS, Int. Work. Component-Based Softw. Eng., Softw. Arch. Reuse, COMSAR 2006*, vol. 2006, no. July 2006, pp. 310–315, 2006, doi: 10.1109/ICIS-COMSAR.2006.10.
- [15] M. H. Shirali-Shahreza and M. Shirali-Shahreza, "Arabic/Persian text steganography utilizing similar letters with different codes," *Arab. J. Sci. Eng.*, vol. 35, no. 1, pp. 213–222, 2010.
- [16] A. F. Al-Azawi and M. A. Fadhil, "Arabic text steganography using kashida extensions with Huffman code," *Journal of Applied Sciences*, vol. 10, no. 5, pp. 436–439, 2010, doi: 10.3923/jas.2010.436.439.
- [17] A. Gutub, W. Al-Alwani, and A. Mahfoodh, "improved method of Arabic text steganography using the extension Kashida" character," *Bahria University Journal of Information & Communication Technologies (BUJICT)*, vol. 3, no. 1, pp. 68–72, 2010.
- [18] S. Al-nofaie, M. Fattani, and A. Gutub, "Capacity improved Arabic text steganography technique utilizing 'Kashida' with whitespaces capacity improved arabic text steganography technique utilizing 'Kashida' with whitespaces related: kashida approach," *3rd Int. Conf. Math. Sci. Comput. Eng. (ICMSCE 2016) Capacit.*, no. February, pp. 38–44, 2016.
- [19] A. A. Hamzah, S. Khattab, and H. Bayomi, "A linguistic steganography framework using Arabic calligraphy," *Journal of King Saud University-Computer and Information Sciences*, 2019, doi: 10.1016/j.jksuci.2019.04.015.
- [20] S. Tyagi, R. K. Dwivedi, and A. K. Saxena, "A high capacity PDF text steganography technique based on hashing using quadratic probing," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 3, pp. 192–202, 2019, doi: 10.22266/IJIES2019.0630.20.
- [21] P. M. Vidhya and V. Paul, "A method for text steganography using malayalam text," *Procedia Comput. Sci.*, vol. 46, 2015, pp. 524–531, doi: 10.1016/j.procs.2015.02.079.
- [22] S. Y. Huang and P. S. Huang, "A homophone-based Chinese text steganography scheme for chatting applications," *J. Inf. Sci. Eng.*, vol. 35, no. 4, pp. 871–886, 2019, doi: 10.6688/JISE.201907_35(4).0010.
- [23] B. Khosravi, B. Khosravi, B. Khosravi, and K. Nazarkardeh, "A new method for pdf steganography in justified texts," *J. Inf. Secur. Appl.*, vol. 45, pp. 61–70, 2019, doi: 10.1016/j.jisa.2019.01.003.
- [24] A. A. Obeidat, "Arabic text steganography using Unicode of non-joined to right side letters," *J. Comput. Sci.*, vol. 13, no. 6, pp. 184–191, 2017, doi: 10.3844/jcssp.2017.184.191.
- [25] A. Mahmood and K. M. A. Hasan, "An efficient 6 bit encoding scheme for printable characters by table look up," In *2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 2017, pp. 468–472.