



Text steganography: enhanced character-level embedding algorithm using font attribute with increased resilience to statistical attacks

Shreya Narasimhan KL¹ · Bala Krishnan R¹

Received: 17 December 2022 / Revised: 8 March 2024 / Accepted: 18 April 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

Text steganography, the science of hiding secret messages in innocent-looking text documents ensures the secrecy of the embedded secret. Cryptography, on the other hand, encrypts and converts the secret message into an unintelligible form ensuring confidentiality, integrity, and authentication of the message. It is to be noted that both steganography and cryptography are expected to work together hand-in-hand in order to attain a better security. Hence, a methodology that combines techniques from both steganography and cryptography is an unavoidable one for the complete protection of sensitive data. This paper aims to achieve one such methodology. The best existing method which works in similar direction is observed to use character spacing feature of text documents to embed the secrets. The method is reported to attain an average embedding capacity of ≈ 2 bits/cover-character and 8-bits/distortion with high imperceptibility and considerable cryptographic security. However, during experimentation, it is noticed that the methodology has the potential vulnerability of leaving clues for attackers with a probability of 0.51% which paves way for possible statistical attacks namely frequency analysis. This vulnerability occurred because of the constant left circular shift operation that is performed after embedding each secret character. This paper aims to address this vulnerability by proposing a novel modification that facilitates the methodology to perform both left and right circular shifts in an unbiased manner, based on the embedded secret, without affecting the notable achievements of the existing methodology. The proposed modification is successfully implemented, and further analysis indicated that the performed modification has in fact embedded the secret characters uniformly thereby reducing the standard deviation by 24%. This increased the resilience of the extended method to potential attacks that rely on statistical techniques.

Keywords Text steganography · Information hiding · Secrecy · Frequency normalization set · Covert communication · Character-level embedding

✉ Bala Krishnan R
balakrishnan@nitt.edu; balakrishnanr1987@yahoo.co.in

¹ Department of Computer Science and Engineering, National Institute of Technology Tiruchirappalli, Tiruchirappalli, Tamil Nadu, India

1 Introduction

In this digital era, almost all information is being generated/created, stored and shared in digital form [1]. Depending on importance, some information necessitates restricting their access from third-parties/malicious-attackers. In most scenarios, it is infeasible for many to physically restrict the data to secure, air-gapped storage devices with 24/7 monitoring. As a result, most data must be protected through techniques that are designed to circumvent potential passive and active attacks. Considering this, organizations employ a variety of information security methodologies to protect leakage of such data while in storage and in transit [2]. Two important fields of research which facilitates to achieve this sort of protection are namely cryptography and steganography.

Cryptography focuses on protecting the data/information by transforming them into an unintelligible form through encryption. The main aspects of cryptography include only confidentiality, integrity, authentication, and non-repudiation, but not secrecy [3]. This facilitates any third party to easily differentiate encrypted information from the other. Once identified, one can use the recent advances in cryptanalytic techniques and the advent of high-performance computation to decrypt and read them at ease. This act emphasizes the importance of secrecy in information security field in order to avoid such unnecessary attention of attackers.

Steganography, on the other hand, focuses on hiding secret data in innocent-looking cover media like text, image, audio, video, network packets, etc. It not only hides the content of secret message but also its existence altogether [4]. It takes advantage of the redundant information present in such media files and performs imperceptible modifications as a way to embed the secrets [5]. This ensures the concealment of such modifications from the preying eyes of attackers. However, one should not immediately conclude that steganography is superior to cryptography. Both are supposed to work together hand-in-hand. Often security professionals advice to use both cryptographic and steganographic techniques as a combination, in order to achieve better security during transmission [6]. Considering this, we aim to develop a novel steganographic technique with considerable cryptographic strength for sensitive organizations.

Of the available steganographic types, techniques that employ text documents as cover media (known as text steganography) is the most appropriate choice in the case of organizations. This is because of: (i) High volume traffic of text documents being exchanged among them [7]; (ii) Text documents have low storage size and thereby require low bandwidth during transmission. These characteristics allow for faster transmission and thereby make it difficult for attackers to differentiate and identify the secret embedded files, and conduct attacks on-the-fly.

Text steganography is not without its challenges. The primary one is the low quantity of redundant information (to use for embedding), when compared with other media formats like images¹ [8]. Thereby, even slight modification in the structure of text documents will lead to visual changes making it evident to detect [9]. Hence, utmost care must be taken while performing modifications so as to not rouse any suspicion. The secondary challenge is that the documents used as cover media must be of similar size to other documents

¹ For example, the presence of sheer number of color pixels in images allows steganographers to modify specific pixel contents without detection.

Table 1 Sample Character and String Mapping [10]

Mapped Character	Frequency Normalization Set (FNS)	Cumulative Frequency	Mapped Character	Frequency Normalization Set (FNS)	Cumulative Frequency
A	WRZFOIN	23.9587	O	MNSLAB.	24.5851
B	F□CWQXP	26.1824	P	TVIZM.A	24.1936
C	JMFXY□Z	25.8691	Q	SGKPHEM	25.4306
D	XA.MNTU	25.0548	R	HEUVIYJ	24.9452
E	YCHIJOS	24.8356	S	IYLNIAK	25.1018
F	KUX□ZJC	25.3523	T	OFJGCDE	24.5694
G	□.BCPQX	25.2272	U	VDEHKLY	25.0079
H	BLOSWMI	25.4777	V	LONQSUR	25.1018
I	PJGALHO	23.9744	W	QP□KXRV	26.7617
J	NSVOBGH	25.0079	X	EZDYGPT	25.3054
K	UWY.TSL	23.0505	Y	.KQUVZ□	26.0101
L	DIREFCQ	25.3367	Z	AXMTRVD	24.8356
M	GHTR.WB	22.753	□	RTWBDNF	24.9766
N	ZBADEFG	25.0392	.	CQP□KW	26.0571

where, “□” represents “space”; “.” represents “dot”

that are normally transmitted by the organization.² This clearly signifies that the number of characters available for embedding is limited. These necessitates that the text steganographic algorithms: (i) Must carefully choose the attributes that are to be modified in order to embed the secrets; (ii) Reduce the number of distortions/modifications by increasing the number of secret bits that are being embedded per modification.

On further study, it was noticed that the best existing methodology which works in a similar direction (steganographic technique with cryptographic strength) generates a Frequency Normalization Set (FNS) comprising of 28 strings using the individual occurrence frequency of English Alphabets, Dot and Space characters (refer Table 1) [10]. The specialty of FNS is that: (i) The length of each string is 7; (ii) A character can occur only once in a string and at a position of all the strings. That is, no row-wise and position-/column-wise repetitions; and (iii) The cumulative occurrence frequency of all the characters in each string sums to a value which is approximately equal to 25 (refer Table 1). As a result, each character gets a unique position in each string and in each position of all the strings. This facilitates it to act as a key during the embedding process. The strings are then mapped to the 28 possible secret characters (English Alphabets, Dot and Space) to obtain the Character and String Mapping (CSM). The CSM is then used to embed the secret characters in the cover document in a serial manner. That is, for each secret character the string that is being mapped in CSM is first identified. The first occurrence of any of the characters of identified string is then searched and marked in the cover document in a serial manner. The marking is done by altering the spacing between the characters. The spacing is either condensed by {0.1, 0.2, 0.3} points or expanded by {0.1, 0.2, 0.3, 0.4} points based on the position of the encountered cover character in the identified string of CSM.

An illustration to embed a secret character (say “O”) in cover work is provided in Table 2 for better understanding. It is to be noticed that, after embedding each secret

² For example, requiring a document comprising of 5000 pages as a cover media will raise suspicion in any type of organization.

Table 2 Procedure to embed the secret character “O” in cover work

Select the String from CSM that is Mapped to “O”		Search for the Occurrence of any One of the Characters from the Selected String in Cover Work	Position of the Identified Cover Character in the Selected String	Modification to be Done to the Selected Character in Cover Work
MNSLAB.	M		0	Condense the identified cover character spacing by 0.1 pt
	N		1	Condense the identified cover character spacing by 0.2 pt
	S		2	Condense the identified cover character spacing by 0.3 pt
	L		3	Expand the identified cover character spacing by 0.1 pt
	A		4	Expand the identified cover character spacing by 0.2 pt
	B		5	Expand the identified cover character spacing by 0.3 pt
	.		6	Expand the identified cover character spacing by 0.4 pt

where, *CSM* represents Character and String Mapping; “.” represents “dot”

character, all the strings in CSM are constantly left circular shifted by one position. This avoids the possibility of embedding the same secret character in the same cover character with the same character spacing even when they occur consecutively. The notable advantage of this methodology is that it achieves a high embedding capacity³ of ≈ 2 -bits per cover character. Besides, it requires lesser number of modifications as it can embed 8-bits in a distortion and is also resilient against the well-known statistical frequency analysis attacks [3].

