http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 7 (4): 654-660, 2008
ISSN 1812-5638
© 2008 Asian Network for Scientific Information

MIMIC-PPT: Mimicking-Based Steganography for Microsoft Power Point Document

Yuling Liu, Xingming Sun, Yongping Liu and Chang-Tsun Li
School of Computer and Communication, Hunan University, Changsha, China

Abstract: Communications via Microsoft Power Point (PPT for short) documents are commonplace, so it is crucial to take advantage of PPT documents for information security and digital forensics. In this study, we propose a new method of text steganography, called MIMIC-PPT, which combines text mimicking technique with characteristics of PPT documents. Firstly, a dictionary and some sentence templates are automatically created by parsing the body text of a PPT document. Then, cryptographic information is converted into innocuous sentences by using the dictionary and the sentence templates. Finally, the sentences are written into the note pages of the PPT document. With MIMIC-PPT, there is no need for the communication parties to share the dictionary and sentence templates while the efficiency and security are greatly improved.

Key words: Text steganography, linguistic steganography, text mimicking, information security, digital forensics

INTRODUCTION

Communications via digital texts have long been a commonplace for personal, business, or academic purposes in these days and digital text has diverse forms, such as webpage, e-mail, various types of formatted text documents, including PDF, DOC, PPT and so on. Thus, it is convenient to transmit secret messages by using text documents as the mediums.

There are two main techniques to protect private communication of text documents. The first technique is cryptography, which encrypts a message to make it unintelligible to humans. Thus, those who do not possess the secret key cannot obtain the original message. Most researchers have made a great deal of effort on that. However, an encrypted communication always arouses suspicion (Petitcolas *et al*., 1999). The second technique is text steganography, which refers to the hiding of information within text documents (Murphy and Vogel, 2007). Unlike cryptography, the goal of text steganography is to convey secret messages in text documents, by concealing the existence of a covert communication (Bergmair, 2007).

Current implementations of text steganography exploit spacing flexibility in typesetting by making minute changes to the layout of different components and to the kerning in order to encapsulate hidden information. The key limitation of this approach is that it is vulnerable to simple retypesetting attacks. The other important method of text steganography is linguistic steganography based on the knowledge of natural language processing. It is much more ambitious, in that it should survive attempts to remove hidden information through file reformatting, OCR or retyping (Topkara *et al*., 2004). Publicly available methods of linguistic steganography can be grouped into two categories. The first group of methods, called text mimicking technique, is based on directly generating a new cover text for a given message. The second group of methods is based on linguistically modifying a given cover text in order to encode a message, while preserving the meaning as much as possible (Chiang *et al*., 2004; Topkara *et al*., 2006). Due to the sensitivity of modifying a given cover text, however, the amount of hidden information is limited. Therefore, this study falls in the former.

PPT is a presentation program developed by Microsoft for its Microsoft Office system. A PPT document is composed of one or more sheets of slides. Each sheet of slide in a PPT document may contain several text frames and a note page. All the text frames of a PPT document constitute the body text, while all the note pages are accessorial explanations, which are often ignored by careless readers and not visible to the audience when presenting. Therefore, the note pages provide a useful vehicle for hiding information in a PPT document. We can directly write encrypted information or whitespace characters into the note pages for the purpose of secret communication. However, it is difficult that the contents of the note pages are interrelated with the contents of the body text and resist attacks by humans or machines.

In this study, we propose a new steganographic method for hiding data in the note pages of PPT documents by utilizing text mimicking technique, called MIMIC-PPT. To provide an opportunity for deniability, we first create a dictionary table and a sentence template database by parsing the body text of a PPT document. Then we randomly select a sentence template and

Corresponding Author: Xingming Sun, School of Computer and Communication, Hunan University, Changsha, China

substitute parts-of speech for words in accordance with the assigned binary bits. The experimental results show that it is feasible to send a secret message in the note pages along with a PPT document. MIMIC-PPT is not only dictionary-free, but also can effectively generate meaningful sentences correlated with the body text to be written into the note pages of the PPT document.

In order to disguise cryptographic information as normal communications to thwart the censorship of ciphertext, it is necessary to introduce text mimicking technique, which converts ciphertext into text that looks innocuous natural language text. Publicly available implementations of linguistic steganography mainly rely on this technique.

