Auto-Summarization-Based Steganography

Abdelrahman Desoky, Mohamed Younis
Dept of Computer Science and Electrical Engineering
University of Maryland, Baltimore County
Baltimore, Maryland, USA
(abd1, younis@umbc.edu)

Hesham El-Sayed
Department of Computer Science
United Arab Emirates University
Al-Ain, United Arab Emirates
(helsayed@uaeu.ac.ae)

Abstract

Steganography is the science and art of avoiding the arousal of suspicion in covert communications. This paper presents a novel steganography methodology that pursues text summarization in order to hide messages. The proposed Summarization-Based Steganography (Sumstega) methodology takes advantage of recent advances in automatic summarization techniques to generate a text-cover. Sumstega does not exploit noise (errors) to embed a message nor produce a detectable noise. Instead, it pursues the variations among the outputs of auto-summarization techniques to conceal data. Basically, Sumstega manipulates the parameters of automatic summarization tools, e.g. how the word frequency weights in the sentence selection, and employs other contemporary techniques such as paraphrasing, reordering, etc., to generate summary-cover that looks legitimate. The popular use of text summaries in business, science, education, news, etc., renders summary an attractive steganographic carrier and averts an adversary's suspicion. The validation results demonstrate the effectiveness of Sumstega.

1. Introduction

Linguistic steganography is the scientific art of avoiding suspicion in covert communications by concealing data in a textual cover. When using any steganographic technique if suspicion is raised, the goal of steganography is defeated regardless of whether or not a plaintext is revealed. Contemporary linguistic steganography approaches found in the literature are not fully capable of passing both computer and human examinations. Such shortcoming is attributed to the fact that these approaches may introduce detectable flaws (noise), such as incorrect syntax, lexicon, rhetoric, grammar, and the content of the linguistic-cover may be meaningless and semantically incoherent. Obviously, these detectable flaws can raise suspicion during covert communications unless there is a legitimate excuse such as flaws made by a person with a speech or writing impediment. Not enough attention is given to these issues.

A Summarization-Based Steganography Methodology (Sumstega) is presented in this paper. Sumstega employs automatic summarization techniques to camouflage a message. The aim of the automatic summarization is to represent the core contents of a long document(s) in a significantly shorter version with minimal human intervention. The process of summarization is also called alteration [1] because it produces a significant smaller and possible different output format from its input. The use of summaries in business, science, education, news, World Wide Web, etc., has become increasingly popular because people often will less likely read verbose documents unless necessary. A summary that is focused on key points tend to attract attention and suffice in many contexts.

In general, automatic summarization techniques alter the input document to generate the required brief version. Sumstega takes advantage of this process to camouflage data in the generated summary. Basically, Sumstega manipulates the parameters of automatic summarization tools, e.g. how the word frequency weights in the sentence selection, and employs other contemporary techniques such as paraphrasing, reordering, etc., in order to embed a message without violating the pattern of an ordinary summary. Thus, Sumstega does not exploit noise (errors) to embed a message nor produce a detectable noise. For example, Sumstega may identify possible variations of legitimate summaries generated by multiple tools, and then embeds data by substituting a set of elements, e.g., sentences, words, etc., of a particular summary with other legitimate elements from peer summaries.

Some of the main advantages of the Sumstega methodology over published approaches are as follows. The popularity of text summaries allows the communicating parties to establish a covert channel to transmit the hidden message. The tremendous amount of summarized text in electronic and non-electronic format makes it impossible for an adversary to investigate all of them. This makes a summary extremely favorable as a steganographic cover in covert communications. In addition, Sumstega is resilient against contemporary attacks including an attack by an adversary who knows Sumstega, i.e.,

978-1-4244-3397-1/08/$25.00 ©2008 IEEE

Sumstega is a public methodology. Basically, the adversary will not be able to distinguish between summaries that have and do not have some concealed messages. Moreover, in Sumstega the hidden message is anti-distortion. Distortion may destroy the hidden message entirely or partially. For example, a text-cover that hides data in the text format, e.g. bold, colors, fonts, etc., distortion may be in the form of wrong interpretation or processing of the formatting escape-character sequence. In Sumstega, the message cannot accidentally get distorted without altering the text.