However, we observed that this methodology has the following three limitations: (i) Secret messages are restricted to English Alphabets (not case sensitive), Dot and Space characters; (ii) Methodology can be adopted only by the languages which have varying letter frequency; (iii) The constant left circular shift by one position after embedding each secret character leaves clues for attackers to identify few characters of the embedded secret, specifically when the same character occurs at a constant distant (of 7) in the secret message and is embedded in the same cover character. It can be easily understood that the first two limitations are inevitable as they are directly related to the method by which the FNS is being generated [10]. Also, they do not pose any security threat as they do not leak any information. However, the third limitation is directly related to the way the rotation is being performed after embedding each secret character which is a serious security threat.

It is noticed that, in an ideal case, the occurrence probability of two characters appearing at a constant distant (of 7) in the secret message to be similar is 3.571% and the probability of embedding them in the same cover character with the same character spacing is 0.51%⁴ (refer Eqs. 1 and 2).

$$\begin{aligned}
 & \text{Probability of occurrence of first and eighth characters to be same} \\
 &= \frac{\text{Number of strings with first and eighth characters to be same}}{\text{Total number of possible strings}} \quad (1) \\
 &= \frac{28^7}{28^8} = 3.571\%
 \end{aligned}$$

$$\begin{aligned}
 & \text{Probability of embedding the same secret character in the same cover character with the} \\
 & \quad \text{Total number of strings with first and eighth characters to be} \\
 & \quad \text{same} \\
 & \text{same character spacing} = \frac{\text{Total number of strings with first and eighth characters to be same}}{\text{Total number of possible strings}} \\
 &= \frac{28^7 \times 7^7}{28^8 \times 7^8} = 0.51\% \quad (2)
 \end{aligned}$$

In this work, we aim to address this limitation by extending the existing methodology which splits the CSM into two groups (namely Group1 and Group2) with minimal difference in their cumulative occurrence frequencies of characters, in the respective groups⁵ (refer Table 3). This ensures that the operations performed on the basis of these groups remain unbiased, and thereby reduce the possible vulnerability to statistical attacks. The

³ Embedding capacity is a measure of the maximum size of the secret that a chosen cover medium can hide [10].

⁴ Though the embedding probability appears to be less, it should be noticed that the calculation is performed by considering the occurrence probability of all characters are uniform which is against reality [10]. In such case, the probability will be higher which further increases the threat.

⁵ During experimentation, it was observed that a large number of such groups exist. Hence, the groups can also be considered as a key providing one more additional layer of security.

Table 3 Grouping of characters

Group1 (Perform <i>Left Circular Shift</i> on all strings in CSM)		Group2 (Perform <i>Right Circular Shift</i> on all strings in CSM)	
Character	Frequency	Character	Frequency
Space	20.2944	e	9.6304
a	6.8431	t	7.5634
s	5.0579	o	6.3107
n	4.8387	i	5.4494
r	3.5703	h	4.9640
u	2.2706	d	3.9305
m	2.0827	l	3.0066
f	1.5033	w	2.2080
b	1.3624	g	1.6912
c	1.2058	y	1.5503
v	0.7830	Dot	1.3937
z	0.0783	k	1.1431
x	0.0626	p	0.8613
q	0.0470	j	0.2975
Total	50.000	Total	50.000

idea here is that the CSM will be left circular shifted by one position when the embedded secret belongs to Group1 and right circular shifted by one position for the other.

It is noteworthy to mention that the proposed modification is designed by considering the Kerchoff’s principle which states that a cryptosystem must remain secure even if the attackers know its complete inner workings except the key [11]. This includes the implementation details such as the character set, key length, attributes used for embedding, etc., which may be used by third parties for launching the attacks.

The organization of the rest of this paper is as follows. Section 2 discusses the existing techniques and their limitations. Section 3 introduces the proposed methodology which includes the embedding and extraction procedures. Section 4 presents the experimental results and Sect. 5 describes the impact of the proposed methodology with an example. Section 6 presents the evaluation of the proposed method in view of the enhanced security features. Section 7 presents the limitations of the proposed method. Section 8 provides the future scope and Sect. 9 concludes this paper.

2 Related works

This section presents the existing techniques of text steganography.

Text steganography, in general, can be broadly classified into two categories as coverless and cover-based embedding techniques. As the name implies, coverless embedding techniques do not require a cover document to embed the secret. Instead, it directly generates the stego work based on the characters in secret message. Example techniques include null cipher [12], missing letter puzzle [9] and hiding data in wordlist [9]. The main issue with these techniques is the low readability of the generated stego work. To address this limitation, some methods like [13, 14] make use of the recent advancements in natural language processing (recurrent neural networks and long short-term memory networks) to

generate a more meaningful stego work. However, they are either limited by size or the similarity between the generated stego work and human-written text. Alternatively, some methods generate a list of Uniform Resource Locators (URLs) in Chinese language by taking advantage of the parity of Chinese characters' stroke number to embed the secrets [15]. Hence, even though the method circumvents the issue of readability, it is restricted to Chinese language. Another method [16] maps each word in the secret message to a list of words in dictionary. The index of the mapped word is then used to generate an 18-bit integer value. The least 15-bits are used to select a username and the remaining 3-bits are used to select a domain name. The selected names are then combined using "@" character (imitating an email address) which is then placed in the "Carbon Copy (CC)" field of forward email template as a way to embed the secret. It is evident that this method leaks the number of words in the secret message as it will be equivalent to the number of email addresses in the CC field. Also, it expands the length of the secret message.⁶

Cover-based embedding techniques involve a cover document which will be modified as a way to embed the secret. Embedding can be performed either at bit-level or character level. Bit-Level Embedding Techniques (BLETs) convert the secret message into a bit stream which is then embedded in the cover document. Most BLETs are language-dependent and can further be divided into two categories as language-specific-character modification and content modification. As the name implies, language-specific-character modification modifies the characters of specific language without changing the content, whereas the other modifies the content itself. Language-specific-character modification methods include the modification of Kashida characters in Arabic alphabet [17–20], representation of Bengali characters in its equivalent roman form [21], etc. It is easy to understand that these language-specific-character modification techniques cannot be generalized for cover documents that are written in languages other than the desired one. Content Modification techniques, on the other hand, modify the content itself to embed the secret. It includes linguistic methods such as synonym substitution [22, 23], exploiting regional spelling difference of words [24], and the substitution of words based on part-of-speech tags [25]. Another approach in this direction performs a series of corrections to the cover document, based on the secret, making it to appear like an expert correcting the mistakes of the document which is written by an author with inferior writing skills. The embedded secret can then be extracted by the receiver by using the change-tracking feature of word processor [22, 26]. It clearly signifies that these methods greatly affect the readability of the generated stego document which is also a sign for suspicion.

Alternatively, language-independent methodologies of BLET exploit the whitespaces of cover documents. One such methodology, UniSpaCh [27], embeds the converted bits using four Unicode space characters whose width is smaller than the normal space character. However, it faces the issue of restricted embedding capacity as each modification can embed only 2-bits of information [27]. To overcome this limitation, some recent works like [28] use 16 different Unicode space characters and thereby embed 8-bits by performing two modifications at a time. That is, it inserts two such characters⁷ after each word that are selected using the eight most common part-of-speech tags which has occurred in the chosen cover work. Some alternative methods employ compression techniques, such as Huffman coding and LZW compression [22, 29, 30], with an aim

⁶ Average length of a word is 4.5 characters (not including space) [43] and average length of an email address is 21.9 characters [44].

⁷ Each character embeds 4-bits and thereby cumulatively achieves 8-bits.

to increase the embedding capacity. Some methods like [31] employ a hybrid method that use a combination of embedding techniques to achieve the same. That is, it uses the combination of inserting white spaces (one, two, or three) between words, shifting the position of words (up or down), inserting special symbols (apostrophe or hyphen) as a way to embed 3-bits at a time. However, it is noticed that BLET techniques can never achieve an embedding capacity attained by Character-Level Embedding Techniques (CLETs) as the latter operates at character-level. That is, CLETs directly embed a secret character (8-bits) in each modification of the cover document [10]. But, the main concern with CLET is to hide the matching frequency pattern between the secret message and the modified characters of the cover document.

