Emoticon-based Text Steganography in Chat

Zhi-Hui Wang
School of Software
Dalian University of Technology
Dalian, Liaoning, China
E-mail: wangzhihui1017@yahoo.cn

Chin-Chen Chang
Department of Information Engineering and Computer Science
Feng Chia University
Taichung 401724, Taiwan, R.O.C.
E-mail: ccc@cs.ccu.edu.tw

The Duc Kieu
School of Computer Science and Engineering
International University, Vietnam National University
Ho Chi Minh City, Vietnam
E-mail: ktduc@hmiu.edu.vn

Ming-Chu Li
School of Software
Dalian University of Technology
Dalian, Liaoning, China
E-mail: li_mingchu@yahoo.com

***Abstract*—Written text contains less redundant spaces which could be used for embedding, so embedding secrets into written texts is a challenging task. To break through the difficulty of text steganography, a text steganography in chat scheme was proposed in 2007. This scheme uses the abbreviations used in SMS-texting language to conceal secret data into chat sentences. As communications via chat room become more and more popular in people's lives, we propose another new text steganography method which is also used in chat rooms. The proposed scheme uses emotional icons (also called emoticons) to hide secret bits into the sentences used in chat. Due to the tremendous numbers of emoticons used in many kinds of chat rooms, this paper improves the embedding capacity of the previous method obviously. In addition, the experimental results show the easy-to-use character of the proposed scheme.**

Keywords-text steganography; linguistic steganography; chat room; emoticon; information hiding

I. INTRODUCTION

Information hiding is one of techniques used to protect sensitive information. Secret data can be concealed into different cover media such as images, videos, audios, and written texts. Embedding secrets into written texts is a challenging task because written text contains less redundant spaces which could be used for embedding. Information hiding techniques in text-domain can be classified into the following categories: synonym substitution, syntactic transformation, typographical errors, translation, and semantic transformation. To our best knowledge, the synonym substitution method was first proposed by Chapman and Davida [4]. Originally, this method aims to protect the privacy of cryptograms to avoid detection by censors. The implementation of this method is a software system called NICETEXT. The NICETEXT system transforms ciphertext into innocuous text which can be transformed back to the original ciphertext. A simple example is cited from [3] to demonstrate how the synonym substitution method works. Suppose that we want to embed three secret bits 101 into the cover sentence "Midshire is a wonderful little city". First, synonym sets S_1 and S_2 of the words wonderful and city, respectively, are generated by using WordNet [5]. These generated synonym sets are S_1 = {decent, fine, great, nice} and S_2 = {metropolis, town}. It is noted that the words in each synonym set are sorted in alphabetical order and indexed from 0. Second, the matrix representing a mixed-radix number is created as

$$\begin{pmatrix} a_1 & a_0 \\ 4 & 2 \end{pmatrix}$$

where $0 \le a_1 < 4$, $0 \le a_0 < 2$, 4 is the number of elements of S_1, and 2 is the number of elements of S_2. Next, the decimal value of the secret bits 101 is 5. Then, the number 5 can be expressed as $5 = 2 \times a_1 + a_0$. Thus, we have $a_1 = 2$ and $a_0 = 1$. This indicates that the cover sentence is encoded as "Midshire is a great little town". That is, the words wonderful and city are replaced with the words great and town, respectively, to embed three secret bits 101 into the above cover sentence.

A syntactic transformation method [6] modifies the grammar structures of sentences to hide a secret message. That means that this method paraphrases sentences by means of semantically equivalent sentence structures such as an active-passive transformation. The transformations used in this method can be facilitated by syntactic parsers. For example, the cover sentence is John wrote this book. We now want to conceal a secret bit b into this sentence. If b equals 0, then the cover sentence is kept unchanged. Otherwise, the cover sentence is changed to the stego sentence as this book was written by John. The problem with this method is that the passivization (i.e. the inverse of activization) can not be always performed for every sentence.

978-1-4244-4607-0/09/$25.00 ©2009 IEEE

The typographical error (also called typo for short) method [9] replaces a selected word with an ambiguous probable typo of it. For example, we want to embed a secret bit *b* into the cover sentence "This is the great book." Suppose that the word "the" is selected by using a secret key *K* for embedding *b*. If *b* equals 0, then the cover sentence remains unchanged. Otherwise, the cover sentence is changed to the stego sentence "This is teh great book." That is, the typing error has been injected into the cover sentence to generate the stego sentence.

