WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Text Steganography's New Approach to Information Hiding

Arpita Mohapatra [1], Dr. G. Manivasagam [2]

Jain University

ABSTRACT

Frequent transmission of encrypted messages may draw the interest of outside parties, such hackers and crackers, and may result in attempts to decrypt and reveal the original messages. In the digital realm, steganography is used to mask a hidden message inside a dubious-looking message, so hiding its presence.

Often used in tandem with encryption, steganography offers a suitable degree of security and anonymity across the communication channel. This paper provides various current text-based steganography methods together with a brief history of steganography and an overview of text steganography. The main drawbacks of text steganography are outlined, along with certain problems with the current fixes. Information hiding is achieved by a hybrid approach that combines inter-word and inter-paragraph space. Depending on the length of the secret message, our technology produces dynamic stego-text with six possible maximum capacities. This paper also looked at each current strategy's main flaws and how our new method might be suggested as a fix.

INTRODUCTION

"Information concealment" encompasses a wide range of actions. Among the most significant subdisciplines is steganography. The word "steganography" (στεγανό-ς, γραφ-ειν) originates from the Greek work "Steganographia" by Johannes Trithemus (1462-1516) and means "covered writing." It's a time-honored technique of hiding information such that a message appears as harmless cover media and is not seen by an eavesdropper. Sending a message over a communication channel with an innocent carrier—such as text, image, audio, or video—is the aim of steganography. In contrast, technical steganography is described as a carrier rather than a text that can be communicated, like other physical media like invisible inks and microdots. New digital technologies have rendered it possible to improve message detection, allowing messages to carry more information and seem less obvious during transmission. An instance of this is the Germans' microdots technique. Microdots uses a technique called microscopic shrink to hide text pictures that are visible only under a microscope. They were used by German spies in several forms, such as hiding messages in letters, on watch faces, and even on ties with spots, as seen in. This study presents a novel approach to text steganography by creating a hybrid technique that makes use of the whitespaces in text right-justification between words and paragraphs. This method is an improvement over the free-form method since it does not rely on a single data encoding methodology. Combining inter-word spacing and inter paragraph spacing into a single embedding process yields an increased capacity for embedding hidden information.

Literature Review

A imitation algorithm for text was suggested by Peter Wayner in his work Mimic Functions, Cryptologic XVI-3. His method is to create text that is replicated and appears to have the original text's actual structure. Steg text was created by Peter Wayner using a set of grammatical rules; the encoding of each word controls the bits of the secret message that are hidden. Since static grammar is the foundation of the grammatical rules, the grammars must be created before the algorithm can be applied. It appears that the algorithm produces structures devoid of context. The user of the system must create the grammar he wants the text to emulate. The caliber of the grammar impacts the caliber of the generated Steg text. The grammar serves as a key to unlock data hiding. Another website that shows imitation is spam mimic. The syntax used in this Wayner's system implementation imitates the style of spam. The benefit of this application, according to the inventor, is that a secret message can be disguised as spam so that no one will be aware that a secret message is being kept hidden. This benefit remains by our suggested approach, but the cover text—which is taken from benign nursery rhymes—has a different content. It produces cover text that incorporates commercial motives, in contrast to Spam mimic.

In their paper, Brassil et al. suggested features coding (character coding), life-shift coding, and word-shift coding as ways to prevent unapproved distribution of papers shared via computer networks. By shifting text lines vertically, a technique known as "line-shift coding" allows a document to be uniquely encoded. Using a technique called word-shift coding, text can be individually encoded in a document by shifting words inside text lines vertically. Character coding, sometimes referred to as feature-specific coding, is a type of coding that is exclusive to the bitmap image included in the document. It makes it possible to examine specific character properties, which are then altered or left unaltered depending on the codeword. Documents are labelled unidentifiably with a codeword that indicates the registered owner to whom the document is transferred. If a document copy is found to have been disseminated illegally, it can be decrypted, and its rightful owner located.