It is noteworthy to mention that most of the traditional CLETs are keyless [32]. That is, they embed the secret characters by directly marking them in the cover document via common typographical errors like misspellings [33, 34], resizing of fonts [33], improper positioning of characters [35], etc. It is clear that an attacker can easily extract the embedded secret once (s)he identifies the embedding technique used. An alternative to this is keyed steganography, where the message can only be read by a receiver who has the key. There are two ways by which a key can be utilized in steganographic techniques. One way is to encrypt the secret first using cryptographic methods, such as the MLPHE-TS system [36], and then embed the resulting ciphertext using steganographic techniques. The other way is to include the elements of cryptographic algorithms in the embedding process itself. The AH4S stegosystem achieves this by using an Omega Network as a key to generate two related characters for each secret character and then embeds them as a 12-bit sequence white space character in front of appropriate cover word [37]. This method suffers from poor embedding capacity as the 8-bit secret character is expanded to 12-bit sequence. The system by [29] uses color-coded e-mail to embed the secret. It is evident that this method may attract attention when utilized in a professional environment as it involves sending an email with color-coded text in message body. Apart from these, the advancements in natural language processing have also been used by a few recent methods. The method [38] maintains two main parameters namely “wordindex” and “loopindex” whose values are first initialized with a random value (serves as a key) and subsequently determined using a hash and modulo functions respectively. To embed a secret character, the method first uses the wordindex value to select a word from an input sentence. It then masks the selected word, and subsequently provides the masked sentence as an input to a Bidirectional Encoder Representations from Transformers (BERT) model. The model uses the remaining words of the sentence and predicts a list of 257 compatible words as a replacement for that masked word of the given sentence. Finally, the method selects the first word which contains the secret character at the position mentioned by loopindex as a replacement for the masked word. Though this method can embed one whole character (8-bits) in a word/distortion, one should not forget to notice the fact that: (i) It can embed 8-bits in only one word referred by wordindex and not in all words of the sentence; (ii) Replaced word although fits grammatically in the respective position of that sentence, it may change the whole context of the sentence and thereby affect the continuity of the paragraph. For example, the sentence “And remember *I* will return” might be changed to “And remember *she* will return” to embed the secret character “s” with the loopindex value as “0” [38]. A technique involving Frequency Normalization Set (FNS) uses a character to string mapping, namely CSM, which then acts as a key during the embedding and extraction process [10]. Details of the same are already explained in Sect. 1 of this paper.

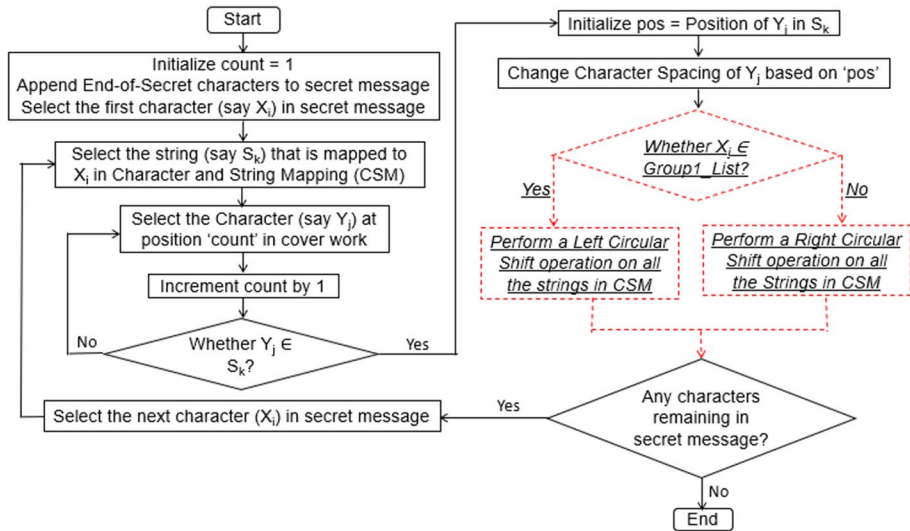


Fig. 1 Flowchart depicting the embedding process

From the conducted survey, it is concluded that the CLET based keyed steganography is superior and the methodology proposed by [10] appears to be efficient. This is because it follows CLET technique to embed the secret and embeds the secret character as it is (8-bits) without any expansion. Also, it uses a key to protect the embedded information which facilitates it to achieve cryptographic properties that are comparable with the poly-alphabetic substitution ciphers of cryptography [10]. Hence, we consider the same as the benchmark for our further study.

3 Proposed work

The proposed work comprises of two modules that include the embedding algorithm and extraction algorithm. As the name implies, the embedding algorithm embeds the secret characters inside the cover document, and the extraction algorithm does the reverse of the embedding algorithm.

3.1 Embedding algorithm

Similar to existing methodology, the proposed method embeds the secret by fractional modification of character spacing. It either condenses the spacing between characters by {0.1, 0.2, 0.3} points or expands the spacing by {0.1, 0.2, 0.3, 0.4} points.

The embedding process starts by selecting the first character from secret message and by selecting the corresponding mapped string from CSM. It then searches for the occurrence of any one character from the selected string in the cover document, in a serial manner. When a matching is found, it finds the position of the matched character in the selected string. Based on its position, the spacing is altered for the selected character in the cover

document. After embedding, instead of performing a constant left circular shift by one position, it checks the group to which the embedded secret belongs. Based on the group, the direction of the circular shift (left or right) is decided (refer Table 3). After embedding all the characters in secret message, similar to the existing methodology, it uses a special/End-of-Secret (EoS) pattern “Dot Space Dot Space Dot” in order to signify the receiver that the end of secret is reached. This pattern is considered as a part of secret message and is embedded after embedding the secret.

The detailed embedding process with proposed modification is provided as pseudo code and flowchart⁸ (refer Fig. 1 for the flowchart). The modified cover work is the generated stego work which must be communicated to the receiver. Along with the stego work, the used CSM, the respective character spacing of the 7 positions, EoS characters and characters of any one group should also be communicated. It is to be noticed that the stego work must be communicated in “Word” or “docx” format and all other information can be sent in any format like docx, txt, pdf, etc.

Pseudo Code of Embedding Procedure (Modified Version of the Embedding Procedure Provided in [10])

Input: Secret_Message, Cover_Work, Character and String Mapping (CSM), Group1_List, End-of-Secret (EoS) characters, character spacing and their respective positions

Output: Modified_Cover_Work or Stego_Work

Int count \leftarrow 1

Secret_Message \leftarrow Secret_Message + EoS Characters

For each character “X_i” in Secret_Message do

String S_k \leftarrow String that is mapped to X_i in CSM

L: Y_j \leftarrow Read the character at position “count” in Cover_Work

count++

If Y_j \in S_k

Int pos \leftarrow Position of Y_j in S_k

Change the character spacing of Y_j based on pos

End if

Else Goto L

If X_i \in Group1_List // Prevents the CSM from embedding the same secret character in the same cover character with the same character spacing, when the respective secret characters are exactly 7 characters apart

CSM \leftarrow Perform a Left Circular Shift on all the Strings in CSM

End if

Else

CSM \leftarrow Perform a Right Circular Shift on all the Strings in CSM

End else

End for

Return Modified_Cover_Work or Stego_Work

⁸ The proposed modifications are highlighted (red color font, italicized, and underlined) in both pseudo code and Fig. 1 for better understanding.