The primary text mimicking method is proposed by Wayner (1992, 1995, 1997, 1999). In his basic mimicry algorithm, the method recodes a text so that its statistical properties of characters are more like that of another different natural language text. The text may fool attacks based upon statistical analysis, but it will not stand up to any analysis that understands the grammar structure. In order to improve the results, Peter Wayner proposes a method to generate texts using probabilistic context-free grammars and to hide information according to the choices it makes (Wayner, 2002). These generated texts are grammatically correct.

Another development in text mimicking is Stego (Walker, 1994), a mimicry method proposed by John Walke. By using a user-defined dictionary, Stego converts a binary file (secret message) into a text that resembles natural language. The text has structure, but does not comply with any grammar rule.

A later development in text mimicking is Texto (Maher, 1995), which includes a structs file that contains some usually-correct English sentence structures and a words file which contains 64 verbs, 64 adjectives, 64 adverbs, 64 places and 64 things. In order to facilitate exchange of binary strings, especially encrypted data, Texto can transform uuencoded or pgp ASCII-armoured ASCII data into English sentences.

A successful development in text mimicking is NICETEXT (Chapman and Davida, 1997a), a mimicry method proposed by Mark Chapman. NICETEXT is an improvement over Texto. The original NICETEXT approach generates a set of meaningful English sentences by large code dictionaries and sentence templates. In their dictionaries, almost 175,000 words are categorized into 25,000 types and within each type a word is assigned a unique binary code. Each sentence template contains a sequence of word-types. The encoder generates a text by randomly choosing a sentence template and selecting words for types in accordance with the assigned binary code. The challenges are to create large and sophisticated dictionaries and to create meaningful sentence templates (Chapman and Davida, 1997a, b). Later, Chapman *et al.* (2001) describes an extensible contextual template approach combined with a synonymy-based replacement strategy, so that more realistic text is generated. Chapman and Davida (2002) extends the NICETEXT protocol to enable deniable cryptography/messaging using the concepts of plausible deniability. In addition, El-Kwae proposes a new technique for hiding multimedia data in text, which is similar to NICETEXT. It introduces some marker types, which are special types whose words do not repeat in any other type. Each generated sentence must include at least one word from the marker types (El-Kwae and Cheng, 2002).

Different from the above text mimicking techniques, Sams Big G PlayMaker (PlayMaker for short) only utilizes normal sentence templates without a dictionary (Gmbh, 2000). In the system, each letter or symbol is corresponding to a normal sentence of a play book.

All the above methods are effective and they can generate cover texts directly. However, the texts produced by these methods are often implausible to human readers and it is unusual to transmit the texts between the communication parties. Moreover, these methods need a great amount of resources (both the time and effort) to design a sophisticated dictionary or a good predesigned grammar. On the other hand, the proposed MIMIC-PPT in this paper provides legitimate cases in using an existing PPT document. And there is no need to share the dictionaries and sentence templates between the communication parties. Furthermore, the generated text not only relates closely to the body text of the PPT document, but also simulates certain aspects of the writing style of the body text. Then the text is written into the note page, which is an intrinsic part of a PPT document and security is thus achieved.

MIMIC-PPT

Similar to other text mimicking techniques, such as the NICETEXT system, dictionaries and sentence templates are necessary in MIMIC-PPT. However, the dictionaries and sentence templates need not be transmitted between the communication parties. By utilizing existing linguistic tools, the senders and the receivers can automate the creation of a dictionary table and some sentence templates according to the following rules: Rule 1 and Rule 2. To make our description clear, two definitions are presented first as follows.

Definition 1: Content words are words that have meaning, such as nouns, verbs, adjectives, adverbs.

Definition 2: Function words are words that exist to explain or create grammatical or structural relationships into which the content words may fit, such as pronouns, prepositions conjunctions, determiners, interrogatives and so on.

Rule 1 (Dictionary table creation rule): First, we extract and segment the body text of a PPT document using the existing morphological analyzer (Toutanova *et al*., 2003; Zhang *et al*., 2005), which can fulfill the task of word segmenting and part-of-speech tagging. Then, we pick up all the content words to obtain a crude dictionary table, where each word and its part-of-speech are on a single line. The same words with the same parts of speech are merged in a line. We record their occurrences as an extra attribute. The basic form of the dictionary table consists of Part-of-speech, Word, Occurrences.

Rule 2 (Sentence template creation rule): For each sentence in the body text, we preserve function words and punctuations while replacing content words with parts of speech to obtain a sentence template.

MIMIC-PPT is divided into two processes, Hiding Process and Retrieving Process. During these two processes, there is no need to share dictionaries and sentence templates. Details of the hiding process and the retrieving process will be described later.

