Text Based Steganography – A Theoritical Proposal of Text Based Hiding Strategy

Ayan Chatterjee, Gourab Dolui, Dr. Uttam Kumar Roy

Abstract—Steganography is a useful tool of hiding information in ways that prevent the detection of hidden messages. It serves better than cryptography which only hides the content of the message not the existence of the message. The output of steganography operation is not actually visible but in cryptography the output is scrambled so that it can draw attention. In Steganography, Original message is being hidden within a carrier such that the changes so occurred in the carrier are not observable. It is a process of combining secret information with the carrier medium gives the hidden message. The hidden information is difficult to detect without retrival. This paper will look at a new proposal on text based information hiding technique (encoding or encryption) and steganolysis (decoding or decryption) technique on top of the existing text based methods. Text based steganography with digital files is not used very often since text files have a very small amount of redundant data. Using this technique we can hide more binary information with minimal number of alphabets. This method also provides a security on secret binary information.

Index Terms—AVL Tree; Hamming Code; Hashing; Recurrence Relation; Binary Information; Encoding; Decoding; Steganolysis.

1 INTRODUCTION

THE Steganography means passing of secret information within an innocent cover and send it to proper recipient who is aware of decoding process. The first use of Steganographic techniques was found in Greece. It was also much popular in pirates to pass some secret information by drawing tattoo on the map or head or some part of the body. It has been found in China and in grille system adopted in WWII that a wooden template is placed over a innocent text but that is carrying a hidden message. In 1980, in UK text based steganography was adopted with word spacing technique. Steganographic techniques became very popular to pass military secret information. During war spread spectrum technique or meteor scatter radio was used to hide both source and the message. Later, as days passed new promising techniques came into picture with carriers like electromagnetic waves, digital images, audio files, video files and communication protocols or network protocols. Later new techniques were introduced like copyright enforcement, digital watermarking techniques, embedding data in images.[3] Today, it is researched for both legal and illegal causes. We can broadly classify steganography into three categories based on selection of cover medium like text based, image based & video based. Text based steganography is an important part of research which has been practised since long period but it has some problem due to lack of redundancy as compared to image & video. In this paper, we have given a theoretical proposal for text based steganography. [1, 4]

- *Ayan Chatterjee is currently pursuing masters degree program in Software engineering in Jadavpur University, India, PH-09874162451. E-mail: ayan1.c2@mail.com*
- *Gourab Dolui is B.TECH in computer engineering in West Bengal University Of Technology, India, E-mail: gdo@tdc.dk*
- *Uttam Kumar Roy is professor in software engineering in Jadavpur University, India, E-mail: u_roy@it.jusl.ac.in*

2 TEXT BASED STEGANOGRAPHY

2.1 Overview

It is a steganographic technique where text is used as a cover medium. It is most difficult to manage due to lack of redundant spaces in the cover media. From long back a lot of investigations and practices have been carried out to do experiment on text medium to pass secret information like line shift coding, word shift coding, feature coding, first letter algorithm, N-letter algorithm, altering the amount of white-space etc. It can be classified into two categories like linguistic & format based. Linguistic can be divided into simentic & syntax method and format based can be divided into line shift encoding, feature coding, white space coding and word shift coding [3, 5].

2.2 Related Work

Multiple researches has been carried out in the field of text based steganography like below:

Moerland proposed a technique where hiding of secret message can be done by altering specific characters in a word and using punctuations like comma, semicolon etc. Low, Maxemchuk, Brassil, Gorman [6] ,Alattar [7] and Kim, Moon, Oh [8] proposed alphabet shifting technique where we simply shift various alphabets inside the pages up or down by a small fraction (such as 1/300th of an inch) or shifting right/left (by 0.5 points) or changing it's font style according to the codebook. The shifted alphabets are undetectable by humans because it is only a small fraction but is detectable when the computer measures the distances between each of alphabets. Differential encoding techniques are normally used in this protocol, meaning if you shift an alphabet the adjacent alphabets are not moved. This helps to detect which alphabets are shifted to represent secret information with the help of computer. Niimi, Minewaki, Noda, Kawaguchi also proposed a technique which uses synonyms of certain words in order to hide the message in the english text. Huang, Yan proposed a method for hiding secure information by adding extra white-spaces in the text.[9] Shirali-Shahreza [10, 11] and Memon,