In a study on data hiding techniques in text, Bender et al. considered three different encoding methods. The three approaches are syntactic, which makes use of punctuation, free space, which manipulates whitespace, and semantic, which manipulates synonymous terms to encode. In justified text, open space algorithms determine the inter-sentence, end-of-line, and inter-word spacing. By adding one or two spaces to the end of each termination character, the inter-sentence spacing approach encodes a binary message into a text. It adds one space to encode a "0," and two spaces to encode a "1." This approach works; its only drawback is that it is inefficient because it needs a lot of text to be encoded. Assuming sentences consist of two lines of text with an average length of 80 characters, one bit per sentence corresponds to a transfer rate of roughly one bit per 160 bytes. This approach is entirely dependent on the text's organization. Empty spaces at the finish of each line are exploited by the end-of-line spacing approach. Data that is encoded with a fixed number of spaces at the end of each line. One bit can be encoded with two spaces, two bits with four spaces, three bits with eight spaces, and so on. It works better than the inter-space method because more data may be buried by employing more spaces. The right-justification of text is a third technique for encoding data using whitespaces that can be applied to text files. By adjusting the placement of the extra spaces, data can be encoded. A single space is read as a "0" in between words. A "1" is represented by two spaces. It is discovered that because of the limitations on justification, not every interworld space can be used as data. To distinguish between interworld spaces that contain concealed data bits and those that are part of the original text, Bender et al. used an encoding technique akin to that of Manchester. One interprets "01" as "1" and "10" as "0." The bit strings “00” and “11” are null.

Objectives

The objective of this project is to create a system that enables users to utilize the hybrid inter-word and inter-paragraph spacing mechanism to conceal a hidden message within a text. Depending on the length of the secret message, the system should be able to dynamically construct stego-text with six possibilities for maximum capacity. The secret message and the text to be used as the cover text should be submitted by the user through an interface built within the system. The secret message should then be hidden within the cover text by using the inter-word and inter-paragraph spacing techniques, and the system should produce stego-text that contains the hidden message. The user should receive the stego-text output from the system.

By examining the spacing patterns and deciphering the encoded message, the system ought to further extract the secret message from the stego-text.

Python should be used to implement the system, together with the proper data structures and object-oriented programming concepts. To guarantee accuracy and resilience, the system needs to be thoroughly tested and documented.

Proposed Methodology

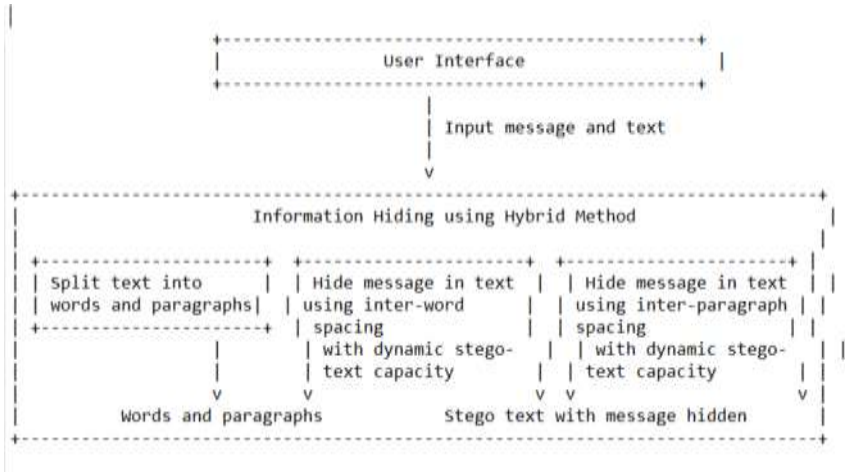
We introduced a brand-new text steganography technique that hides information by utilising spaces between words and paragraphs. The unique feature of the approach is that it gives the user six options depending on how long the hidden message is, and it generates a cover-text dynamically. The goal of future research should be to strengthen the decoding algorithm's resistance. This is because word processing software will remove the gaps, erasing the hidden data. In addition, it is essential to maximise the embedded scheme's capability by considering different compression techniques.