Hiding process: In order to hide a secret message in a PPT document with the text mimicking technique, the hiding process consists of three stages: a preprocessing stage, a generating stage and a writing stage. The preprocessing stage is to automatically create a dictionary table and some sentence templates according to Rule 1 and Rule 2 and to encrypt the secret message into a binary string. The generating stage is to convert the binary string into a set of innocuous sentences by utilizing the dictionary table and the sentence templates. The generated sentences are related to the body text of the PPT document. The writing stage is to write the sentences into the note pages of the PPT document to obtain a stego-document.

In the preprocessing stage, we introduce Rule 1 to create a dictionary table D and Rule 2 to automate constructing a sentence template database S, which is a set of sentence templates. The dictionary table D includes all the content words in the body text, while the sentence template database is a set of sentences with function words, punctuations and parts of speech of content words.

A secret message M is encrypted to get an m-bit binary string $C = c_1c_2...c_m$, where each c_i is a bit. Because it is unlikely that m equals the number of bits required to terminate the generated sentence at the end of a sentence template, or the end of a word, the length of message is added in front of C and strings of random 0's and 1's are appended to the end of C. That is, we hide into the PPT document a binary string $C = l_1l_2...l_Lc_1c_2...c_mr_1r_2...$, with $|C| = l_1l_2...l_L$ being the length of the secret message with the value m and r_i being the appended bits that are selected randomly. The senders and the receivers should agree on the value of L beforehand, such that the receivers can fully recover C in the retrieving process.

After the preprocessing stage, the binary string C' is converted into some innocuous sentences by utilizing the dictionary table D and the sentence templates database S. First, the dictionary table is partitioned into 4 small tables according to the parts of speech of the words and all the words of each small table are mapped into binary codes using Huffman coding. The previously mentioned occurrences of words are used to assign variable-length Huffman codes to different words. Short Huffman codes are assigned to words with higher occurrences and longer ones to those with lower occurrences; this results in the frequently-occurring words having to be used more often in the generated sentences. Then, a sentence template is selected randomly. According to current bits of the binary string C, each part-of-speech of the sentence template is replaced with the proper word in the corresponding small tables, thus a generated sentence is obtained.

In the writing stage, we firstly compare the number of sentences produced in the generating stage with the number of slides of the PPT documents. Then, the sentences are written into the note pages of the PPT document evenly.

The details of the hiding process are presented in the algorithm below:

ALGORITHM 1: HIDING ALGORITHM

Input: A PPT document T and a message to be hidden C'.
Output: A stego-document T'.

Steps:

(1) Preprocess the body text of the PPT document T to extract a dictionary table D and a sentence template database $S = \{s_1, s_2..., s_n\}$, where s_s is a sentence template.
(2) Partition the dictionary table D into 4 small tables D_1, D_2, D_3, D_4 according to the set of parts of speech $P = \{n, v, a, d\}$ and construct a Huffman tree H_x for D_x, = 1, 2, 3, as follows.

(a) Create a leaf node n_i for each word in D_x and assign the occurrences of each word o_i to the node n_i.
(b) Initialize a set Q to contain all of the leaf nodes.
(c) Find in Q node n_i and n_j with the lowest occurrences and then remove node n_i and n_j from Q.
(d) Create a new node n_{ij} with the occurrences $o_{ij} = o_i+o_j$ and assign n_i as its left child and n_j as its right child.
(e) If Q is empty, then tree H_x has been constructed and take n_{ij} as its root; else, add node n_{ij} to Q and go to Step 2c).
(3) Randomly select a sentence template $s_s \in S$.
(4) For each $t_t \in S$, substitute word w_w for part-of-speech t_t to generate a new sentence s_s, where word w_w is determined as follows:
(a) Starting from the root of tree H_x, traverse H_x to its left child if the current bit of C' is 0 or to its right child.
(b) Go to the next bit of C' and continue traversing in a similar way until w_w is reached.
(5) Repeat steps 3) through 4) until the end of C'_{L+m}.
(6) Write the generated sentences s_1' s_2' into the note pages of T to yield a stego-document T'.

Retrieving process: In the retrieving process, we firstly create a dictionary table according to Rule 1. And then, we utilize the dictionary table to decode the corresponding binary bits of the words parsed from the note pages. The sentence templates have nothing to do with the retrieving process. The details are described in the following algorithm.

ALGORITHM 2: RETRIEVING ALGORITHM

Input: the stego-document T'.
Output: the message C'.