The remainder of this paper is organized as follows: Section 2 briefly provides some background and related work discussion; Section 3 introduces the Sumstega methodology; Section 4 demonstrates implementation of Sumstega; and Section 5 concludes the paper and highlights directions for future research.

2. Background and Related Work

This section presents a brief overview of automatic summarization systems and a review of prior work on linguistic steganography that are related to Sumstega.

2.1 Automatic Summarization

The field of automatic summarization has enjoyed significant advances in recent years and is still promising more in the future. Automatic summarization systems employ a procedure that may be based on one or more of the following: statistical measures, knowledge base, artificial intelligence, computational linguistics, and other related techniques to achieve its goal [1][2]. Some examples of automatic summarization systems are AutoSummarize [3], SweSum [4], Inxight Summarizer [5], DimSum [6], Objects Search [7] etc. Automatic summarization approaches may be categorized into three types: high level, low level, and hybrid approaches.

High level summarizers, also referred to as shallow approaches, mainly employs sentence extraction and reordering techniques to capture the essence of the document and generate a readable text. Most of these approaches produce a summary that is a subset of the original document and are thus easy to implement. On the other hand, low-level summarizers, which also are referred as deep approaches, employ knowledge base and other related techniques, such as artificial intelligence and natural language generation, in order to produce an abstract of the text. Obviously, low-level summarizers are significantly more sophisticated than high level ones and involve some implementation complexity. Hybrid approaches do exist and are often used for generating a summary of multiple documents.

Due to space constraints only high-level approaches are used to illustrate the implementation of Sumstega.

2.2 Linguistic Steganography

Contemporary steganography approaches are usually categorized based on the cover type such as text, image, or audio. Textual steganography can be further classified as Textual Format Manipulation (TFM) and Textual Fabrication (linguistic steganography) [8]. In TFM, comparing the original text with the modified text will reveal the hidden message. On the other hand, linguistic steganography techniques generate an entire text-cover for hiding a message rather than manipulating an existing text. The most notable approaches are null cipher [9], mimic functions [10], NICETEXT and SCRAMBLE [11][12], and translation-based [13].

A null cipher is a predetermined protocol of character and word sequence that is read according to a set of rules such as read every seventh word or read every ninth character in a message [9]. Apparently, suspicion is raised because the user is forced to fabricate a text-cover according to a predetermined protocol that is not legitimate. Applying a brute force attack may reveal the entire message. Mimic functions [10], on the other hand, employs the inverse of the Huffman Code by inputting a data stream of randomly distributed bits. Although the text generated by mimic functions is shown to be resilient to statistical attacks, the output text is gibberish rendering it extremely suspicious. Context Free Grammars (CFG) and van Wijnaarden grammars have been employed to enhance the readability of the output text of mimic functions. However, the text is still nonsense, full of syntax errors, and semantically erroneous.

NICETEXT and SCRAMBLE pursues text substitution at the word-level, using a large dictionary [11][12]. However, suspicion is raised because some synonymous words are not semantically compatible [13]. Furthermore, if the adversary has the original text and semantically analyzes it, he may detect a fingerprint of NICETEXT because of the reuse of the same piece of text. Finally, Christian Grothoff et al., in 2005, introduced the translation-based scheme, which hides a message in the translation errors (noise) that are naturally generated by a machine translation. The major problem with the translation-based scheme is that the improvement of machine translations will increase the possibility of suspecting a hidden message [13] and will thus make it unattractive approach.

Sumstega overcomes the deficiencies of these approaches. Unlike the translation-based approach, Sumstega does not use noise (errors) in concealing a message. In addition, Sumstega does not create noise that causes an adversary to suspect the existence of a hidden message in a summary-cover, as will be demonstrated later. Basically, any linguistic error that

may exist in a summary-cover is caused by the auto-summarizer rather than Sumstega. Therefore, further improvements in automatic summarization systems will render Sumstega a more resilient approach.

3. Sumstega Methodology

The main idea of the Sumstega methodology is to exploit the variations among the outputs of auto-summarization techniques to conceal data. Basically, Sumstega manipulates the parameters of automatic summarization tools, e.g. how the word frequency weights in the sentence selection, and employs other contemporary techniques such as paraphrasing, reordering, etc., to generate summary-cover that looks legitimate. The popular use of text summaries in business, science, education, news, etc., renders summary an attractive steganographic carrier and averts an adversary's suspicion.