Khowaja, Kazi [12] also proposed a steganography method on Arabic, Persian and Urdu text as there are abundance of points and each point shifting can hide information. Alla, Prasad proposed a hindi text steganography technique based on vowels and consonants. Shirali-Shahreza proposed a text steganography technique that hides secret message in the English text by using different spellings of the words. Wang, Chang, Kieu, Li proposed an emotion based text steganography technique. Proposal has also been given on altering html tags to hide binary data.

3 NEW APPROACHES

3.1 Proposed Work

The techniques proposed here are the result of collaborative study of existing word count method, AVL tree, and hamming code & its height calculation, alphabet shifting, letter count method and special character based steganometric approaches. In this way, we can store binary information in the pages of a book and using its lines & words per page. Text based steganography using digital files is not used very often since text files have a very small amount of redundant data. Using this technique we can hide huge binary information with minimal number of alphabets. This method also provides a security on secret binary information. For this proposed study, we need to choose pages and secret binary information & divide the binary information in those pages in such a way that they can hide the information completely with minimal use of alphabets. Scan all lines in each page and prepare a resulting height chart for AVL Tree.

T(h)=1+T(h-1)+T(h-2) where T(1)=1,T(2)=2.T(h) is representing minimal number of node present in the AVL tree of height 'h' (h>=1).

We have discussed below algorithms with example to show new text based stegano approach based on below sentence in a page:

"Graphology is claimed to be useful for everything from understanding health issues, morality and past experiences to hidden talents, and mental problems. The person that uses the concepts of graphology to this end is known as graphologist. However, the graphology is not restricted to this. Forensic document examiners (FDE) use it to examine handwriting in order to detect authenticity or forgery. A type of handwriting that is subject of analysis very often is the signature. With the power of computers growing exponentially, researchers have tried to use the ideas of graphology and the expertise of FDE to automatically analyze and verify signatures. The criterion used to select them was if they were feasible computationally. Then, we establish a relationship between features from these two fields in order to propose a set of features that can be applied to automatic signature verification. The classifiers used are the hidden Markov models. Finally, we discuss the advantages and drawbacks of using such features in context of signature verification."

3.2 Algo#1: Enhancement of Word Count Method

Input: A page containing sentences.

Output: Alphabet sequence representing hidden binary information.

Step 1: Count number of words present in every line of that page.

Step 2: If multiple lines produce same word count, consider the first line.

Step 3: create an AVL tree using those words count as nodes.

Example: create an AVL Tree using nodes 22, 15, 8, 16, 14 , 27 , 11

Step 3.1: If height of the AVL tree is > = 4 then Goto step 4 else Goto next page. (height >=1 && minimal group size is equal to 3 bits).

Step 4: Divide the secret binary information in small groups <= (height of the tree -1) <where h>=1>.

Example: If in a page 10 lines are present and out of 10 lines three have same word count so we have to choose any one of them. Total lines are seven and height of the AVL tree is 4. Maximum group size (4-1)=3.

We are going to hide 011001010. So, we are diving it among three groups 011, 001, 010.

Step 5: Put binary weight on the edges like value: 0 to the left edge and 1 to the right edge. (Hamming code approach)

Step 6: Give a name to the nodes based on their initials (in corresponding sentences). If two initials are found same then choose initial searching next character on first come first serve basis. (Taking inspiration from Markov Algorithm)

Example: 22 (G), 15(T), 8(H), 16(F), 14(A) , 27(E) , 11(C).

Step 7: calculate binary value of each node except the root one.

