Text Steganography using Extensions Kashida based on the Moon and Sun Letters Concept

Anes.A.Shaker, Farida Ridzuan, Sakinah Ali Pitchay
Faculty of Science and Technology
Universiti Sains Islam Malaysia
Nilai, Negeri Sembilan, Malaysia

Abstract—**Existing steganography methods are still lacking in terms of capacity. Hence, a new steganography method for Arabic text is proposed. The method hides secret information bits within Arabic letters using two features, which are the moon and sun letters and the redundant Arabic extension character "-" known as Kashida. The Arabic alphabet contains 28 letters, which are classified into 14 sun letters and 14 moon letters. This classification is based on the way these letters affect the pronunciation of the definite article (ال) at the beginning of words. This method uses the sun letters with one extension to hold the secret bits '01', the sun letters with two extensions to hold the secret bits '10', the moon letters with one extension to hold the secret bits '00' and the moon letters with two extensions to hold the secret bits '11'. The capacity performance of the proposed method is then compared to three popular text steganographic methods. Capacity is measured based on two factors which are Embedding Ratio (ER) and The Efficiency Ratio (TER). The results show that the Letter Points and Extensions Method produces 24.91% and 21.56% as the average embedding ratio and the average efficiency ratio correspondingly. For the Two Extensions 'Kashida' Character Method, the results for the average embedding ratio and the efficiency ratio are 56.76% and 41.81%. For the Text Using Kashida Variation Algorithm method, the average embedding ratio and the average efficiency ratio are 31.61% and 27.82% respectively. Meanwhile, the average embedding ratio and the efficiency ratio for the Proposed Method are 61.16% and 55.70%. Hence, it is concluded that the Proposed Method outweighs the other three methods in terms of their embedding ratio and efficiency ratio which leads to the conclusion that the Proposed Method could provide higher capacity than the other methods.**

Keywords—Text steganography; Arabic text; extension Kashida; capacity

I. Introduction

Steganography can be described as the concealment of confidential messages implanted within other apparently regular messages, graphics or sounds [1]. Steganography is defined as the study of these invisible communications. Steganography deals with ways of hiding the existence of the communicated data in such a way that it remains confidential [2]. It maintains secrecy between two communicating parties. In text steganography, secrecy is achieved by embedding data into cover text and generating a stego-text. There are different types of steganography techniques and each has its own strengths and weaknesses. According to the medium used for the steganography, carrier files can be termed as cover text, cover images, cover audio, cover video or cover network [2]. The main drawback for all the methods is their low capacity wherein only a small amount of bits are allowed to be hidden. The lower the capacity of a method the bigger the carrier file must be to hide the secret message. Hence, a new method is proposed which use the concept of moon and sun letters with extension Kashida. Both the sun letters and moon letters are at the beginning of a word preceded by (ال) [3]. The Arabic language comes in two groups with each group consisting of 14 letters.

The objectives of this paper are: 1) to present a proposed work using the concept of moon and sun letters with extension Kashida, and 2) to present an evaluation study of the proposed work compared to three existing methods. The rest of this paper is organized as follows: related works are presented in Section II followed by an explanation of the proposed work in Section III. Section IV explains how the evaluation was carried out. Section V presents the results and a discussion of the results. Finally, the conclusion and future research suggestions are presented in Section VI.

II. Related Works

For the past few years, a lot of research has focused on the development and potential applications of Arabic script steganography.

A. Steganography Using Multiple Diacritics

An entire message can be hidden in a single diacritic mark by generating a number of extra-diacritic keystrokes equal to the binary number representing the message. For this scenario, consider this example of (110001)b as a secret message, the first diacritic is repeated 3 extra times (3 = (11)b); the second one, 0 extra times (0 = (00)b); and the third one, 1 extra time (1=(01)b) [4].

B. Word Spelling Method

The author presents a new text steganography method for hiding data in English texts. This method is based on substituting US and UK spellings of words. In English some words have different spelling in US and UK. For example "program" has a different spelling, in UK (program), and US (program). By using this feature, the author proposes his method for hiding data in an English text. In this method, the data is hidden in the text by substituting such words [5].