To illustrate how Sumstega can be used, consider the following scenario. Bob and Alice are on a spy mission. Before they start their mission, which requires them to reside in two different countries, they set the rules for communicating covertly using their professions as a justification. To make this work, they establish a business relationship as follows. Bob and Alice are journalists working for the same corporation, and they agree to use Sumstega. They generate summaries of real news scripts to make their covert communications more legitimate. When Bob wants to send a covert message to Alice, Bob either posts summarized articles online for authorized clients and staff to access or he sends them via email. These Sumstega-prepared summaries conceal a message. Covert messages transmitted in this manner will not look suspicious because Bob and Alice are journalists and their interaction is legitimate. The use of an auto-generated summary in such a profession is natural given the constraints on publishing space and the little time a reader may allocate for reading. Furthermore, Bob and Alice are not the sole recipients. There are other non-spy journalists, staff, and clients who send and receive such articles, further warding off suspicion. However, only Bob and Alice will be able unravel the hidden message because they know the rules of the game. When Alice or Bob communicate, they use real news data from their professions and their established business relationship to legitimatize their interaction.

Sumstega is composed of three modules whose ultimate goal is to define a configuration for the communicating parties to use. The first module mainly determines Sumstega encoding parameters, meaning what aspect of the auto-summarization process would be used to hold steganographic code. These parameters are then used by the second and third modules to define a message encoder and a camouflage scheme, respectively. Figure 1 shows the interaction among the Sumstega modules and how the steganographic summarizer is used by the sender and recipient. The following subsections explain the Sumstega modules.

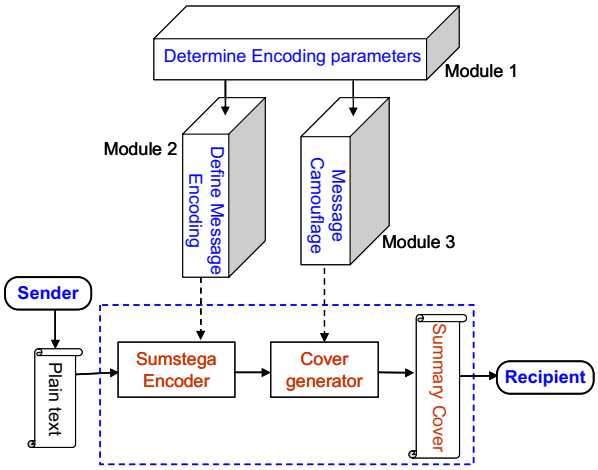

Figure 1: An illustration of the interaction of the various Sumstega modules and how the outputs of the individual modules are used for covert communication.

3.1 Determining Encoding Parameters

This module (Module 1) is responsible for generating a Sumstega configuration that the sender and receiver must agree on so that the hidden message can be extracted. There are numerous parameters to the auto-summarization process that can be exploited as a vehicle for concealing a message. A parameter in this context means some input value that a user may set to shape out the generated summary. Examples of these parameters include the desired upper and lower bound on the size of the summary, how word frequency weights in the sentence selection, reduction ratio compared to the original text, etc. Despite the feasibility of controlling these parameters by the sender and the determination of their values at the receiver, one would argue that only very short messages that are of few bits may be concealed. In order to support relatively longer messages, Sumstega employs a novel technique by employing multiple auto-summarizers or different implementations of the same auto-summarizer.

In general, automatic summarization systems employ numerous methodologies such as extraction, abstraction, semantic equivalence, information equivalence, etc., to generate summaries [1]. The implementation of each of these methodologies may further involve one or multiple techniques such as statistical analysis, knowledge base, artificial intelligence, and computational linguistics. Such diverse set of methodologies/techniques usually yields

summaries that differ in the sentences selected from the original text, phrasing of sentences that convoy the same meaning, order of sentences, etc. This module of Sumstega identifies a mix of these methodologies and/or techniques to generate multiple distinct summaries. The picked mix is considered a part of the Sumstega configuration that a sender and receiver must agree on. As will be explained shortly, Sumstega will use the internally generated summaries to camouflage the data in a summary-cover.