Example: G (11) /T (X) /H (0) /F (1) /A (10) /E (111) /C (01)

NB: Node T denotes no binary information. Suppose our secret information is 0110010 and we are dividing into three groups like 011 , 001 , 0XX. '0XX' means 'HT' (a single T means no further information in that page. No need to use a second 'T').

Step 8: Represent each binary group in terms of Alphabets representing nodes with possible minimal use of alphabets.

Example: 011 : CF or HG 001 : HC or HHF 010 : HA or CH

Step 8.1: If you want to hide the code words in that page goto step 9 else stop here and pass the information separately as (min_alphabets(all possible combinations)).

Step 9:

For I = 1 to Total number of groups do

- check number of solutions.
- Choose first solution and check the alphabet sequence in the lines of that page
- If successful take the first solution else go for second
- Put the secret Key information in the page.

End For

End of the algorithm.

3.3 Algo#2: Enhancement of Letter Count Method

Input: A page containing sentences.

Output: Alphabet sequence representing hidden binary information.

Step 1: Count number of letters present in every line of that page (Ignore spaces or other punctuations).

Step 2: If multiple lines produce same letter count, consider any single one. Finally, Count number of lines present in that page.

Step 3: create an AVL tree using those letters count.

Example: create an AVL Tree using nodes 130, 70, 41, 89, 64 , 149 , 140, 42, 92

Step 3.1: If height of the AVL tree is > = 4 then Goto step 4 else Goto next page. (height >=1 && minimal group size is equal to 3 bits).

Step 4: Divide the secret binary information in small groups <= (height of the tree -1) <where h>=1>.

Example: If a page 10 lines are present and out of 10 lines three have same word count so we have to choose any one of them. Total lines are nine and height of the AVL tree is 4. Maximum group size (4-1)=3.

We are going to hide 011001010 in the above page. So, we are diving it among three groups 011, 001, 010.

Step 5: Put binary weight on the edges like value: 0 to the left edge and 1 to the right edge. (Hamming code approach)

Step 6: Give a name to the nodes based on their initials (in corresponding sentences). If two initials are found same then choose from next letter. (Taking inspiration from Markov Algorithm).

Example: 70(T), 41(H), 130(G), 64(A), 42(C) , 89(F) , 149(W), 92(I), 140(E).

Step 7: calculate binary value of each node except the root one.

Example: G (1) /T (X) /H (0) /F (10) /A (01) /E (111) /C (010)/W (11)/I (101)/E (110)

NB: Node T denotes no binary information. Suppose our secret information is 0110010 and we are dividing into three groups like 011 , 001 , 0XX. '0XX' means 'HT' (a single T means no further information in that page. No need to use a second 'T').

Step 8: Represent each binary group in terms of Alphabets representing nodes with possible minimal use of alphabets.

Example:

011 : HW 001 : HA 010 : C

Step 8.1: If you want to hide the code words in that page goto step 9 else stop here and pass the information separately as (min_alphabets(all possible combinations)).

Step 9:

For I = 1 to Total number of groups do

- check number of solutions.
- Choose first solution and check the alphabet sequence in the lines of that page
- If successful take the first solution else go for second
- Put the secret Key information in the page.

End For

End of the algorithm.

3.4 Algo#3: Modified White Space – Dot - LineBreak-Tab-Hidden Character" Manipulation with Rotation Technique to hide secret binary information based on page number

Prepared Table for Hiding Binary Information:

Initial Assumptions:

<Hidden character><dot>: Beginning of Representation

<Dot><Hidden character>: End of Representation

<Dot>line break><space>: No Information

<Dot><line break><Tab>: No Information

N.B: Hidden character is "0 with Space".