C. Vertical Displacement of the Points

This method makes use of dotted letters. Some language texts, which include Arabic and Persian, come with a substantial number of dotted letters. The Arabic text has 26 characters, of which 13 have dotted letters, while the Persian text has 32 characters, of which, 22 have dotted letters. With this method, ‘1’ is encoded to move up the point, or else ‘0’ is encoded. This process is replicated for the following dotted characters in the text as well as the following bits of information [6].

D. Mixed-case Font

The concept for this method was formed during an Internet search for popular fonts used for chatting and presentations. The author came across an innovative kind of font that can type capital and small letters in sequence. For instance, if one typed the word ‘software’, this word would appear as ‘SoFtWaRe’. Sometimes the size of the letters would differ, and at other times they would be of similar size. The authors developed an innovative text steganography technique for the transmission of confidential information using this newly-discovered font [7].

E. Move the Diacritic Up

The Arabic language comes with diacritics. The inclusion of these diacritics in most Arabic texts is usually non-obligatory and this study emphasized the employment of this characteristic. The vertical shifting of the diacritic is in accordance to the character. ‘Zero’ denotes no change, and ‘one’ denotes the increased distance between the letter and its diacritics [8].

F. Using “La” Word

This method is proposed based on the feature code using the “La” word. This word is obtained by connecting “Lam” and “Alef” letters into a single word. The hiding process is based on the existence of two forms of this word which are the special form “La” (“ لا”) which has a unique code and the normal form “La” (“لـا”) by inserting Arabic extension character between the “Lam” and “Alef” letters. The normal form “La” is used to hide bit zero while bit one is hidden using the special word [9].

G. Improved ‘La’ word

The authors proposed an enhanced method for the utilization of the “La” word which involved the use of a different Unicode of ‘Lam” and ‘Alef’ to exploit the ‘La’ word into both special and normal forms. This recommendation takes into account the fact that each letter comes with four dissimilar outlines depending on its location in the word [10].

H. Sharp-edges Method

This method exploits the sharp-edged Arabic characters for the concealment of confidential information. Keys are introduced to facilitate the positioning of the secret bit. The diverse number of sharp edges in Arabic characters enhances the concealment effectiveness of bits ‘1’ and ‘0’. The character with one sharp edge can conceal either secret bit ‘1’ or ‘2’. Concurrently, if the number of sharp edges is two, the possible bit location is 11, 10, 00 or 01 [11].

I. Using Letter Points and Extensions method

This process takes advantage of the fact that more than half the text letters in the Arabic language come with dots. While these dotted letters were loaded with the confidential bit “one”, the letters without dots were loaded with the confidential bit “zero”. As the confidential information needs to conform to the cover-text letters, not every letter is loaded with confidential bits. Other than the letters used to indicate the particular letters containing the confidential bits, redundant Arabic extension characters are also included in the system. The advantage that comes with letter extensions is the fact that their utilization does not affect the writing content in any way [12].

J. Two-extension ‘Kashida’ Character

This paper presents a novel steganography method useful for Arabic and other similar languages. This method benefits from the feature of having the Kashida character, “ـ” in Arabic script. An extension character is inserted after a letter in the cover object if the secret bit is ‘zero’. Instead, if the secret bit is ‘one”, two consecutive extension characters will be inserted [13].

K. Enhanced Kashida

The author utilized the Kashida by encoding the original text document with Kashida according to a specific key which was produced before the encoding process. Kashida are inserted before a specific list of characters {ذ _ د_ؤ _و_أ _ا} until the end of the key is reached where the kashida is inserted for a bit 1 and omitted for a bit 0. This process is repeated until the end of the document is reached in a round robin fashion [14].

L. Text Using Kashida Variation Algorithm (KVA)

Most of the previous methods apply the same procedure for the whole text which may allow steganalysis to study the text format, hence, to breaking the code or, in other words, find the hidden message. However, this study proposed a method to apply four scenarios randomly to improve data privacy.

The method presents four scenarios. The first scenario is by adding Kashida after pointed letters to be encoded as one, otherwise, it is encoded as zero. The second scenario is by adding Kashida after nonpointed letters to be encoded as one, otherwise, it is encoded as zero. The third scenario is by adding Kashida after letters to be encoded as one. Otherwise, it is encoded as zero. The fourth scenario is by adding Kashida after letters to encode as zero. Otherwise, it is encoded as one. This method provides a high embedding ratio as it allows bits to be encoded in four different scenarios [15].