The translation-based method [8] embeds secret bits into a cover text by exploiting frequent errors in automatic translation engines. In this method, the steganographic encoder translates the sentences in the source text into the target language. The steganographic encoder generates multiple translations for each sentence and selects one of these to conceal secret bits. The most challenging method is the semantic transformation method [1]. That is, the meaning of the cover text needs to be preserved after secret bits have been embedded. This goal can be achieved by using word sense disambiguation and semantic role parsers. Recently, the text steganography method in chat rooms has been proposed [7]. This method hides secret bits into the sentences used in chat rooms. The review of this method is presented in Section II.

The remaining of this paper is structured as follows. Section II reviews the text steganography method in chat rooms. The proposed method is detailed in Section III. The experiments and discussions are presented in Section IV. Some conclusions are made in Section V.

II. THE PREVIOUS WORK

The text steganography in chat was proposed in 2007 [7]. This scheme aims at hiding secret messages in text sentences used in chat rooms such as Windows Live Messenger (also called MSN for short) or Yahoo Messenger (also called YM for short). The scheme uses the abbreviations used in SMS-texting language [2] to conceal secret data into chat sentences. A part of SMS acronyms is shown in Table I. Initially, a list of abbreviations of word or phrases needs to be prepared. The list is a collection of abbreviations used in SMS-texting and abbreviated words used in dictionaries. The structure of the list is in the form of Table I. This list (also called the reference list) is pre-shared between a sender and a receiver when a chat session is started.

TABLE I. SOME ACRONYMS USED FOR SMS-TEXTING

Acronym	Translation
B4	Before
B4N	Bye for now
CM	Call me
CUL	See you later
EZ	Easy
F2F	Face to face
GR8	Great!
HRU	How are you
IC	I see
U	You

In this scheme, secret bits are embedded as follows. If the secret bit is 0, then the sender will use the complete form of a word or phrase taken from the reference list in the sentence sent to the receiver. Otherwise, the abbreviated form of a word or phrase will be used. For example, if the sender wants to embed two secret bits 10 into the sentence "Please call me when you feel upset," then this sentence is encoded as "Please CM when you feel upset." At the receiving side, the receiver searches each word of the received sentence in the reference list (e.g. the list shown in Table I). For each found word, if the word is in the complete form, then a secret bit is extracted as 0. Otherwise, a secret bit is extracted as 1. In this example, the word "CM" is found in the reference list and it is in the abbreviated form so the secret bit 1 is extracted. Next, the word "you" exists in the reference list and it is in the complete form so the secret bit 0 is extracted. This means that the receiver can successfully receive two secret bits 10.

With the purpose of improving the embedding capacity of the above method, we propose a new text steganography method which is also used in chat rooms. The proposed scheme uses emotional icons (also called emoticons) to hide secret bits into the sentences used in chat. The details of the proposed method are described next.

III. THE PROPOSED METHOD

This section presents a new text steganography method also used in chat. The proposed method embeds secret information into emotional icons (also called emoticons for short) in chat rooms over the Internet. The chatter can use these emoticons to express his feeling or mood more vividly than use words. Fig.1 shows a part of the emoticons used in an MSN chat room.

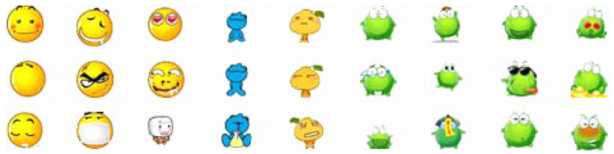


Figure 1. Some emoticons in MSN chat room.

When the chatters want to conceal the secret information, they should do the preparative step first. The detail of this procedure is shown in Fig. 2.

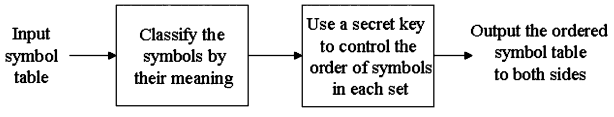


Figure 2. The flowchart of the preparative procedure.

At first, the sender's emoticon table should be unanimous with the receiver's emoticon table. This means all the emoticons in the sender's emoticon table are the same with the ones of the receiver's emoticon table, even the order of those emoticons should be the same. Next, the sender and

the receiver classify those emoticons in the emoticon table into several sets according to the meaning of every emoticon. The emoticons belong to the same set have the same meaning such as the 'cry' emoticon set, the 'smile' emoticon set, and the 'laugh' emoticon set, and so on. Every emoticon belongs to one set. Fig. 3 shows the 'cry' emoticon set in the MSN chat room. The order number of an emoticon, counting from 0, in its set is the secret bits that will be embedded. Thus, the proposed steganographic scheme uses a secret key to control the order of emoticons in each constructed set. Only the sender and the receiver keep this key. As we can see from Fig. 3, there are 16 cry emoticons in this set. Therefore, we can embed four bits per emoticon. Then, if the sender uses the second emoticon to express his cry feeling, the receiver can extract the secret bit as 0001 from the received message.