Steps:

(1) Preprocess the body text of stego-document T' to create a dictionary table D.
(2) Partition the dictionary table D into 4 small tables D_1, D_2, D_3, D_4 according to the set of parts of speech P = {n, v, a, d} and construct a Huffman tree H_x for D_x using step 2) described in Algorithm 1.
(3) Extract all the note pages from the stego-document T'.
(4) Parse sentences of the note pages by using the morphological analyzer to obtain a sequence of words with part-of-speech w_1/t_1 w_2/t_2....
(5) If $t_t \in P$, decode the corresponding bits of word w_w in the following way:
(a) Starting from the root of the Huffman tree H_x, traverse it to the leaf node w_w and record the path traversed.
(b) Analyze the path traversed and set the current bit of C' to 0 if the path goes down a left child; or to 1. Go to the next bit of C' for each child traversed.
(6) Repeat step 5) until C'_{L+m} has been retrieved.

EXPERIMENTS AND RESULTS

MIMIC-PPT is applicable to any language that has a morphological analyzer or part-of-speech tagger, e.g. English, Chinese and Japanese. Different from English texts, Chinese texts are explicit concatenations of characters and words are not delimited by spaces. Thus, it is more difficult and challengeable to implement MIMIC-PPT for Chinese texts. According to the algorithms, we utilize stanford log-linear part-of-speech Tagger (Toutanova *et al*., 2003) to implement an English MIMIC-PPT system and Chinese morphological analyzer IRLAS (Zhang *et al*., 2005) to implement a Chinese MIMIC-PPT system. In both systems, we assume that the note pages of PPT documents have no texts. If there are some sentences in the note pages, we delete the existing sentences and write the generated sentences. For the ease of description, we firstly take the English PPT document Practical Writing at the URL http://sfl.xjtu.edu.cn/center/writing/up/1147021618.ppt for example.

Firstly, the body text is extracted from the PPT document and tagged by the stanford log-linear part-of-speech Tagger (Toutanova *et al*., 2003) to obtain a sequence of words with parts of speech. Then, we pick up all the content words and record the occurrences of each word. And then we assign each word a Huffman code according to the occurrences to obtain a dictionary table. Due to the limit of space, Table 1 shows the occurrences and the resulting Huffman codes for the small table of adverbs. According to punctuations, we segment the body text sentence by sentence. Each sentence is to replace all the content words with the corresponding parts-of-speech to obtain a sentence template. Some

Table 1: A dictionary table of adverbs

Part-of-speech	Word	Occurrences	Huffman code
Adverb	Never	2	000
Adverb	Carefully	2	001
Adverb	Only	1	0100
Adverb	Verbally	1	0101
Adverb	Also	2	011
Adverb	Not	5	10
Adverb	Far	1	11000
Adverb	Too	1	11001
Adverb	There	1	11010
Adverb	Again	1	11100
Adverb	Really	1	11100
Adverb	So	1	11101
Adverb	Precisely	1	11110
Adverb	Already	1	11111

Table 2: Some sentence templates

No.	Sentence template
1	<n> or <n> <n>
2	<v> <a> that your <n> <v> <a>, <a> and <a>
3	<v> of <v> on <n>
4	<v> the <n> by <v> and <v> the <v> <n>
5	<n> of <n> <v>
6	What <v> the <n> about?
7	<n>the<n><d>,<v> to <v> out <d> what the <n> <v> about
8	How <a> <n> <v> <d> in the <n> and what <v> their <n>?
9	To <v> what <v>, <v> <n> in a <a> <n>
10	<v> two or <a> <n> to <v> a <a> <n>

Table 3: Some generated sentences by MICMIC-PPT system

No.	Generated sentence
1	Idea or events step-by-step
2	Tell straightforward that your charts writing sure, loyal and much
3	Be of figure on scene
4	Tell the order by are and tell the following information
5	Series of events writing
6	What stretch the order about?
7	Practical the object also, reading to writing out carefully what the scene help about
8	How useful step-by-step writing never in the expository-composition and what is their order?
9	To writing what be, is Pie in a faithful observe
10	Illustrated two or coherent perspective to used a orderly details

selected sentence templates are shown in Table 2, where <n>, <v>, <a> and <d> represent parts of speech of a noun, a verb, an adjective and an adverb respectively. We take the abstract of this research as a secret message to be encrypted and designate the length of message L = 16. Table 3 shows some sentences generated by the English MICMIC-PPT system. Finally, these sentences are evenly written into the note pages of the PPT document.