3.2 Message Encoder

Sumstega creates an encoded representation of a message and then camouflages it in a summary cover. The obvious constraint that Sumstega imposes on the message encoder (Module 2) is to generate steganographic code that can be embedded in the cover. For example, when a small message is to be concealed using the reduction ratio pursued during the auto-summarizer, the encoded message has to be in the range [0, 99]. In fact, practically the range may be constrained by the size of the document to be summarized, e.g. it may be meaningless to have a summary that is 95% of a large document. While the encoding of long messages will be less constrained, it still has to factor in the encoding parameters picked by Module 1, as illustrated shortly.

Given the availability of numerous encoding techniques in the literature that fit [14], the balance of this section will focus on an example that illustrates how to meet the message encoding constraints. This example will be used in Section 4 to demonstrate the applicability of Sumstega. In the example, the encoding is done as follows. A message is first converted to a binary string. The string can be a binary of cipher text or a compressed representation. The binary string is then partitioned into groups of m bits. The value of m is determined based on the number "n" of different high-level summaries that are produced, as specified by the encoding parameters (Module 1). Basically, m is set to log n. If n= 4, i.e., four different summaries, the bit pattern 00, 01, 10, or 11 will be implied if a sentence in the summary-cover uniquely matches that of the first, second, third or fourth internally-generated summary, respectively. Multiple matches imply null data bits. Again, this encoding scheme is just for illustration and many alternate, and more sophisticated, schemes can be employed.

3.3 Message Camouflage

Sumstega camouflages data in a summary-cover. Basically, the sender will pick an original document that the receiver has access to. Sumstega employs contemporary auto-summarization techniques to generate the summary-cover. Based on the encoding parameters, which depends on the size of the message, Sumstega may involve one or multiple auto-summarizers. Obviously, the receiver has to be aware of the name and configuration of the auto-summarizer(s) used in generating the summery-cover to make sure the concealed message can be correctly extracted. As indicated earlier, very small messages can be concealed using the configuration parameters, e.g. reduction ratio. In that case, only one summary is internally generated.

For long messages the camouflage process is different. Sumstega internally generate multiple summarizes using the set of auto-summarization techniques picked by Module 1. To conceal a message, Sumstega exploits the differences and similarities among these summaries. The key idea is to mix sentences, phrases and words from the various summaries based on the encoded message. For example, if four auto-summarizes are involved (n=4 and thus m=2) and the first group of bits in the binary string is "10", Sumstega looks for the first sentence, phrase or word in the third summary that is distinct from all other summaries and includes it in the generated summary-cover. The selection of sentences, phrases or words is determined by Module 1 and again has to be agreed upon by the communicating parties. Section 4 shows a detailed example.

Finally, Sumstega may perform some fine tuning to enhance the quality of the summary-cover in terms of flow, injunctions, etc.

4. Sumstega Implementation

This section demonstrates an example of an actual implementation of the Sumstega methodology. In the example, the letter "X" is to be concealed. The ASCII representation, which is "01011000" for the letter "X", is used to form a binary string. Sumstega employs four publicly-accessible auto-summarization tools [3][4] [7][15] that are capable of generating dissimilar high-level summaries, i.e., they apply different sentence extraction techniques. The four auto-summarizers are assigned the code 00, 01, 10, and 11, respectively. The news article [16] is then picked as a base document for which summaries are generated using these tools. The summaries are not included due to space constraints. The cover generated by Sumstega is shown in Figure 2. Table 1 shows a subset of the coding table. Basically, all combinations for the similarity among the corresponding sentences in the various summaries considered by Sumstega are listed and each combination is assigned a binary code that will be implied when a particular sentence (underlined in bold in Table 1) is included in the summary-cover. Given the size of the complete table, only entries relevant to this example are shown.

"London was on high alert on the morning that police surveillance teams stationed outside an apartment block in South London spotted de Menezes leaving his building on his way to work. The police were looking for Hussain Osman, whose address was in the same building as de Menezes, and had attempted to bomb the Shepherd's Bush London tube station the previous day. Terror attacks on July 7 had killed 52 commuters, and just the previous day, more suspects had gone on the run after devices they planted on London's public transport network failed to explode. Amid those basic facts, a host of questions have arisen about police procedures under pressure and their response to finding themselves under investigation. When he boarded a train, they cornered him and shot him with special bullets designed to cause maximum physical damage. The court's verdict leaves the Metropolitan Police facing a penalty and legal costs that together amount to about $1.1 million."