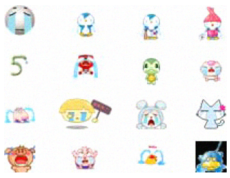


Figure 3. The 'cry' emoticon set in the emoticon table.

A. *The Embedding Procedure*

The embedding procedure is to find the correct emoticon according to the given secret message and then deliver it to the receiver by the chat tool. If the sender and the receiver use this approach to deliver the secret data, the receiver will try to make some opportunities for the sender to use the emoticons on purpose. Thus, when the sender is aware that he can use an emoticon to express his feeling in the chat with the receiver, he should find the corresponding emoticon set first. The second step is to count the number of emoticons in the found emoticon set. Let this number be N. Next, the number of secret bits that can be embedded into one emoticon is calculated by $n = \lfloor \log_2 N \rfloor$. At the third step, the sender obtains n bits of the secret data and transforms them into the decimal number d, where $0 \le d < N$. At last, the sender finds the dth emoticon in the found emoticon set and delivers it to the receiver.

In addition, the proposed scheme uses the following two methods to increase the hiding capacity of each emoticon from n bit to $n + 2$ bits. The first method uses the emoticon's position to embed one bit. Most of the time, chatters put the emoticon at the beginning or the end of a sentence. There is no difference between these two positions, so one secret bit can be embedded by putting an emoticon at different positions. On the other hand, the punctuation between words and an emoticon is not necessary. The existence of the punctuation does not influence the meaning of a sentence. Therefore, the second method uses this characteristic to increase the hiding capacity of one bit per emoticon.

B. *The Extracting Procedure*

The extracting procedure is a comparing procedure in this paper. When the receiver receives an emoticon, he finds the order number of this emoticon in the pre-shared emoticon table. Then, the receiver transforms this decimal number into a binary number, which are the secret bits sent by the sender.

The following is an example used to illustrate the embedding and the extracting procedures. Suppose the secret message to be embedded is '11100000111' and the classified emoticon table to be used is showed in Fig. 4. The following chat scenario will show how the secret message is delivered to the receiver. In this example, putting an emoticon at the beginning of a sentence indicates embedding a secret bit 0 and placing an emoticon at the end of a sentence implies embedding a secret bit 1. In addition, if there is the punctuation between words and an emoticon, this indicates that the secret bit 0 has been embedded. Otherwise, the secret bit 1 has been embedded.

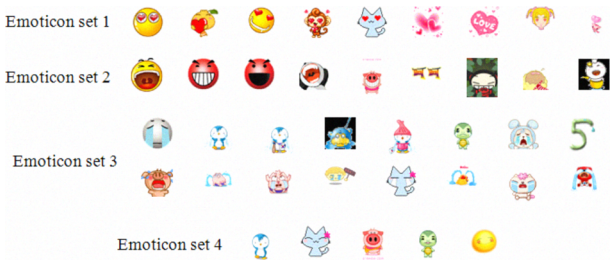


Figure 4. The emoticon table of the example.

Chat scenario:

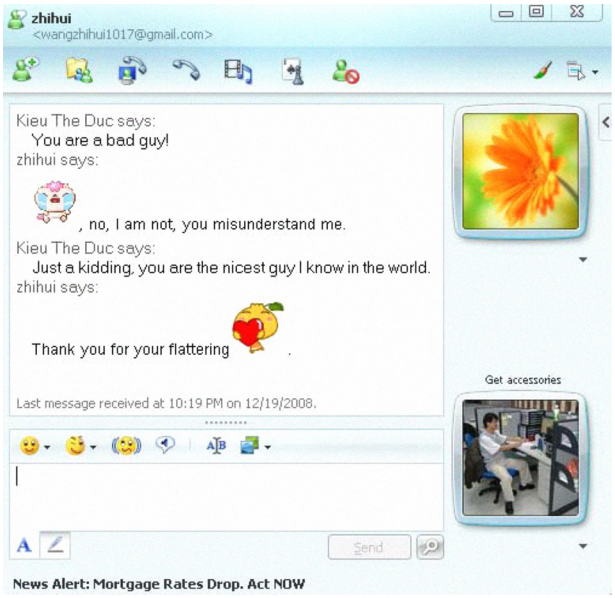
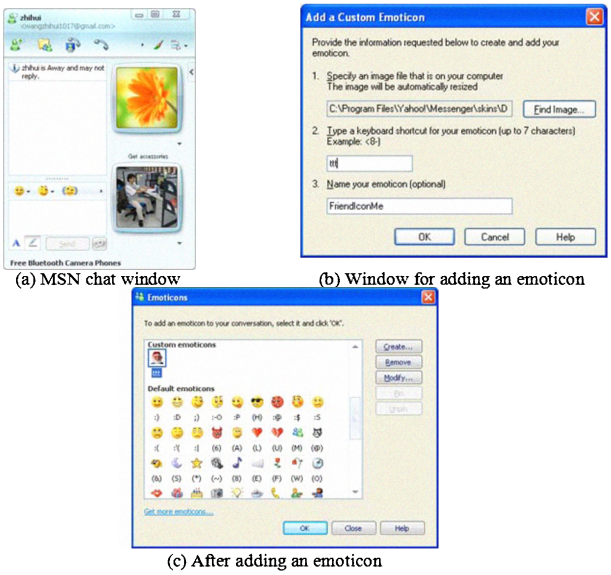


