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

Anees 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 “ـ” 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 (ٍ) 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 (ٍ) [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 (ٍ = (11)b); the second one, 0 extra times (ٍ = (00)b); and the third one, 1 extra time (ٍ = (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 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’.

J. Two-extension ‘Kashida’ Character

This method is proposed based on the feature code using “La” word. This method was produced before the encoding process. Kashida are inserted before a specific list of characters \{١١١١١١١١١١١١١١\} 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].

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 \{١١١١١١١١١١١١١١\} 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>
<thead>
<tr>
<th>TABLE I. MOON AND SUN LETTERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moon letter</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
</tbody>
</table>

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).

\[
ER = \frac{\text{Total letters of cover text} - \text{Letters of embedded message}}{\text{Total letters of cover text}}
\]  

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

\[
\text{TER} = \frac{\text{Total letters of cover text} - \text{Letters of embedded message}}{\text{Total letters of cover text}}
\]

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.
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.

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

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.

\\begin{table}[h!]
<table>
<thead>
<tr>
<th>No</th>
<th>Cover text</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><a href="http://www.almadapaper.net/3784">www.almadapaper.net/3784</a></td>
</tr>
<tr>
<td>2</td>
<td><a href="http://www.almadapaper.net/2440">www.almadapaper.net/2440</a></td>
</tr>
<tr>
<td>3</td>
<td><a href="http://www.almadapaper.net/3000">www.almadapaper.net/3000</a></td>
</tr>
<tr>
<td>4</td>
<td><a href="http://www.almadapaper.net/3001">www.almadapaper.net/3001</a></td>
</tr>
<tr>
<td>5</td>
<td><a href="http://www.almadapaper.net/3010">www.almadapaper.net/3010</a></td>
</tr>
<tr>
<td>6</td>
<td><a href="http://www.almadapaper.net/3100">www.almadapaper.net/3100</a></td>
</tr>
<tr>
<td>7</td>
<td><a href="http://www.almadapaper.net/2001">www.almadapaper.net/2001</a></td>
</tr>
<tr>
<td>8</td>
<td><a href="http://www.almadapaper.net/2010">www.almadapaper.net/2010</a></td>
</tr>
<tr>
<td>9</td>
<td><a href="http://www.almadapaper.net/2111">www.almadapaper.net/2111</a></td>
</tr>
<tr>
<td>10</td>
<td><a href="http://www.almadapaper.net/2054">www.almadapaper.net/2054</a></td>
</tr>
</tbody>
</table>

\end{table}