In summary, most of diacritics-based methods are simple to implement and provide higher capacity and robustness than others [10]. However, these methods cannot be used in text, in which, the appearance of all diacritics is important, like the Holy Quran. Conversely, Kashida-based methods provide good capacity [10] and could be used in printed documents with different font formats. However, they are easily detected or observed. Thus, many researchers add more security features to decrease the number of Kashidas and enhance the capacity [10]. On the other hand, shifting line, word or points methods are simple to implement however, their drawback is the high probability of destroying the watermark when retyping or printing [10]. Another weakness is that they are also noticeable by Optical Character Recognition (OCR) programs [10].

III. Proposed Work

A new method that could provide higher capacity is needed to improve the implementation of steganography generally. The proposed work hides the message in Arabic text using the characteristics of Arabic language. In Arabic language, there are two groups of letters, namely sun letter (solar letters) and moon letter (lunar letters). The secret text is hidden in the form of zeros and ones represented by the 16-bit Unicode for each character (the UTF-8 encoding scheme uses 16 bits to represent one Arabic character). Table 1 consists of the letters classifications.

TABLE I. Moon and Sun Letters

Moon letter				**Sun letter**			
1	أ	**8**	خ	**1**	ت	**8**	ش
2	ب	**9**	ف	**2**	ث	**9**	ص
3	غ	**10**	ع	**3**	د	**10**	ض
4	ح	**11**	ق	**4**	ذ	**11**	ط
5	ج	**12**	ي	**5**	ر	**12**	ظ
6	ك	**13**	م	**6**	ز	**13**	ن
7	و	**14**	هـ	**7**	س	**14**	ل

The proposed method presents four scenarios. The first scenario is implemented by adding a Kashida after a sun letter to represent (00). The second scenario is implemented by adding two kashidas after a sun letter to represent (11). In the third scenario, a kashida is added to represent (01) after a moon letter. The fourth scenario is implemented by adding two kashidas after a lunar letter to represent (10). The pseudo code of the new proposed method is presented in Fig. 2.

IV. Evaluation

The proposed work is evaluated and compared with three other methods [12]-[14]. The first method uses the letter points and extensions method [12]. The second method is the two extension "Kashida" character and the third method is the frequency recurrence of characters. The main aim of the research is to improve the steganography in terms of capacity. Capacity is determined by the embedding ratio and the efficiency ratio features.

Equation (1) is used to calculate embedding ratio (ER).

$$\mathrm{ER} = \frac{\mathit{Total\ letters\ of\ cover\ text} - \mathit{Letters\ of\ embedded\ message}}{\mathit{Total\ letters\ of\ cover\ text}} \quad (1)$$

The efficiency ratio (TER) is computed following (2).

$$\mathrm{TER} = \frac{\mathit{Total\ letters\ of\ cover\ text} - \mathit{Letters\ of\ embedded\ message}}{\mathit{Total\ letters\ of\ cover\ text}} \quad (2)$$

```
Algorithm Encode
Input: Cover text denoted by $C_t$
        Binary secret message denoted by $m_1, m_2, .... m_n$
        where n= the message size
Output: Stego_Object denoted by S
Step 1: S=$C_t$
Step 2: for i=1 to n step 2
        if(($m_i$ = 0 and $m_{i+1}$ = 0) or ($m_i$ = 1 and $m_{i+1}$ = 1))
              Search S to get first Sun letter location k
              if($m_i$ = 0 and $m_{i+1}$ = 0)
                     Insert one Kashida in location k+1
              else
                     Insert two Kashida in location k+1
              end if
        else
            Search S to get first Moon letter location k
             If($m_i$ = 0 and $m_{i+1}$ = 1)
                     Insert one Kashida in location k+1
              else
                     Insert two Kashida in location k+1
             end if
        End if
End for
Step 3: output S
```

```
Algorithm Decode
Input: Stego_Object denoted by S
Output: Binary secret message denoted by  $m_1, m_2, ... m_n$ where
n equal the message size
i=1
k=1
while k <=  S_length
Begin
   j=1
   if the $S_k$ letter is  Sun letter
   Begin
      if ($S_{k+j}$ = Kashida and $S_{k+1+j}$ = Kashida)
      Begin
          $m_i$=1 and mi+1=1
else mi=0 and mi+1=0
end if
else
             if (Sk+j = Kashida and Sk+1+j = Kashida)
 Begin
  mi=1 and mi+1=0
else mi=0 and mi+1=1
end if
 end if
          i=i+1
          j=j+1
      End while
Output Binary secret message m
```

Fig. 1. Pseudo code and decode of the proposed method.

