Coverless Text Steganography Based on Maximum Variable Bit Embedding Rules

Ning Wu¹, Poli Shang², Jin Fan³, Zhongliang Yang⁴, Weibo Ma¹, Zhenru Liu¹

¹School of Electrical Engineering, Lanzhou Institute of Technology, Lanzhou 730050, China
²School of Electronic and Electrical Engineering, Lanzhou Petrochemical Polytechnic, Lanzhou 730060, China
³School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China
⁴Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
wun0016@163.com

Abstract. As a carrier of information hiding, text is difficult to hide secret because of its less redundancy. The robustness and security of some traditional methods based text are not high. Based on Markov chain model, a coverless text steganography algorithm was proposed in this paper. According to the characteristic and value of transition probability in the model, the algorithm used the maximum variable bit embedding instead of the usual fixed bit embedding. In order to verify the effectiveness of the algorithm, different datasets and algorithms were selected to test. By comparing with results, the proposed model had higher concealment and hidden capacity.

1. Introduction
The technology of hiding secret information into general media and transmitting it through public channel is called steganography. The text steganography method based on carrier automatic generation, also called coverless text steganography, could hide information by generating encrypted text which accords with statistical characteristics of natural language. It could better avoid steganalysis based on statistical features.

Based on Markov chain model, a coverless text steganography algorithm based on maximum variable bit embedding rule was proposed in this paper. This algorithm generated the corresponding steganographic text from the secret information through the model. This algorithm did not embed fixed bits of secret information into every word of text, but retained the characteristics of the training model to the greater extent, and combined with the transition probability, aimed at generating steganographic text closer to the training text.

2. Markov chain model and steganography application

2.1. Markov chain model
Markov chain is a stochastic process in probability theory and mathematical statistics, which has Markov property and exists in discrete exponential set and state space. It describes a sequence of states,
each of which depends on the preceding finite state. At each step of Markov chain, according to the probability distribution, the system may change from one state to another state or keep the current state. If the conditional probability distribution of $X_{n+1}$ for past states is only a function of $X_n$, then there is an equation (1)

$$P(X_{n+1}=x|X_1=x_1, X_2=x_2, \ldots, X_n=x_n) = P(X_{n+1}=x|X_n=x_n)$$

(1)

where $x$ is a state in the process. The above equation could be regarded as a Markov property.

2.2. Application of steganography
In natural language processing, Markov model is often used as a good approximation of natural language statistical model. This model is therefore often used in steganography. [1] proposed a method of hiding and extracting information based on Markov chain and state transition diagram, as shown in Fig.1.

![State transition diagram and steganography](image)

Steganography process. The initial state is given as $S_0$. If the secret information that needs to be embedded is 0 11, the encrypted text $S_0$ $S_1$ $S_6$ will be generated according to the state transition diagram.

Extraction process. This process is an inverse process of steganography. If the encrypted text is $S_0$ $S_2$ $S_8$, compared with the state transition diagram, the original information is 1 1.

In the state transition diagram, the probability of transferring from the current state to the next specific state is called the transition probability. In literature [2-3], in order to simplify the process of natural sentence generation and related analysis, some methods assumed that all transition probabilities from a given state to any other state are equal. The others although recognized the importance of transition probabilities, the ones were not well utilized.

3. Coverless text steganography based on variable bit embedding rules
In order to better deal with steganalysis based on statistical features of natural language and to generate high quality natural sentences relying on the sample library to the greater extent, a new steganography model was proposed by combining transition probability with variable bit embedding rule on the basis of Markov chain model.

3.1. Selection of statements
The state transition diagram obtained from the sample library is shown in Fig.2.
In Fig.2, the transition probability based on statistics indicated the degree of connection between words in the sample text. According to statistics, the transition probability of text $S_0$ to text $S_1$, $S_2$ and $S_3$ was 0.4, 0.5 and 0.1, respectively, and so on. In previous algorithms, binary coding was usually matched to the state transition diagram after diagram established. These coding numbers were usually matched state transition diagram by fixed bit number. For the example above, this method may cause some problems. If embed 2 bits, the information expression for the successive branches of $S_0$ is sufficient. However, for $S_2$, half of the post-order branching text would be deleted from the model, which could not better reflect the original characteristics of the sample library in probability.

3.2. Binary information matching and steganography

In order to generate steganographic text as close as possible to the features of training samples, the method preserved the features of original text as much as possible in matching binary coding with the state transition diagram. In order to preserve better sentences in the sample text, the transition probability was sorted and the branches were adjusted. The maximum number of bits that could be embedded in successive branches of text fragment $i$, that is, $EB_{\text{max}}$:

$$EB_{\text{max}} = \lfloor \log_2 n \rfloor$$

where $n$ is the original number of successive branches of the text fragment $i$.