System architecture

The user enters the message to be hidden and the text to be used for information concealing through the system's user interface. Next, the "Split text into words and paragraphs" module is used to divide the text into words and paragraphs.

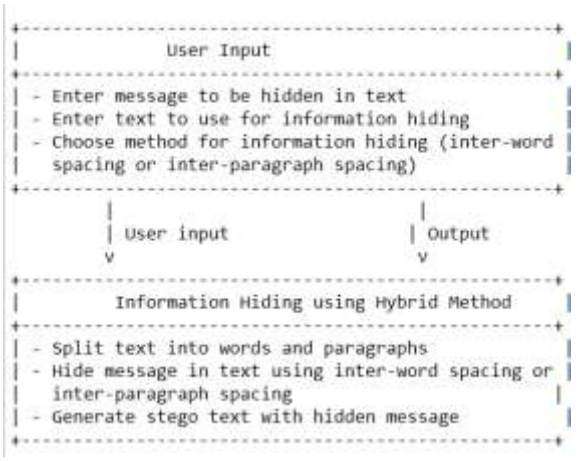
Afterwards, the user can choose to conceal the message within the text by utilising either inter-word spacing or inter-paragraph spacing. A dynamic stego-text capacity is used by the "Hide message in text using inter-word spacing" and "Hide message in text using inter-paragraph spacing" modules to guarantee maximum capacity based on the length of the secret message.

The result is a stego text with the selected method used to conceal the message. With the use of a hybrid technique that blends inter-word and inter-paragraph spacing, users can easily conceal messages in text thanks to the system architecture.



Use case diagram

The user's interaction with the "Information Hiding using Hybrid Method" system is depicted in the use case diagram. The information to be hidden, the text to be used, and the information hiding method (inter-word or inter-paragraph spacing) are provided by the user.

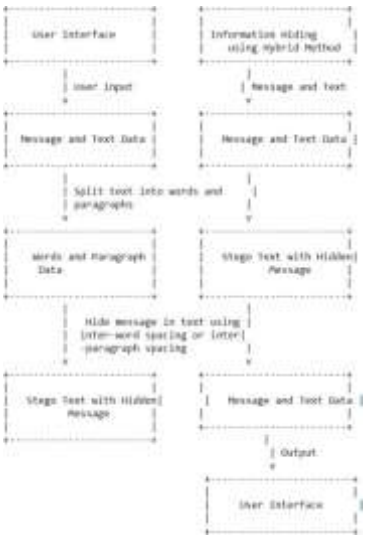


Data flow diagram

The data flow between the "User Interface" and "Information Hiding using Hybrid Method" modules is depicted in the data flow diagram.

The text data and message are entered by the user and sent to the "Information Hiding" module. The text is divided into words and paragraphs by the "Information Hiding" module, which then uses inter-word or inter-paragraph spacing to conceal the message.

The concealed message in the generated stego text is then sent back to the user interface for output. The system's modules' interactions and the data flow among them are depicted in the data flow diagram.

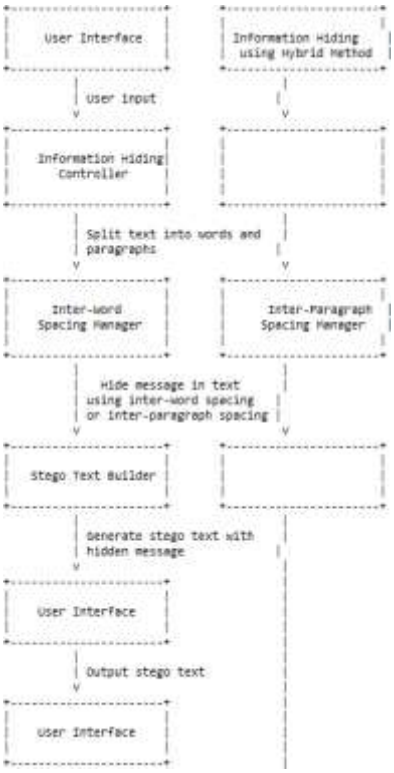


Sequence diagram

The interactions and flow of events between the user interface and the different system modules are depicted in the sequence diagram.