Manipulation Techniques	Storing Binary Information	Value After MOD operation
<dot>	0	1
<space><dot>	01	
<dot><space>	10	
<space><dot><space>	00	
<dot><line break>	1	
<space><dot><line break>	11	

Manipulation Techniques	Storing Binary Information	Value After MOD operation
<dot>	11	2
<space><dot>	0	
<dot><space>	01	
<space><dot><space>	10	
<dot><line break>	00	
<space><dot><line break>	1	

Manipulation Techniques	Storing Binary Information	Value After MOD operation
<dot>	1	3
<space><dot>	11	
<dot><space>	0	
<space><dot><space>	01	
<dot><line break>	10	
<space><dot><line break>	00	

Manipulation	Storing Bina-	Value After

Techniques	ry Information	MOD operation
<dot>	00	4
<space><dot>	1	
<dot><space>	11	
<space><dot><space>	0	
<dot><line break>	01	
<space><dot><line break>	10	

Manipulation Techniques	Storing Binary Information	Value After MOD operation
<dot>	10	5
<space><dot>	00	
<dot><space>	1	
<space><dot><space>	11	
<dot><line break>	0	
<space><dot><line break>	01	

Manipulation Techniques	Storing Binary Information	Value After MOD operation
<dot>	01	6
<space><dot>	10	
<dot><space>	00	
<space><dot><space>	1	
<dot><line break>	11	
<space><dot><line break>	0	

Input: Page containing page number and sentences.

Output: Space - Dot – LineBreak-Tab-Hidden Character sequence representing hidden binary information.

Step 1: Count Number of Lines (var: NOL).

Step 2: Count Number of bits present in secret binary information (var: NOB)

Step 3: If NOB <= CEIL (NOL /2) then GoTo Step 4 else GoTo End

Step 4: Find the page number from the footer section. (Var: PN)

Step 5: var: result: = (Sum of Digits of PN) MOD 6 (as in the table we have mentioned six sequence types).

Step 6: Based on the value of result variable, choose the sequence from the correct table and apply it accordingly in the text.

Step 7: END.

3.5 Algo#4: Combining two methods to hide same binary information: (Algo#1 & Algo#3)

Step 1: Count number of words present in every line of that page (var: NOW)

Step 2: Count Number of Lines (var: NOL) <all lines are considered>

Step 3: If multiple lines produce same word count, consider any single one. Finally, Count number of lines present in that page. (Var: NOL_U)

Step 4: Count Number of bits present in secret binary information (var: NOB)

Step 5: create an AVL tree using those words count.

Step 5.1: If height of the AVL tree is > = 4 then Goto step 4 else Goto next page. (Height >=1 && minimal group size is equal to 3 bits).

N.B: Check height from the prepared height chart.

Step 6: If (NOB <= CEIL (NOL/2) AND (height of the AVL tree is > = 4)) = TRUE Then

Boolean A: = CALL PROCEDURE-A

Boolean B: = CALL PROCEDURE-B

If (A AND B) = TRUE THEN

GOTO Step 7

ELSE GOTO Step 8

END IF

ELSE GOTO Step 8

END IF

Step 7: Success & GOTO 9

Step 8: Failure & GOTO 9

Step 9: END

End of the algorithm.

Boolean PROCEDURE-A:

Goto Step#4 of Algo#1 to Step#9

If all steps are successful then return TRUE else FALSE

END PROCEDURE-A

Boolean PROCEDURE-B:

Step 1: If NOB <= CEIL (NOL /2) then GoTo Step 4 else GoTo End

Step 2: Find the page number from the footer section. (Var: PN)

Step 3: var: result: = (Sum of Digits of PN) MOD 6 (as in the table we have mentioned six sequence types).

Step 4: Based on the value of result variable, choose the sequence from the correct table and apply it accordingly in the text.

Step 5: If all steps are successful then return TRUE else FALSE

END PROCEDURE-B

4 COMBINING ALGO#1 AND #2 TO ALGO#3

It has some advantages as narrated below:

- In a page we can use both representations to hide same or different binary information.
- If we hide different binary information then this technique will help in double data hiding in the same page.
- If we hide same binary information in a page with two hiding strategies (algo#1, algo#2 & algo#3) then it will help us as below:

Receiver can extract hidden information separately using corresponding decoding method and can check if both information is same – if same then received information is correct else tampered.