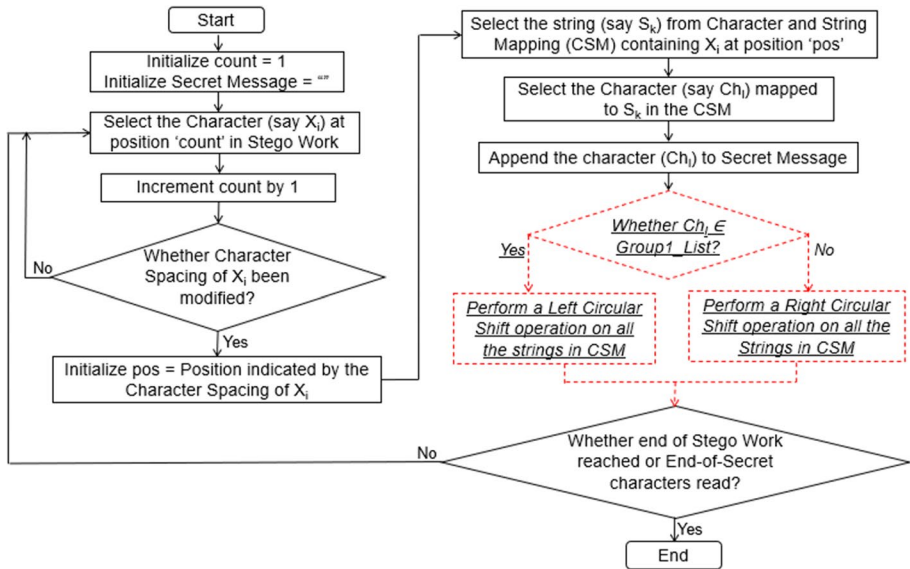


Fig. 2 Flowchart depicting the extraction process

3.2 Extraction algorithm

The extraction process is the reverse of embedding algorithm and is also almost similar to the existing methodology. The detailed extraction process with proposed modification is provided as pseudo code and flowchart⁹ (refer Fig. 2 for the flowchart).

The extraction process starts by reading the received stego work character-by-character. Whenever it encounters any character whose spacing is altered, it identifies the corresponding position using the spacing value. It then searches for the string in CSM which contains the identified character in the corresponding position. The character which is mapped to that string in CSM is the embedded secret. After extraction, instead of performing a constant left circular shift by one position, decision is taken based on the group to which the extracted secret character belongs (refer Table 3). The process is repeated until it encounters the EoS character pattern or the end of document. Reaching the end of the document without encountering the EoS characters indicates, the receiver, that a corrupted stego work is received.

⁹ The proposed modifications are highlighted (red color font, italicized, and underlined) in both pseudo code and Fig. 2 for better understanding.

Secret Message: Text Steganography is the technique of hiding secret data in text documents to avoid detection.

Cover Work:

INTRODUCTION

Internet which is extensively used to share any kind of information does not imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence during transmission, security to information plays a major role for Sensitive Organizations where the leakage of small amount of information will lead to critical problem. Information Security

Stego Work:

INTRODUCTION

Internet which is extensively used to share any kind of information does not imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence during transmission, security to information plays a major role for Sensitive Organizations where the leakage of small amount of information will lead to critical problem. Information Security

Stego Work (Stego Characters Highlighted for Understanding Purpose):

INTRODUCTION

Internet which is extensively used to share any kind of information does not imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence during transmission, security to information plays a major role for Sensitive Organizations where the leakage of small amount of information will lead to critical problem. Information Security

Color-Coding Information:

- Violet → Expanded the Character Spacing by 0.4 pt
- Indigo → Expanded the Character Spacing by 0.3 pt
- Blue → Expanded the Character Spacing by 0.2 pt
- Green → Expanded the Character Spacing by 0.1

- Yellow → Condensed the Character Spacing by 0.1 pt
- Orange → Condensed the Character Spacing by 0.2 pt
- Red → Condensed the Character Spacing by 0.3 pt

Extracted Secret Message:

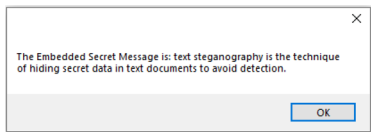


Fig. 3 Result of embedding a secret message in a cover document (*Existing Methodology*)

Secret Message: Text Steganography is the technique of hiding secret data in text documents to avoid detection.

Cover Work:

INTRODUCTION

Internet which is extensively used to share any kind of information does not imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence during transmission, security to information plays a major role for Sensitive Organizations where the leakage of small amount of information will lead to critical problem. Information Security

Stego Work:

INTRODUCTION

Internet which is extensively used to share any kind of information does not imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence during transmission, security to information plays a major role for Sensitive Organizations where the leakage of small amount of information will lead to critical problem. Information Security

Stego Work (Stego Characters Highlighted for Understanding Purpose):

INTRODUCTION

Internet which is extensively used to share any kind of information does not imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence during transmission, security to information plays a major role for Sensitive Organizations where the leakage of small amount of information will lead to critical problem. Information Security

Extracted Secret Message:

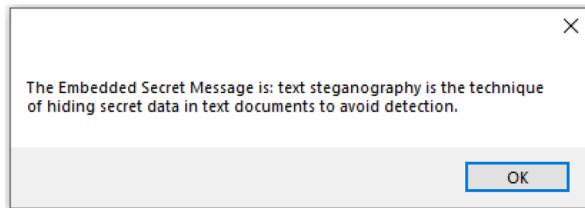


Fig. 4 Result of embedding a secret message in a cover document (*Proposed Methodology*)

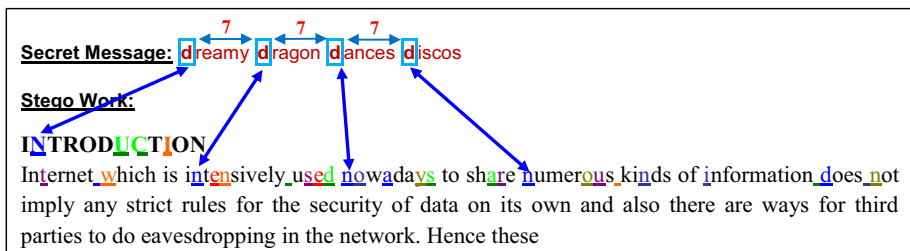


Fig. 5 Result of embedding process (*Existing Methodology*)

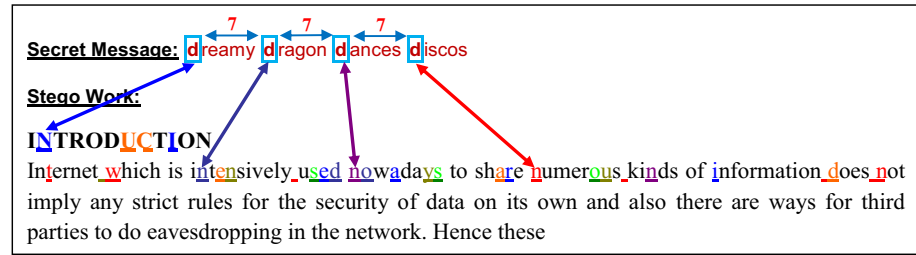


Fig. 6 Result of embedding process (*Proposed Methodology*)

Pseudo Code of Extraction Procedure (Modified Version of the Extraction Procedure Provided in [10])

Input: Stego_Work, Character and String Mapping (CSM), Group1_List, End-of-Secret (EoS) characters, character spacing and their respective positions

Output: Secret Message

```
Int count ← 1
```

```
String Secret_Message ← “”
```

Repeat

```
Char Xi ← Read the character at position “count” in Stego_Work
count++
```

If character spacing of $X_i \neq 0.0$ or Normal

Int pos \leftarrow Respective position, based on the character spacing of X_i

String $S_k \leftarrow$ String in CSM that has X_i at position pos

Char $Ch_l \leftarrow$ Character that is mapped to S_k in CSM

$$\text{Secret_Message} \leftarrow \text{Secret_Message} + \text{Ch}_1$$

If $Ch_j \in \text{Group1 List}$

CSM \leftarrow Perform a Left Circular Shift on all the Strings in CSM

End if

Else

CSM \leftarrow Perform a Right Circular Shift on all the Strings in CSM

End else

Until ((count > Total number of characters in Stego Work) || EoS is read)

Return Secret_Message

4 Experimental results

The first line of defense of any steganographic method is the secrecy/imperceptibility of the performed modification(s) [10]. In order to verify the imperceptibility of the performed modifications, a sample secret message, cover work and the result of embedding the secret message in the cover work for both existing and proposed methodologies are provided in Figs. 3 and 4 respectively. It is to be noticed that the stego characters in both Figures are highlighted/color-coded for understanding purpose.¹⁰ From the stego works, it can be

¹⁰ The same color-coding information is used for all the subsequent Figs. 4, 5 and 6.

understood that both methodologies do not attract any attention as they do not make any perceptible modifications.

5 Impact of proposed methodology

This section highlights the possibility of the potential attack that can be performed on the existing methodology and how the proposed modification overcomes the same.