Figure 5. The chat scenario.

When the receiver gets the emoticon  as shown in Fig. 5, he will look for its order number in the emoticon set of the emoticon table. This emoticon is located at the position 14 in the emoticon set 3 so the receiver gets the decimal number d = 14. Then, the binary code of d is obtained as 1110. This emoticon is placed at the beginning of the sentence and there is a comma between the emoticon and the words. Thus, two secret bits 00 are also extracted. The same procedure will be done when the receiver gets the second emoticon  . At last, the receiver gets the secret message '11100000111'.

IV. Experiments And Discussions

It is clear that the pre-shared emoticon table and the number of elements in each emoticon set affect the hiding capacity of the proposed method. Different chat tools (e.g. MSN, Yahoo Messenger) have their own emoticons available to users for use. An emotional feeling can be expressed by different emoticons in different chat tools. For example, the smile feeling is indicated by  in YM and  in MSN, respectively. In addition, these chat tools have a functionality that allows chatters to customize (i.e. create, remove, and modify) the emoticons at their disposal. The aforementioned matters facilitate the creation of the pre-shared emoticon table and the grouping of similar emoticons into the emoticon sets. That is, the pre-shared emoticon table can be enlarged and the number of elements in each emoticon set can be increased. The adding of an emoticon in the MSN chat room is demonstrated in Fig. 6.



(a) MSN chat window (b) Window for adding an emoticon

(c) After adding an emoticon

Figure 6. Demonstration of adding an emoticon.

The chatter can click on the down arrow beside the icon on the window in Fig. 6(a) to display the window as shown in Fig. 6(b). From this window, the chatter selects the icon file and type in the shortcut (e.g. ttt), then clicks the button OK. The added emoticon is shown in Fig. 6(c).

The proposed method can be combined with the method in [7] to increase the hiding capacity. For example, the sentence "Please call me when you feel upset" can be used to embed secret bits 101110. The stego sentence will be "Please CM when you feel  ."

References

[1] M. Atallah, V. Raskin,, C. F. Hempelmann, M. Karahan, R. Sion, U. Topkara, and K. E. Triezenberg, "Natural language watermarking and tamperproofing," In Petitcolas, F.A.P. (Ed.), Proc. of the 5th International Workshop on Information Hiding, Noordwijkerhout, The Netherlands, Oct. 2002, pp. 196-212.

[2] K. Beare, "SMS-Texting," English as 2nd language [Online]. Available: http://esl.about.com/

[3] R. Bergmair, "Towards linguistic steganography: a systematic investigation of approaches, systems, and issues," Technical Report, University of Derby, Nov. 2004.

[4] M. Chapman and G. Davida, "Hiding the hidden: a software system for concealing ciphertext as innocuous text," Proc. of the First International Conference on Information and Communication Security, Beijing, China, Nov. 1997, pp. 335–345.

[5] C. Fellbaum, WordNet: an electronic lexical database, MIT Press, 1998.

[6] B. Murphy and C. Vogel, "The syntax of concealment: reliable methods for plain text information hiding," In Delp III, E.J., Wong, P.W. (Eds.), Security, Steganography, and Watermarking of Multimedia Contents IX, San Jose, CA, Feb. 2007.

[7] M. H. Shirali-Shahreza and M. Shirali-Shahreza, "Text Steganography in chat," Proc. of the Third IEEE/IFIP International Conference in Central Asia on Internet the Next Generation of Mobile, Wireless and Optical Communications Networks, Tashkent, Uzbekistan, Sep. 2007, pp. 1-5.

[8] R. Stutsman, M. J. Atallah, C. Grothoff, and K. Grothoff, "Lost in just the translation," Proc. of the 2006 ACM Symposium on Applied Computing, Dijon, France, 2006, pp. 338-345.

[9] M. Topkara, U. Topraka, and M. J. Atallah, "Information hiding through errors: a confusing approach in Delp III," E.J., Wong, P.W. (Eds.), Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, Feb. 2007, pp. 65050V.