Compared with the existing systems of linguistic steganography, as mentioned before, MICMIC-PPT system can generate texts more efficiently and securely (Table 4). In order to evaluate the efficiency, we take the same message (the abstract of this study) as input to generate texts by using the existing systems and the Chinese MIMIC-PPT system. Thereinto, we take the first PPT document on the following website (http://www.pku.edu.cn/cernet2004/pptlist.htm) as an example. The numbers of words and bytes of the generated texts are showed in Table 4, where words of the Chinese MIMIC-PPT system are Chinese characters. Because of the inherent differences between English and Chinese, one byte (8 bits) represents an English letter, while two bytes (16 bits) represent a Chinese character. We also introduce the expansion rate to measure the efficiency, which is the ratio of the number of bytes of the generated text divided by the number of bytes of the secret message. The results indicate that the expansion rate of the Chinese MIMIC-PPT system is lower than other systems. This is achieved for the reason that we pick all the content words, which are most frequently used and we utilize Huffman coding to avoid discarding any content word.

Table 4: Comparison of several systems

The systems	Words	Bytes	Expansion rate	Lexical items	Syntactically correct	Semantically coherent
Mimicry applet	36076	151682	232.64	Valid	Yes	No
Stego	3716	15720	24.11	Valid	No	No
Texto	2150	9010	13.82	Valid	Yes	No
Nicetext	9721	41995	64.41	Valid	Yes	No
Play maker	4829	21095	32.35	Valid	Yes	No
Chinese MIMIC-PPT	2243	4592	7.04	Valid	Yes	No

To demonstrate the qualities of the texts produced by these systems, three levels of linguistic correctness are conducted, namely lexical level, syntactic level and semantic level. Utilizing some existing resources of lexical and syntactic analysis, it is observed that all the generated texts contain valid lexical items and they are syntactically correct texts, except for the text produced by Stego. It is because Stego is only dictionary-based, while not complying with any sentence template. Due to limits of current automatic semantic analysis, we manually evaluate semantic coherence. Every individual sentence of the generated texts makes sense. However, it should be noted that the sequences of sentences of all the generated texts do not have coherent contexts. Some results of several systems are also shown in Table 4.

SECURITY ANALYSIS

Different from existing linguistic steganography methods, to transmit a generated text along with a PPT document is more reasonable and secure on MIMIC-PPT system. It is normal to send and receive a meaningful PPT document via the Internet. And a note page is an essential part of each slide in PPT documents, which provides accessorial description for the presentation. Through parsing the generated text of the Chinese or English MIMIC-PPT systems, all the words are the content words used in the body text of PPT documents and most words are high-frequency words. Additionally, the sentence templates are also the styles of the body text of PPT documents. Therefore, the notes will simulate the content and the writing style of the body text so that it can provide the opportunity of deniability. Deniability is derived from the fact that even if an adversary finds the notes suspicious, the sender may claim that the notes are the real explication of the representation.

Due to the random choosing of the sentence templates derived from existing sentences in the body text, the sequence of sentences generated by the MIMIC-PPT system does not add up to an comprehensible text, as showed in Table 4. However, the sequence of sentences will be later written into the note pages evenly, so the

necessity for semantically coherence between sentences should not be taken as an absolute requirement of the MIMIC-PPT system. Each sentence produced by the MIMIC-PPT system is derived from the sentence templates of the body text, thus it is possible to draw attention from a human reader. In addition, first encrypting the secret message before the generating stage can ensure that an adversary cannot obtain the real secret message even if he or she knows our algorithm.

To strengthen the security of the MIMIC-PPT system, the generated text should be as imperceptible as possible to adversaries. This can be achieved by the following ways. The first one is to dynamically select a key-dependent subset of the dictionary table. The other one is to utilize natural language generation techniques for the purpose of creating sophisticated sentence templates. Moreover, not all the sentence templates derived from the body text are appropriate for the generating stage and thus it is necessary to choose some sentence templates to obtain a sentence template database according to some selection rules. In addition, a PPT document can be extended to the form of a Microsoft PowerPoint Show document (PPS for short), in which the notes are fully invisible for the readers.

CONCLUSION

This study presents a new method of text steganography for Microsoft Power Point documents, MIMIC-PPT. The body text is mimicked to generate some sentences to be written into the note pages. Thus, the method can provide deniability and be applied to other document formats that offer notes. Additionally, the generating stage is not done by creating large and sophisticated type dictionaries beforehand; rather, the dictionary and sentence templates are automatically generated by parsing the body text of PPT documents. Therefore, the communication parties need not share dictionaries and sentence templates. As a result of Huffman coding in each small dictionary table, the generated text can not only mimic the statistical properties of the body text, but also enhance the efficiency.