5.1 Vulnerability of existing methodology

Below is an experiment that is performed with a tailored cover work and secret message. The secret message is customized to have the same character repeated with the same period as the length of the key, which is 7. The cover work is also chosen in such a way that all the specified/repeated characters get embedded in the same cover character with the same character spacing (refer Fig. 5). This is achieved by setting the period of repetition of characters in secret message similar to the length of the key. This scenario is possible when one tries to embed a lengthier secret message or a biased key of some other encryption algorithm.

This vulnerability allows an attacker to align all the modified characters in groups of seven, and identify all such occurrences where a secret character is embedded in the same cover character with the same character spacing. A frequency chart of English characters can then be utilized to identify these characters with statistical analysis [3]. Besides, one other consequence of this potential attack is that the attacker can target to identify the spaces in the secret message which can further damage the security of the method, as the common words can easily be identified based on their length [39]. The attacker can then utilize the obtained information to further identify the key used and thereby decode the entire secret message.

5.1.1 Reasons for attack potential

1. Number of characters between each occurrence of “d” in secret is same (ie: 7)
2. Length of string in CSM is 7
3. No. of shift after embedding each secret character is same (ie: 1)
4. Direction of shift after embedding each secret character is same (ie: Left)

5.2 Resilience of the proposed methodology to statistical attacks

The above vulnerability is overcome by the proposed methodology by introducing a group-based left or right circular shift in CSM (refer Fig. 6). Attackers cannot determine the sequence of shifts as it is decided based on the secret character that is being embedded. In addition, the groups are carefully chosen to balance out the frequency of occurrence of characters in both groups. This reduces the likelihood of any bias in the ratio of left and right shifts that are being performed. As a result, the identified groups ensure that the attacker does not have any opportunity to perform a frequency-based or word-length based attack to recover the key or embedded secret.

From Fig. 6, it can be noticed that even the periodic repetition of secret characters getting embedded in the same cover characters does not result in the same character

Table 4 Number of left and right rotations

Number of Left Rotations				Number of Right Rotations			
Character	To Embed S1	To Embed S2	Average	Character	To Embed S1	To Embed S2	Average
Space	850	913	881.5	Dot	48	51	49.5
a	356	374	365	d	166	138	152
b	73	80	76.5	e	523	474	498.5
c	120	116	118	g	79	94	86.5
f	93	96	94.5	h	210	201	205.5
m	83	135	109	i	268	294	281
n	286	281	283.5	j	2	4	3
q	3	4	3.5	k	27	40	33.5
r	242	254	248	l	214	160	187
s	338	267	302.5	o	313	298	305.5
u	96	104	100	p	63	67	65
v	26	44	35	t	387	351	369
x	3	9	6	w	81	80	80.5
z	3	4	3.5	y	44	67	55.5
Total	2572	2681	2626.5	Total	2425	2319	2372

spacing. This clearly indicates that the proposed methodology has indeed overcome the mentioned attack.

6 Evaluation of the proposed methodology

The proposed methodology neither modifies the embedding probability of secret characters nor the way the cover character that is being selected for embedding the secret. This ensures that the notable achievements of the existing methodology namely: (i) Embedding capacity = 27.87% (\approx 2-bits per cover character); (ii) 8-bits per distortion; (iii) High secrecy/imperceptibility level of the performed modification; and (iv) Uniformness in embedding the secret character (irrespective of its occurrence frequency); remains unaffected [10].

The proposed methodology modifies only the way the rotations that are being performed after embedding each secret character and subsequently the character spacing value that is being written in the cover character. That is, instead of a constant left circular shift operation, the proposed method performs both left and right circular shifts. This modification changes only the character spacing value that is being written in the cover character. As the proposed method is expected to have considerable cryptographic strength, the method must be unbiased and uniform random (which are the basic requirements of any cryptographic algorithm [40, 41]) while embedding the secret. Hence, the proposed methodology is evaluated for the same by using the following two factors: (i) Number of left and right rotations performed (to verify the unbiasedness); and (ii) Uniform distribution of stego characters in the available character spacings (to verify the uniform randomness).

Table 5 Uniform distribution in stego characters (*Existing Methodology*)

Character	-0.3	-0.2	-0.1	0.1	0.2	0.3	0.4	Mean	Variance	Std. Dev
Space	56	66	52	63	70	55	67	61.28571	41.06122	6.407903
Dot	7	4	1	5	4	3	8	4.571429	4.816327	2.194613
a	52	59	52	39	54	54	47	51	34.85714	5.903994
b	9	9	14	14	9	10	15	11.42857	6.530612	2.555506
c	19	33	21	11	25	24	24	22.42857	38.2449	6.184246
d	27	24	22	19	31	29	20	24.57143	17.95918	4.237828
e	66	72	84	58	78	86	86	75.71429	101.0612	10.05292
f	16	21	25	20	24	19	15	20	12	3.464102
g	22	17	14	14	18	11	14	15.71429	11.06122	3.325842
h	26	31	32	23	27	36	26	28.71429	17.06122	4.130524
i	74	58	57	71	71	85	70	69.42857	79.10204	8.893933
j	0	0	0	0	0	1	0	0.142857	0.122449	0.349927
k	1	1	1	2	5	2	2	2	1.714286	1.309307
l	14	20	14	13	21	19	16	16.71429	9.061224	3.010187
m	11	11	13	10	6	10	13	10.57143	4.816327	2.194613
n	69	65	74	74	68	61	66	68.14286	19.26531	4.389226
o	63	52	49	56	51	59	54	54.85714	20.40816	4.51754
p	8	5	4	8	3	4	3	5	4	2
q	0	1	1	0	0	1	1	0.571429	0.244898	0.494872
r	43	44	48	36	52	52	46	45.85714	26.97959	5.194188
s	36	35	38	45	45	39	31	38.42857	22.81633	4.776644
t	52	35	61	51	61	49	53	51.71429	65.91837	8.119013
u	13	12	6	16	11	12	15	12.14286	8.979592	2.996597
v	2	2	2	0	2	2	1	1.571429	0.530612	0.728431
w	11	8	4	10	5	7	12	8.142857	7.836735	2.799417
x	0	3	0	1	0	1	0	0.714286	1.061224	1.030158
y	14	9	15	8	9	11	16	11.71429	9.061224	3.010187
z	2	0	0	0	2	1	0	0.714286	0.77551	0.880631
Overall	713	697	704	667	752	743	721	713.8571	704.6939	26.54607

6.1 Number of rotations performed

As per the proposed methodology, the characters belonging to Group1 contributed to the number of left rotations, and the other contributed to the right rotations (refer Table 3). This indicates that the number of left and right rotations during the embedding process solely depend on the secret characters that are being embedded. As mentioned earlier in Sect. 1, the character groupings were carefully chosen to minimize the difference in cumulative occurrence frequency between the groups with an aim to avoid bias in the number of left and right circular shifts. To verify the same, experiments were conducted by using different secret messages S1 and S2, both consisting of 5000 characters. Table 4 depicts the number of left and right rotations performed for each character and the total counts as well.

The obtained results indicate that an average of 2626.5 left rotations and 2372 right rotations were performed on the strings in CSM. This is evident to conclude that the groupings have been fairly successful at distributing the secret characters between

Table 6 Uniform distribution in stego characters (*Proposed Methodology*)

Character	-0.3	-0.2	-0.1	0.1	0.2	0.3	0.4	Mean	Variance	Std. Dev
Space	57	74	53	68	65	49	63	61.28571	65.91837	8.119013
Dot	2	2	4	6	4	6	8	4.571429	4.244898	2.060315
a	42	58	52	56	56	46	47	51	31.71429	5.631544
b	9	9	10	10	12	23	7	11.42857	24.2449	4.923911
c	23	17	22	24	21	23	27	22.42857	7.959184	2.821203
d	26	19	28	26	25	21	27	24.57143	9.387755	3.063944
e	68	89	72	74	81	81	65	75.71429	60.4898	7.777519
f	21	25	11	20	25	26	12	20	33.14286	5.756983
g	19	15	18	9	12	18	19	15.71429	13.06122	3.614032
h	26	26	33	26	31	29	30	28.71429	6.77551	2.602981
i	66	63	70	76	60	78	73	69.42857	38.81633	6.230275
j	0	0	0	0	1	0	0	0.142857	0.122449	0.349927
k	3	0	2	3	3	1	2	2	1.142857	1.069045
l	16	19	15	20	11	18	18	16.71429	7.918367	2.813959
m	8	7	10	8	12	13	16	10.57143	9.102041	3.016959
n	76	64	71	60	71	56	79	68.14286	60.97959	7.808943
o	42	56	49	62	57	60	58	54.85714	41.83673	6.468132
p	6	2	5	9	3	5	5	5	4.285714	2.070197
q	0	1	1	0	0	1	1	0.571429	0.244898	0.494872
r	52	52	35	44	49	44	45	45.85714	30.12245	5.488392
s	36	43	42	42	35	30	41	38.42857	20.2449	4.499433
t	46	46	65	52	46	55	52	51.71429	40.77551	6.38557
u	15	10	14	12	12	15	7	12.14286	7.265306	2.695423
v	2	3	1	2	1	0	2	1.571429	0.816327	0.903508
w	7	5	8	11	6	8	12	8.142857	5.55102	2.35606
x	0	2	1	0	0	2	0	0.714286	0.77551	0.880631
y	14	12	12	8	18	10	8	11.71429	10.77551	3.282607
z	0	1	1	0	0	0	3	0.714286	1.061224	1.030158
Overall	682	720	705	728	717	718	727	713.8571	218.6939	14.7883