Figure 2: The cover-text generated by Sumstega using four different extraction-based auto-summaries.

To camouflage the binary string "01011000", Sumstega compares the four summaries and identifies the first sentence in the summary-cover. Since all summaries start with the same sentence, it is included in the summary-cover implying null data. The second sentences in the 1st and 2nd summaries are similar and both are different from those in the 3rd and 4th summaries. According to Table 1, which is agreed upon by both the sender and the receiver, including this sentence implies a "0". The third sentences of the 2nd and 3rd summaries are similar; including this sentence in the summary-cover implies "1". The 4th, 5th and 6th sentences in the summary-cover are picked from 2nd, 3rd and 1st summaries respectively. These sentences are distinct, implying "01", "10" and "00", respectively.

Table 1: The steganographical code for some combinations of sentence-level similarity among the different summaries. The sentence that will be included in the cover is underlined. For example when all sentences are distinct the code depends on the picked sentence. When summaries #1 and #2 has the same sentence the code is "0" when this sentence is picked.

Code	Summary #1	Summary #2	Summary #3	Summary #4
00	***S1***	S2	S3	S4
0	***X***	***X***	S3	S4
…	…	…	…	…
01	S1	***S2***	S3	S4
1	S1	***X***	***X***	S4
…	…	…	…	…
10	S1	S2	***S3***	S4
…	…	…	…	…

5. Conclusions

In this paper, Sumstega, a novel methodology for steganography, has been presented. Sumstega achieves legitimacy by basing the camouflage of a message on auto-summarization of documents. Messages are neither concealed as noise (errors) nor cause a detectable noise. Instead, Sumstega pursues the variations among the auto-summarization techniques to conceal the data. The popularity of automatic summarization has been on the rise in business, science, World Wide Web, education, news, etc., rendering document summaries an attractive steganographic carrier. Currently, Sumstega is being subjected to extensive steganalysis. The validation results will be reported in the near future.

References

[1] Mani, I., *Automatic Summarization*, John Benjamins Publishing Company, 2001.

[2] Jones, K. S., "Automatic Summarising: The State of the Art," *Info. Processing Mgmt,* 43(6), pp. 1449–1481, 2007.

[3] Microsoft Word 97, built-in AutoSummarize.

[4] Hassel, M. and Dalianis, H., SweSum - Automatic Text Summarizer, http://swesum.nada.kth.se/index-eng-adv.html.

[5] www.inxight.com/products/sdks/sum

[6] SRA Corporation, DimSum Summarizer: http://sra.com

[7] http://www.objectssearch.com/summary/index.jsp

[8] Bennett, K., "Linguistic Steganography: Survey, Analysis, and Robustness Concerns for Hiding Information in Text," CERIAS Tech Report 2004-13, Purdue Univ, 2004.

[9] Kahn, D. *The Codebreakers: The Story of Secret Writing*, revised ed. Scribner, December 1996.

[10] Wayner, P., "Mimic Functions," *Cryptologia*, Vol. XVI/3, pp. 193-214, 1992.

[11] Chapman, M., and Davida, G., "Hiding the Hidden: A Software System for Concealing Ciphertext as Innocuous Text," in the *Proceedings of the International Conference on Information and Communications Security*, Beijing, P. R. China, November 1997.

[12] Chapman, M., et al., "A practical and Effective Approach to Large-Scale Automated Linguistic Steganography," in the *Proceedings of the Information Security Conference (ISC'01)*, Malaga, Spain, 2001.

[13] Grothoff C., et al., "Translation-based steganography." In the *Proceedings of Information Hiding Workshop (IH 2005)*, Barcelona, Spain, June 2005.

[14] Koblitz, N., *A Course in Number Theory and Cryptography,* 2nd Ed., Springer, pp. 54-76, 1994.

[15] LTRC, IIIT, Automatic Text Summarizer: http://search.iiit.net/~jags/summarizer/index.cgi

[16] TIME Magazine: www.time.com/time/world/article/0,8599,1679108,00.html