Because most expressions of the body text in PPT documents are concise, the sentence templates extracted from them are simpler than general sample texts. In addition, the content of each note page is not closely related to the content of the slide when the generated sentences are written into the note page evenly. We intend to investigate some natural language generation techniques in the near future for improving the sentence templates and adaptively writing the generated sentences into the note pages.

ACKNOWLEDGMENTS

This research is supported by Key Program of National Natural Science Foundation of China (No. 60736016), National Basic Research Program of China (No. 2006CB303000), and National Natural Science Foundation of China (No. 60573045).

REFERENCES

Bergmair, R., 2007. A comprehensive bibliography of linguistic steganography. Proceedings of 9th Conference on Security, Steganography and Watermarking of Multimedia Contents, San Jose, CA.

Chapman, M.T. and G.I. Davida, 1997a. Nicetext. Website, August 1997. http://www.nicetext.com/.

Chapman M.T. and G.I. Davida, 1997b. Hiding the hidden: A software system for concealing ciphertext as innocuous text. Proceedings of the 1st International Conference on Information and Communications Security.

Chapman, M.T., G.I. Davida and M. Rennhard, 2001. A practical and effective approach to large-scale automated linguistic steganography. Proceedings of the 4th International Conference on Information Security, LNCS., 2200: 156-165.

Chapman, M.T. and G.I. Davida, 2002. Plausible deniability using automated linguistic steganography. Proceedings of International Conference Infrastructure Security, LNCS., 2437: 276-287.

Chiang, Y.L., L.P. Chang, W.T. Hsieh and W.C. Chen, 2004. Natural language watermarking using semantic substitution for Chinese text. Proceedings IWDW 2003, LNCS., 2939: 129-140.

EI-Kwae, E.A. and L. Cheng, 2002. HIT: A new approach for hiding multimedia information in text. Proceedings of the SPIE 4675: 132-140, San Jose, CA.

Gmbh, S., 2000. Sams Big Play Maker. Website, http://www.scramdisk.clara.net/play/playmaker.html.

Maher, K., 1995. Texto. Circulating on the web, February 1995. http://www.ecn.org/crypto/soft/texto.zip.

Murphy, B. and C. Vogel, 2007. The syntax of concealment: Reliable methods for plain text information hiding. Proceedings of 9th Conference on Security, Steganography and Watermarking of Multimedia Contents, San Jose, CA.

Petitcolas, F.A.P., R.J. Anderson and M.G. Kuhn, 1999. Information hiding-a survey. Proceedings of the IEEE, 87(7): 1062-1078.

Topkara, M., C. Taskiran and E.J. Delp, 2004. Natural Language Watermarking. In: Proceedings of the SPIE, Delp Edward, J. and W. Wong Ping (Eds.). San Jose CA, 5681: 441-452.

Topkara, M., G. Riccardi, D.H. Tur and M.J. Atallah, 2006. Natural language watermarking: Challenges in building a practical system. Proceedings of International Conference on Security, Steganography and Watermarking of Multimedia Contents, San Jose, CA, pp: 106-117.

Toutanova, K., D. Klein, C.D. Manning and Y. Singer, 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. Proceedings of HLT-NAACL, Edmonton, pp: 173-180.

Walker, J., 1994. Steganosaurus. Circulating on the web, December 1994. http://www.fourmilab.ch/stego/stego.shar.gz.

Wayner, P., 1992. Mimic functions. Cryptologia, 16 (3): 193-214.

Wayner, P., 1995. Strong theoretical steganography. Cryptologia, 19 (3): 285-299.

Wayner, P., 1997. Mimicry Applet. Website, http://www.wayner.org/texts/mimic.

Wayner, P., 1999. Mimic Functions and Tractability. Technical Report, Cornell University, Department of Computer Science.

Wayner, P., 2002. Disappearing Cryptography: Information Hiding: Steganography and Watermarking. 2nd Edn. San Mateo, CA: Morgan-Kaufmann, pp: 81-128.

Zhang, H.P., T. Liu, J.S. Ma and X.T. Liao, 2005. Chinese word segmentation with multiple postprocessors in HIT-IRLab. Proceedings of SIGHAN, pp: 172-175.