The evaluation is carried out similarly to [6]. Ten cover texts are selected from highly circulated Iraqi newspapers [6]. The word "GOOD" is used to be embedded as secret text in the cover texts. The source of the cover text is presented in Table 2.

TABLE II. COVER TEXT SOURCES

No	Cover text
1	www.almadapaper.net/3784
2	www.almadapaper.net/2440
3	www.almadapaper.net/3000
4	www.almadapaper.net/3001
5	www.almadapaper.net/3010
6	www.almadapaper.net/3100
7	www.almadapaper.net/2001
8	www.almadapaper.net/2010
9	www.almadapaper.net/2111
10	www.almadapaper.net/2054

V. RESULTS AND DISCUSSION

An example of embedding the word ‘GOOD” in Cover Text 1 using the four related methods for comparison are presented in Table 3.

TABLE III. EMBEDDING RESULTS

Methods	The Cover texts after imbedding
A Novel Arabic Text Steganography Method Using Extensions [12]	ولـكـنَّ اللهجـة الـعـاميـة طغـت على ألسنِ النـاسِ، حتَّـى أنَـتّـا وجدنا الـبـعض ممن هو من بـنـي جـلدتـنـا قد دعـا البـعض إلى إلغـاء الـتـكلم بـاللغـة الفصـحـى.
Steganography in Arabic text using Kashida variation algorithm method [14]	ولكنَّ اللهجة العامية طغـت على ألسنِ الناسِ، حتَّى أنَـنـا وجدنا الـبعض مـمـن هو مـن بني جلـدتـن قـد دعـا الـبعض إلـى إلغاء الـتـكـلـم باللغة الفصحى.
Improved Method of Arabic Text Steganography Using the Extension “ Kashida ” Character [13]	ولـكـنَّ الــلـهـجــة الــعــامـيــة طـغـت عــلـــى ألــســنِ الــنــاسِ، حــتَّـــى أنَــتّـــا وجـــدنـــا الـبـعـض مــمـن هو من بني جلدتنا قد دعا البعض إلى إلغاء التكلم باللغة الفصحى.
Proposed method	ولكنَّ الـلــهــجــة الـعــامــيـــة طـغـت عـلــى ألــسنِ الناسِ، حتَّى أنَّنا وجدنا البعض ممن هو من بني جلدتنا قد دعا البعض إلى إلغاء التكلم باللغة الفصحى.

The calculation of Embedding Ratio (ER) and the efficiency ratio (TER) are presented in Table 4.

TABLE IV. ER AND TER RESULTS

Methods	Capacity Evaluation	
	Average ER	Average TER
A Novel Arabic Text Steganography Method Using Extensions [12]	24.91%	21.56%
Steganography in Arabic text using Kashida variation algorithm method [14]	31.61%	27.82%
Improved Method of Arabic Text Steganography Using the Extension “ Kashida ” Character [13]	56.76%	41.81%
Proposed method	61.16%	55.70%

Table 4 shows the average embedding ratio and the efficiency ratio results for letter points and the extensions method as 24.91% and 21.56%, respectively. The results for this method are pretty low because it is based on the concept of pointed letters. The existence of sentences without pointed letters could have a high impact on the capacity performance of this method. For Kashida Variation Algorithm method, the results for average embedding ratio and the efficiency ratio are 31.61% and 27.82%, respectively. The results are low because this method is also based on the concept of pointed letters. Similarly, as in the previous method, reliance on pointed letters in the sentence could have an effect on its capacity performance. For the two extension “Kashida” character methods, the results for the average embedding ratio and the efficiency ratio are 56.76% and 41.81%, respectively. The results for this method are considered good because it was developed based on the concept of adding Kashida after any letter. Hence, this method does not rely on certain characteristics possessed by any letter in the cover text. The proposed work results for the average embedding ratio, and the efficiency ratio, are 61.16% and 55.70%, respectively. This method produces higher results compared to the others due to its chosen features. The first features which are the moon and sun letters allow secret bits to be hidden in any letter as all Arabic words will contain either moon or sun letters. In addition, the proposed method allows two secret bits to be hidden in a letter. Thus, more secret bits can be hidden in shorter sentences.