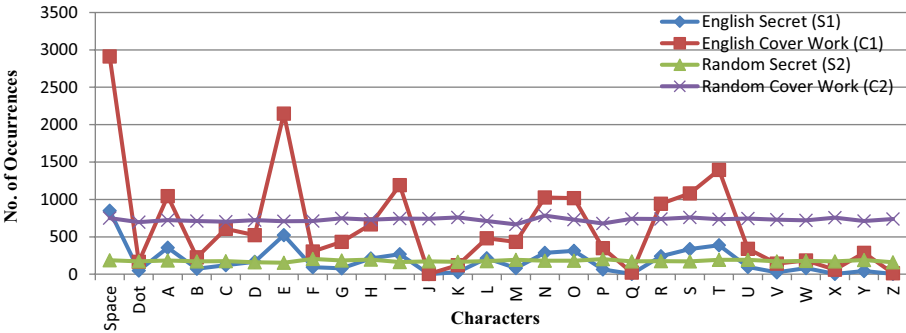


Fig. 7 Occurrence frequency of characters in both secret message and cover work (*both Random and English*)

the left and right rotations. This facilitates one to easily infer that an attacker can no longer rely on the periodical shift towards either direction for performing the statistical attacks.

6.2 Uniform distribution of stego characters

In order to study the impact of proposed methodology on the statistical effects during the embedding process, the degree of utilization of each potential character spacing was studied for each cover character that is being used for embedding the secret. The measure of success depends on the evenness of distribution among the available character spacings which can be noticed from the standard deviation values. Tables 5 and 6 displays the calculation of embedding 5000 character English secret message (S1) in English cover document (C1) of both existing and proposed methodologies respectively.

Additionally, random secret message (S2) comprising of 5000 characters and random cover document (C2) were generated. The individual occurrence frequency of all characters in S1, S2, C1 and C2 are provided in Fig. 7. Experiments were conducted for all the possible combinations namely embedding S1 in C2, S2 in C1 and S2 in C2 and the results of these experiments are provided in Table 7.

The results of these experiments indicate a decrease in standard deviation by 24% (refer Table 7) when compared with the existing methodology. This shows that the proposed methodology has more evenly distributed the character spacings over stego characters and hence is more resilience to statistical attacks.

6.3 Metadata analysis

A cause for concern is that the modifications performed on the cover work will lead to visible changes in the metadata of the file, allowing attackers to identify on-the-fly whether a document carries a secret or not. The primary attribute of concern is the file size, which may vary due to the modified values of the character spacing that must be stored for each individual change. To study this, the files resulting from the experiments of embedding S1 and S2 in C1 and C2 were examined before and after embedding the secret.

Table 8 records the results, which indicate that although there is an increase in file size, the change is only in the order of Kilo Bytes (KBs) and hence will not be distinguished from other documents that are being transmitted. It should be noted that the increase in the

Table 7 Uniform distribution of stego characters

Secret Message	Cover Document	Standard Deviation	
		Existing Methodology	Proposed Methodology
S1	C1	26.54607085	14.78830205
S1	C2	29.4348121	27.75108014
S2	C1	26.20095029	24.34111328
S2	C2	33.67794652	21.49228623
Overall		28.96494494	22.09319543

Table 8 Size Comparison before and after embedding the secret

Secret Message	Cover Document	Cover Work Size (KB)	Stego Work Size (KB)
S1	C1	32	63
S1	C2	29	59
S2	C1	32	64
S2	C2	29	59
Overall		30.5	61.25

order of KB is due to the number of characters in the secret message. That is, 5000 characters which is approximately 3 pages.¹¹

7 Limitations of the proposed method

From Sect. 6 it can be understood that, when compared with the existing methodology, the proposed one: (i) Withstands the statistical attacks better; (ii) Distributes the number of left and right rotations evenly among the stego characters; and (iii) Improves the uniform distribution of stego characters among the available spacing values. However, the proposed method is not the one without shortcomings. This is because of the fact that the existing methodology performs a constant left circular shift operation after embedding each secret character whereas the proposed one considers the group to which the most recently embedded secret character belongs, to decide among the two circular shifts (left and right). Hence, in the case of proposed methodology, any mistake in the most recently extracted secret character at recipient side will greatly affect the correctness of the subsequently extracted one(s).

An attacker can easily achieve the same by performing three different kinds of attacks, to the generated stego work, namely: (i) Modification attack; (ii) Insertion attack; and (iii) Deletion attack. That is, an attacker: (i) Can modify the spacing value of a stego character; (ii) Add a valid spacing value to an unmodified character in the stego work; and (iii) Delete the spacing value of a stego character. Considering this, the subsequent discussion of this section provides a performance comparison of the existing and proposed methodologies in the case of the mentioned attacks and is segregated into two parts. The first part discusses the possibilities of identifying such attacks at the recipient side and the second one focuses on the procedure of handling such attacks (once identified).

7.1 Detection of attacks

As both methodologies perform similar operation (circular shift) after embedding each secret character and uses the same pattern (EoS characters) to mark the end of the embedded secret, the identification of such attacks is almost similar in both methodologies and very straightforward. In the case of modification attack, the constant left circular shift operation in existing methodology restricts the attack to affect only the extraction of that

¹¹ The average number of characters per page is 1800 [42].

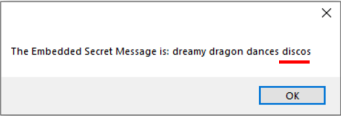
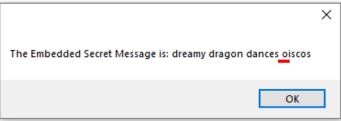
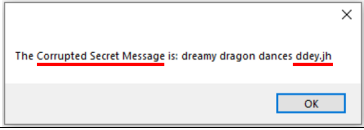
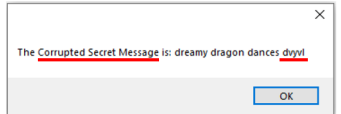
(a)	<p>Secret Message: dreamy dragon dances discos</p> <p>Stego Work (Stego Characters Highlighted for understanding purpose): INTRODUCTION Internet <u>wh</u>ich is <u>int</u>ensively <u>use</u>d <u>no</u>wadays to share <u>nu</u>merous <u>ki</u>nds of <u>in</u>formation <u>do</u>es <u>no</u>t imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence these</p> <p>Extracted Secret Message:</p> 
(b)	<p>Type of Attack: Modification</p> <p>Stego Work (Stego Characters Highlighted for understanding purpose): INTRODUCTION Internet <u>wh</u>ich is <u>int</u>ensively <u>use</u>d <u>no</u>wadays to share <u>nu</u>merous <u>ki</u>nds of <u>in</u>formation <u>do</u>es <u>no</u>t imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence these</p> <p>Extracted Secret Message:</p>  <div data-bbox="840 455 1032 525" style="border: 1px solid black; padding: 5px;"> Spacing on "n" is altered from "+0.2pt" to "-0.2pt" </div>
(c)	<p>Type of Attack: Insertion</p> <p>Stego Work (Stego Characters Highlighted for understanding purpose): INTRODUCTION Internet <u>wh</u>ich is <u>int</u>ensively <u>use</u>d <u>no</u>wadays to share <u>nu</u>merous <u>ki</u>nds of <u>in</u>formation <u>do</u>es <u>no</u>t imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence these</p> <p>Extracted Message:</p>  <div data-bbox="840 754 1032 825" style="border: 1px solid black; padding: 5px;"> Spacing of "m" is altered from "Normal" to "+0.2pt" </div>
(d)	<p>Type of Attack: Deletion</p> <p>Stego Work (Stego Characters Highlighted for understanding purpose): INTRODUCTION Internet <u>wh</u>ich is <u>int</u>ensively <u>use</u>d <u>no</u>wadays to share <u>nu</u>merous <u>ki</u>nds of <u>in</u>formation <u>do</u>es <u>no</u>t imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence these</p> <p>Extracted Message:</p>  <div data-bbox="840 1063 1032 1134" style="border: 1px solid black; padding: 5px;"> Spacing of "o" is altered from "-0.1pt" to "Normal" </div>