Conditions for combining two methods to hide same binary information:

(NumberOfBinaryBits of hidden information <= CEIL(NumberOfLines in that page /2) AND (height of the AVL tree is > = 4)) must come true.

((Hiding binary information with algo#1 & algo/#2 in the page) AND (Hiding same binary information with algo#3 in that page)) must come true.

5 CREATION OF SECRET KEY FOR RECEIVER

It can be done based on below approaches:

- Create a private key with alphabet positions on each page and send it along with the cover media. Receiver side will retrieve the secret information based on the private key.
- Create a private key with alphabet sequence on each page and send it along with the cover media. Receiver side will retrieve the secret information based on the private key.
- For the above steps we have taken reference from alphabet shift coding technique and taking inspiration from line & word shift coding protocol.

  Example: 'S in gu lar' of original 'Singular'

5 CONCLUSION

Word Count Method gives priority on counting number of words present in a line while Letter Count Method gives priority on counting number of letters present in a line. So, probability in finding similar lines applying algo#1 will become narrow when we are applying algo#2. It helps in increasing distinct line count and as a result, resulting AVL tree gets an opportunity to increase it's height or better variation in alphabet choosing. Hence, group size of secret binary information gets increased automatically and it helps to store more information with minimal use of alphabets. We can apply algo#1 and algo#2 alternatively in the pages of a book for hiding binary information. It also enhances security mechanism.

We can merege algo#1 & algo#2 to algo#3, to make information hiding more strong and incorporating decoding strategy at the same timepaper.

REFERENCES

[1] International Journal of Advanced Science and Technology Vol. 35, October, 2011.

[2] International Journal of Network Security & Its Applications (IJNSA), Vol.5, No.1, January 2013

[3] Proceedings of the 3rd National Conference; INDIACom-2009Computing For Nation Development, February 26 – 27, 2009 Bharati Vidyapeeth's Institute of Computer Applications and Management, New Delhi

[4] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding", IBM Systems Journal, vol. 35, Issues 3&4, 1996, pp. 313-336

[5] K. Bennett, " Linguistic Steganography: Survey, Analysis, and Robustness Concerns for Hiding Information in Text ", Purdue University, CERIASTech Report 2004-13.

[6] S.H. Low, N.F. Maxemchuk, J.T. Brassil, and L. O'Gorman, "Document marking and identification using both line and word shifting", Proceedings of the Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '95), vol.2, 2-6 April 1995, pp. 853 - 860.

[7] A.M. Alattar, and O.M. Alattar, "Watermarking electronic text documents containing justified paragraphs and irregular line spacing", Proceedings of SPIE -- Volume5306, Security, Steganography, and Watermarking of Multimedia Contents VI, June 2004, pp. 685-695.

[8] Y. Kim, K. Moon, and I. Oh, "A Text Watermarking Algorithm based on Word Classification and Inter word Space Statistics", Proceedings of the Seven

[9] D. Huang, and H. Yan, "Inter word Distance Changes Represented by Sine Waves for Watermarking Text Images", IEEE Transactions on Circuits and Systems for Video Technology, vol. 11, no. 12, December 2001, pp. 1237-1245.

[10] M. H. Shirali-Shahreza, and S. Shirali-Shahreza, "A New Approach to Persian/Arabic Text Steganography", Proceedings of 5th IEEE/ACIS international Conference on Computer and Information Science and 1st IEEE/ACIS, June 2006.

[11] M. H. Shirali-Shahreza, and S. Shirali-Shahreza, "A Robust Page Segmentation Method for Persian/Arabic Document", WSEAS Transactions on Computers, vol. 4, Issue 11, Nov. 2005, pp. 1692-1698.

[12] J.A. Memon, K. Khowaja, and H. Kazi, "Evaluation of steganography for Urdu /Arabic text", Journal of Theoretical and Applied Information Technology, pp 232-237