VI. CONCLUSION AND FUTURE WORK

This paper presents a novel steganography method useful for Arabic language electronic writing using extension Kashida based on the concept of moon and sun letters. The proposed method uses sun letters with Kashida to represent (01), sun letters with two Kashidas to represent (10), moon letters with Kashida to represent (00) and moon letters with two Kashidas to represent (11). Kashida characters are used beside the Arabic letters to note which specific letter is holding the hidden secret bits. Letter extension is used as it will not affect the writing content. The proposed method outweighs the other three methods because of its capacity performance results. It can be concluded that choosing the right features to hide secret text is critical in determining the capacity performance of a steganography method. The advantage of implementing the moon and sun letters concept is that it is able to increase the probability of hiding the secret bits in any letter. Nonetheless, it is also very important to maintain the imperceptibility aspect while improving capacity. In future, this method will be evaluated in terms of its imperceptibility.

REFERENCES

[1] A. Siper, R. Farley, and C. Lombardo, “The Rise of Steganography,” Proc. Student/Faculty Res. Day, CSIS, Pace Univ., pp. 1–7, 2005.

[2] J. Kour and D. Verma, “Steganography Techniques –A Review Paper,” Int. J. Emerg. Res. Manag. &Technology, vol. 9359, no. 35, pp. 2278–9359, 2014.

[3] عبد السلام محمد هارون. "قواعد الاملاء." مكتبة الانجلومصرية، 1993.

[4] A. Gutub, Y. Elarian, S. Awaideh, and A. Alvi, “Arabic Text Steganography Using Multiple Diacritics,” no. May 2016, 2008.

[5] M. Shirali-Shahreza, “Text steganography by changing words spelling,” Int. Conf. Adv. Commun. Technol. ICACT, vol. 3, no. March, pp. 1912–1913, 2008.

[6] 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, pp. 310–315, 2006.

[7] A. Ali and A. Saad, "New Text Steganography Technique by using Mixed-Case Font," Online J. Comput. Sci. Inf. Tehcnology, vol. 3, no. 2, pp. 138–141, 2013.

[8] A. Odeh and K. Elleithy, "Steganography in arabic text using full diacritics text," 25th Int. Conf. Comput. Appl. Ind. Eng. CAINE 2012 4th Int. Symp. Sens. Netw. Appl. SNA 2012, no. November, 2012.

[9] M. Shirali-shahreza and M. H. Shirali-shahreza, "An Improved Version of Persian / Arabic Text Steganography Using ' La ' Word," no. August, pp. 26–27, 2008.

[10] R. A. Alotaibi and L. A. Elrefaei, "Arabic Text Watermarking : A Review," Int. J. Artif. Intell. Appl., vol. 6, no. 4, pp. 01–16, 2015.

[11] N. A. .KBM//Roslan, R. Mahmod, and N. I. Udzir, "Sharp-edges method in Arabic text steganography," J. Theor. Appl. Inf. Technol., vol. 33, no. 1, pp. 32–41, 2011.

[12] W. Al-Alwani, A. Bin Mahfooz, and A. A. A. Gutub, "A Novel Arabic Text Steganography Method Using Extensions," Proceeding World Acad. Sci. Eng. Technol., vol. 1, no. 3, pp. 483–486, 2007.

[13] A. Gutub, W. Al-Alwani, and A. Mahfoodh, "Improved Method of Arabic Text SteganographyUsing the Extension „Kashida‟ Character," Bahria Univ. J. Inf. , vol. 3, no. 1, pp. 68–72, 2010.

[14] Y. M. Alginahi, M. N. Kabir, and O. Tayan, "An Enhanced Kashida-Based Watermarking Approach for Increased Protection in Arabic Text-Documents Based on Frequency Recurrence of Characters," no. 5, pp. 381–392, 2014.

[15] A. Odeh, K. Elleithy, and M. Faezipour, "Steganography in Arabic text using Kashida variation algorithm (KVA)," 9th Annu. Conf. Long Isl. Syst. Appl. Technol. LISAT 2013, no. September 2013, 2013.