Fig. 8 Results of performing attacks in the output of exiting methodology

particular character alone and not others (refer Fig. 8b). Hence, the performed modification can easily be identified by the misspelling(s) that are present in the extracted secret message. However the same attack on the proposed method, can result in one of the following two possibilities: (i) If the modified spacing value also corresponds to a secret character which belongs to the same group of the originally embedded secret character (refer Table 3), then it behaves similar to the existing methodology as mentioned above (refer

(a)	<p>Secret Message: dreamy dragon dances discos</p> <p>Stego Work (Stego characters highlighted for understanding purpose): INTRODUCTION Internet <u>w</u>hich is <u>i</u>ntensively <u>u</u>sed <u>n</u>owadays to share <u>n</u>umerous <u>k</u>inds of <u>i</u>nformation <u>d</u>oes <u>n</u>ot imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence these</p> <p>Extracted Secret Message:</p> <div data-bbox="435 331 785 449"></div>
(b)	<p>Type of Attack: Modification</p> <p>Stego Work (Stego characters highlighted for understanding purpose): INTRODUCTION Internet <u>w</u>hich is <u>i</u>ntensively <u>u</u>sed <u>n</u>owadays to share <u>n</u>umerous <u>k</u>inds of <u>i</u>nformation <u>d</u>oes <u>n</u>ot imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence these</p> <p>Extracted Secret Message:</p> <div data-bbox="426 631 793 758"></div> <div data-bbox="846 449 1043 538"><p>Spacing of “n” is altered from “-0.3pt” to “+0.3pt”</p></div>
(c)	<p>Type of Attack: Modification</p> <p>Stego Work (Stego characters highlighted for understanding purpose): INTRODUCTION Internet <u>w</u>hich is <u>i</u>ntensively <u>u</u>sed <u>n</u>owadays to share <u>n</u>umerous <u>k</u>inds of <u>i</u>nformation <u>d</u>oes <u>n</u>ot imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence these</p> <p>Extractor Secret Message:</p> <div data-bbox="435 940 785 1067"></div> <div data-bbox="846 758 1043 846"><p>Spacing of “n” is altered from “-0.3pt” to “-0.1pt”</p></div>
(d)	<p>Type of Attack: Insertion</p> <p>Stego Work (Stego characters highlighted for understanding purpose): INTRODUCTION Internet <u>w</u>hich is <u>i</u>ntensively <u>u</u>sed <u>n</u>owadays to share <u>n</u>umerous <u>k</u>inds of <u>i</u>nformation <u>d</u>oes <u>n</u>ot imply any strict rules for the security of data on its own and also there are ways for third parties to do eavesdropping in the network. Hence these</p> <p>Extracted Secret Message:</p> <div data-bbox="426 1240 793 1360"></div> <div data-bbox="846 1067 1043 1155"><p>Spacing of “m” is altered from “Normal” to “+0.2pt”</p></div>

Fig. 9 Results of performing attacks in the output of proposed methodology

Fig. 9b). That is, it affects only the extraction of that character alone; and (ii) If they do not belong to the same group, then it will lead to the extraction of a meaningless/corrupted secret message (refer Fig. 9c). In such a case, the existence of the intermediate attacker can easily be spotted as the: (i) Received stego work generates an invalid text after the modified/attacked character; and (ii) Embedded secret does not end with the EoS characters.

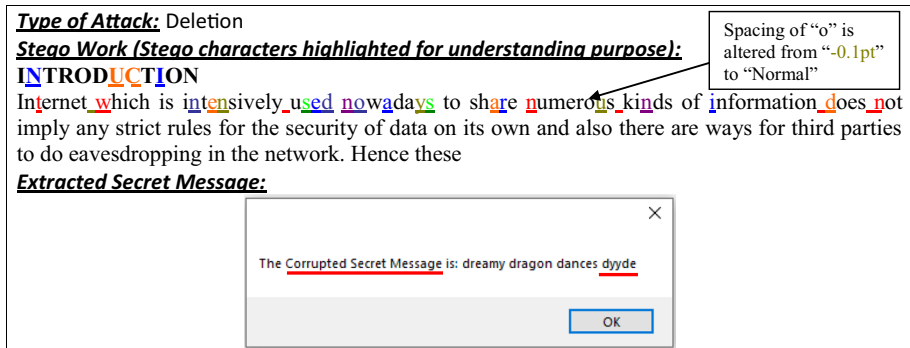


Fig. 10 Result of performing attacks in the output of proposed methodology

Unlike the modification attack, the insertion and deletion attacks behave similarly in both methodologies. That is, it results in the extraction of a corrupted message leaving the same above-mentioned two clues (refer Figs. 8c, d, 9d and 10).

7.2 Handling of attacks

Similar to the detection of attacks, the handling procedures of the mentioned attacks also have some commonalities in both methodologies. In the case of modification attack, the misspellings in both methodologies (refer Figs. 8b and 9b) can easily be rectified with the help of a dictionary. However the corrupted secret message (refer Fig. 9c), in the case of proposed methodology, can be handled by identifying the extracted secret character from where the message is not meaningful and then perform a different rotation than the one that is being insisted by the most recently extracted secret character. For example, in the case of Fig. 9c, perform a right circular shift operation after extracting the secret character "v" (as opposed to the one insisted by Table 3). This will facilitate to extract the remaining secret characters at ease. Once extracted, the mistake in character "v" can be rectified by following the similar procedure mentioned to correct the error in Fig. 9b.

In the case of insertion attack, both existing and proposed methodologies can ignore the most recently extracted secret character and also undo the most recent circular shift operation that was performed after extracting the secret character. In the case of deletion attack, the existing methodology can perform an additional left circular shift operation before extracting the secret character from where the message corruption starts. After performing the additional shift operation, it can continue extracting the remaining embedded secret. Finally, it can make use of the dictionary to correct the misspelling in one of the words where the message corruption occurred. However, in the case of proposed methodology, it must follow a trial-and-error method with high human intervention. That is, it must try both left and right circular shifts at the point where the message corruption starts and then proceed further with the one which provides a meaningful text.

Though the procedure of handling the three attacks sound simple and straightforward in both methodologies, one should not forget the fact that the distinction among the: (i) insertion and deletion attacks in the case of existing methodology; and (ii) modification, insertion and deletion attacks in the case of proposed methodology; cannot be made as they all

end up in a corrupted secret message (refer Figs. 8c, d, 9c, d and 10). This again leaves the recipient in a situation of handling them through a trial-and-error method.

From the above discussions, it is clear that the existing method handles the mentioned attacks better than the proposed methodology, as there is no dependency between the circular shift operation and the most recently extracted secret character. Also the extraction of the secret message becomes more challenging, in the case of proposed methodology, when the attacker engages in a combination of these attacks which will require a high human intervention. In such a case, this can be considered as a limitation of the same.

8 Future work

Future work can focus to make the proposed method attain a cryptographic strength which is comparable with encryption algorithms that operate at bit-level [3].

9 Conclusion

Securing the information while in transit and in storage is a major concern for any sensitive organization. Organizations mostly employ cryptographic techniques to achieve the same which ensures confidentiality, integrity, and authentication of the information but fails to provide secrecy. Steganography, on the other hand, provides secrecy but it lacks security. Hence, a methodology that combines techniques from both steganography and cryptography is a beneficial one as it provides an additional layer of security. The best existing method which works in similar fashion is observed to use character spacing feature of text documents to embed the secrets. Though the method is reported to have noticeable benefits like high embedding capacity and requires less number of distortions to embed the secrets, it has the potential vulnerability of leaving clues for attackers leading to possible attacks. The vulnerability is noticed in the constant left circular shift operation that is being performed after embedding each secret character. This paper handled the limitation by proposing a methodology that performs both left and right circular shift operations in an unbiased manner without affecting the reported advantages of the existing methodology. The proposed modification is tested thoroughly and the experimental results indicated that the performed modification have in fact increased the resilience of the extended method to potential attacks that rely on statistical techniques.