Input from the user is received by the system and sent to the "Information Hiding Controller" module. Depending on the approach selected, the "Information Hiding Controllers" module communicates with either the "Inter-Word Spacing Manager" or the "Inter-Paragraph Spacing Manager" module.

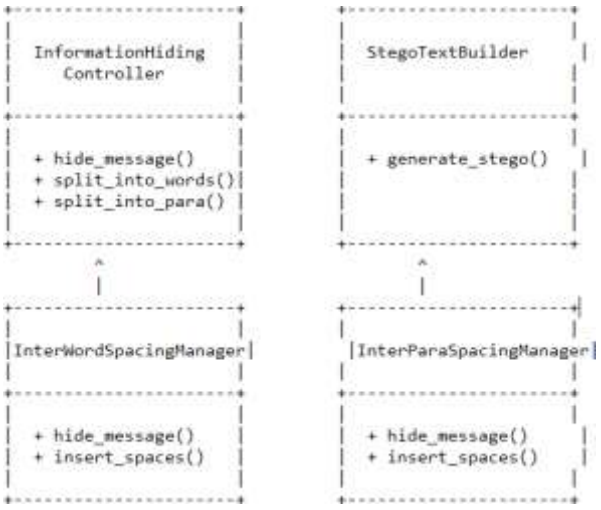
The "Inter-Word Spacing Manager" or "Inter-Paragraph Spacing Manager" module then hides the message in the text, and passes the resulting words and paragraphs data to the "Stego Text Builder" module. The "Stego Text Builder" module generates the stego text with the hidden message, and passes it back to the user interface for output.



Class diagram

The classes participating in the information hiding system are depicted in the class diagram, along with their linkages and workings. The information hiding procedure is coordinated by the "InformationHidingController" class. It offers techniques to divide the text into words and paragraphs and to use the inter-word or inter-paragraph spacing strategy to conceal the content.

The stego text with the concealed message is created by the "StegoTextBuilder" class. The stego text is generated using a single process. Using their corresponding methods, the "InterWordSpacingManager" and "InterParaSpacingManager" classes oversee hiding the message. Each of them has a way to add spaces when needed and conceal the information. All things considered, the class diagram aids in revealing the arrangement and connections among the classes within the system.



References

1. https://www.hindawi.com/journals/mpe/2022/4641559/
2. https://github.com/vijaywargiya/ipmanagement-blockchain
3. https://scholar.google.com/

S. Nakamoto, “Bitcoin: A Peer-To-Peer Electronic Cash System,” 2008, https://papers.ssrn.com/sol3/papers.cfm? abstract_id=3440802.

V. Buterin, “A Next-Generation Smart Contract and Decentralization Application Platform,” White Paper, vol. 3, 2014.

C. Cachin, “Architecture of the Hyperledger blockchainfabric,” in Proceedings of the Workshop Distributed Cryptocurrencies and Cons (Csensus) Le, IBM Research, Chicago, IL, USA, June 2016.

R. G. Brown, J. Carlyle, I. Grigg, and H. Mike, Corda: An Introduction, Wiley-blackwell, Hoboken, NJ, USA, 2016.

H. M. Corda, A Distributed Ledger, Blockchain, Luxembourg, Luxembourg, 2016.

T. McConaghy, R. Marques, and A. M¨uller, BigchainDB: Ascalable Blockchain Database, Blockchain, Luxembourg, Luxembourg, 2016.

Beijing Peer Safe Technology Co Ltd, White Paper for Blockchain Data Base Application Platform, Beijing Peer Safe Technology Co Ltd, Beijing, China, 2017.

T. Fit, Whitepaperfortencent Trust SQLTencent Research Institute, Beijing, China, 2017.

Y. Yuan and F.-Y. Wang, “Blockchain: the state of the art and future trends,” Acta Automatica Sinica, vol. 42, no. 4, pp. 481–494, 2016.

P. He, Y. Ge, and Y. F. Zhang, “Survey on blockchain technology and its application project,” Computer Science, vol. 44, no. 4, pp. 1–7, 2017.