Acknowledgements The authors would like to sincerely thank the editors and reviewers for their valuable comments and suggestions.

Data availability Data used for experiments will be provided based on reasonable request.

Declarations

Ethical approval This article does not contain any studies performed by any of the authors that involves human participants or animals.

Informed consent Informed consent was obtained from all the individual participants included in the study.

Conflict of interest The authors declare that we do not have any conflict of interest.

Agreement We declare that this work is original and not considered for publication in any other publication media.

References

- Kemp S (2019) Digital 2019: global digital overview. [Online] Available: <https://datareportal.com/reports/digital-2019-global-digital-overview>. Accessed 17 Dec 2022
- Ahmad A, Maynard SB, Park S (2014) Information security strategies: towards an organizational multi-strategy perspective. *J Intell Manuf* 25:357–370
- Forouzan BA, Mukhopadhyay D (2011) *Cryptography and network security*. Tata McGraw-Hill Education
- Mishra R, Bhanodiya P (2015) A review on steganography and cryptography. *Int Conf Adv Comput Eng Appl*, Ghaziabad, India, pp 119–122
- Kishor SN, Ramaiah GNK, Jilani SAK (2016) A review on steganography through multimedia. *Int Conf Res Adv Integr Navig Syst (RAINS)*, Bangalore, India, pp 1–6
- Song S, Zhang J, Liao X, Du J, Wen Q (2011) A novel secure communication protocol combining steganography and cryptography. *Procedia Eng* 15:2767–2772
- Callaham J (2018) [Online]. Available: <https://www.windowcentral.com/there-are-now-12-billion-office-users-60-million-office-365-commercial-customers>. Accessed 3-10-2022
- Bender W, Gruhl D, Morimoto N, Lu A (1996) Techniques for data hiding. *IBM Syst J* 35(3.4):313–336
- Agarwal M (2013) Text steganographic approaches: a comparison. *Int J Netw Secur Appl* 5:91–106
- Ramakrishnan BK, Thandra PK, Satya MurtySrinivasula AV (2017) Text steganography: a novel character-level embedding algorithm using font attribute. *Secur Commun Netw* 9(18):6066–6079
- Stamp M (2011) *Information security principles and practice*, 2nd edn. Wiley
- Gleason N (1987) *Fun with codes and ciphers workbook*. Dover Publications
- Peng W, Wang T, Qian Z, Li S, Zhang X (2023) Cross-modal text steganography against synonym substitution-based text attack. *IEEE Signal Process Lett* 30:299–303
- Cao Y, Zhou Z, Chakraborty C, Wang M, Wu QMJ, Sun X Yu K (2022) "Generative steganography based on long readable text generation," *IEEE transactions on computational social systems*, pp 1-11
- Wang K, Gao Q (2019) A coverless plain text steganography based on character features. *IEEE Access* 7:95665–95676
- Maji G, Mandal S (2020) A forward email based high capacity text steganography technique using a randomized and indexed word dictionary. *Multimed Tools Appl* 79:26549–26569
- Alanazi N, Khan E, Gutub A (2021) Efficient security and capacity techniques for Arabic text steganography via engaging Unicode standard encoding. *Multimed Tools Appl* 80:1403–1431
- Gutub AA-A, Alaseri KA (2021) Refining Arabic text stego-techniques for shares memorization of counting-based secret sharing. *J King Saud Univ – Comput Inform Sci* 33(9):1108–1120
- Alanazi N, Khan E, Gutub A (2022) Inclusion of Unicode Standard seamless characters to expand Arabic text steganography for secure individual uses. *J King Saud Univ – Comput Inform Sci* 34(4):1343–1356
- Al-Nofaie SMA, Gutub AA-A (2020) Utilizing pseudo-spaces to improve Arabic text steganography for multimedia data communications. *Multimed Tools Appl* 79:19–67
- Khairullah M (2019) A novel steganography method using transliteration of Bengali text. *J King Saud Univ – Comput Inform Sci* 31(3):348–366
- Mahato S, Khan DA, Yadav DK (2017) A modified approach to data hiding in Microsoft word documents by change tracking technique. *J King Saud Univ – Comput Inform Sci* 32(2):216–224
- Qi C, Xingming S, Lingyun X (2013) A secure text steganography based on synonym substitution. *IEEE Conference Anthology, China*, pp. 1–3
- Shirali-Shahreza M (2008) Text steganography by changing words spelling. In: *10th International conference on advanced communication technology*. Gangwon, Korea (South), pp 1912–1913
- Banik BG, Bandyopadhyay SK (2018) Novel text steganography using natural language processing and part-of-speech tagging. *IETE J Res* 66(3):384–395
- Liu T-Y, Tsai W-H (2007) A new steganographic method for data hiding in microsoft word documents by a change tracking technique. *IEEE Trans Inf Forensics Secur* 2:24–30
- Por LY, Wong K, Chee KO (2012) UniSpaCh: a text-based data hiding method using unicode space characters. *J Syst Softw* 85(5):1075–1082

28. Almawayhi MAM, Sulaiman R, Shukur Z (2024) New text steganography technique based on part-of-speech tagging and format-preserving encryption. *KSII Trans Internet Inf Syst* 18(1):170–191
29. Malik A, Sikka G, Verma HK (2016) A high capacity text steganography scheme based on LZW compression and color coding. *Eng Sci Technol Int J* 20(1):72–79
30. Khosravi B, Khosravi B, Khosravi B, Nazarkardeh K (2019) A new method for pdf steganography in justified texts. *J Inform Secur Appl* 45:61–70
31. Gurunath R, Samanta D (2023) A new 3-bit hiding covert channel algorithm for public data and medical data security using format-based text steganography. *J Database Manag* 34(2):1–22
32. Brassil JT, Low S, NMF, (1999) Copyright protection for the electronic distribution of text documents. *Proc IEEE* 87(7):1181–1196
33. Bennett K (2004) Linguistic steganography: survey, analysis, and robustness concerns for hiding information in text. CERIAS Tech Report 2004-13, Center for Education and Research in Information Assurance and Security. Purdue University, pp 1–29
34. Topkara M, Topkara U, Atallah MJ (2007) Information hiding through errors: a confusing approach. In: *Proc. SPIE 6505, Security, Steganography, and Watermarking of Multimedia Contents IX*, 65050V
35. Brassil JT, Maxemchuk NF (1999) Copyright protection for the electronic distribution of text documents. *Proc IEEE* 87(7):1181–1196
36. Naqvi N, Abbasi AT, Hussain R, Khan MA, Ahmad B (2018) Multilayer partially homomorphic encryption text steganography (MLPHE-TS): a zero steganography approach. *Wireless Pers Commun* 103:1563–1585
37. Hamdan AM, Hamarsheh A (2017) AH4S: an algorithm of text in text steganography using the structure of omega network. *Secur Commun Netw* 9(18):6004–6016
38. Ozturk E, Mesut AS, Fidan OA (2024) A character based steganography using masked language modeling. *IEEE Access* 12:14248–14259
39. [Online]. Available: <https://www.learncbse.in/words-by-length/>. Accessed 17 Dec 2022
40. Creating biased random numbers from a cryptographically secure source. Available: <https://codeql.github.com/codeql-query-help/javascript/js-biased-cryptographic-random/>. Accessed 3 Mar 2024
41. "What is the difference between uniformly and at random in crypto definitions?," [Online]. Available: <https://crypto.stackexchange.com/questions/20839/what-is-the-difference-between-uniformly-and-at-random-in-crypto-definitions>. Accessed 3 Mar 2024
42. [Online]. Available: <https://www.lexika-translations.com/blog/what-is-a-standard-page/#:~:text=One%20standard%20page%20is%20comprised%20of%201%2C800%20characters%20with%20spaces>. Accessed 17 Dec 2022
43. Pierce JR (1980) *An introduction to information theory: symbols, signals and noise*, 2nd edn. Dover Publications, New York
44. How long is the average email address? [Online]. Available: <https://www.atdata.com/blog/long-email-addresses#:~:text=if%20you%20want%20your%20form,show%20at%20least%2031%20characters>. Accessed 17 Dec 2022

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.