Reserved branch number is

$$RB_i = 2^{\lfloor \log_2 n \rfloor}$$

Correspondingly, the number of branches deleted is

$$DB_i = n - RB_i = n - 2^{\lfloor \log_2 n \rfloor}$$

Fig.2 Hypothetical state transition diagram

Fig.3 Adjustment and matching of binary information
According to the above rules, the state transition diagram was adjusted. Fig. 3 describes a new state transition diagram, which takes Fig. 2 as the initial state transition diagram and is adjusted to match the binary information. Steganography was accomplished according to the diagram.

The steganography process could be described as follows:

1. According to the selected sample library, all sentences with the same first word are selected.
2. According to the results of the first step, a Markov state transition diagram is established.
3. Sort the transition probabilities with the same node from large to small. If the number of successive branches of the node is \( n \), the former \( RB_i = 2^{\log_2 n} \) branches will be retained, and the remaining branches will be deleted.
4. Match the state transition branches with explicit binary coding.
5. Input secret information into the model to generate steganographic text.

3.3. Steganographic information extraction

In order to restore secret text correctly, the same state transition diagram and same rules should be used. When the receiver obtained the steganographic information, the correct original transmission information was extracted by comparing with the state transition diagram.

The extraction process could be described as follows:

1. According to the pre-agreed sample library, all sentences with the same first word are selected.
2. According to the pre-agreed rules and the result of the first step, a Markov state transition diagram is established.
3. According to the pre-agreed rules, the probabilities with the same node are sorted from large to small, and the number of successive branches of each node is \( n \), then the pre-\( RB_i = 2^{\log_2 n} \) branches are retained, and the remaining branches are deleted.
4. According to the pre-agreed rules, the state transition branches are matched with explicit binary coding.
5. Comparing steganographic text with model to restore original information.

4. Experiments and analysis

In order to test the performance of the algorithm, two kinds of datasets on the network were chosen, one was about movie reviews [4], and the other was about news [5], as shown in Table.1.

| Dataset    | Film review | News |
|------------|-------------|------|
| Average Length | 19.94 | 22.24 |
| Sentence Number | 1,283,813 | 1,962,040 |
| Words Number  | 25,601,794 | 43,626,829 |

Since the primary function of our algorithm was to hide secret information through new generated text, the indicators in natural language could be used to measure the steganography effect. Perplexity was chosen as an indicator. In natural language processing, the degree of perplexity could be used to measure the quality of language generation model. The larger the value, the worse the language model. The perplexity is expressed by the following formula:

\[
Perplexity = 2^{-\frac{1}{m} \sum_{i=1}^{m} \log p(s_i)} \tag{5}
\]

where \( s_i = \{w_1, w_2, w_3, ..., w_m\} \) is the generated sentence, \( p(s_i) \) indicates the probability distribution over words in sentence \( s_i \) and the probability is calculated from the language model of the training texts. \( m \) is the total number of generated sentences.
From Fig.4, the perplexity of our model was smaller compared with the other two similar models. This showed that the steganographic text generated by our model was closer to the training sample. In the second evaluation index, the embedding rate was chosen. It was used to calculate how much information could be embedded in text, which was an important index to evaluate the performance of steganography algorithm. The mathematical expression is as follows:

$$ER = \frac{1}{N} \sum_{i=1}^{N} \frac{(L_i-1)}{B(s_i)}$$

where $N$ is the number of generated sentences and $L_i$ is the length of $i$-th sentence. $B(s_i)$ indicates the number of bits occupied by the $i$-th sentence in the computer. Two text steganography methods are compared with the model proposed in this paper, and the following results are obtained.

From Fig.5, it shows that our model has more steganography capacity than other models.

5. Conclusion
In this paper, a maximum variable bit steganography method based on Markov was used to realize the information hiding. Previous similar algorithms mostly used fixed bit embedding method for steganography. For the model with complex structure, a large number of text relations will be lost, which will affect the quality of steganographic text generation. In the research, the characteristics were preserved to the greater extent according to the model structure obtained from the samples. Through relevant experiments, the effectiveness of the algorithm was verified. In the future research, the algorithm will be further discussed on this basis, with a view to further improving the algorithm from different perspectives.

Acknowledgments
This work was supported by the National Key Research and Development Program of China (2018YFB1003205) and Gansu Natural Science Foundation (1506RJZA057).

References
[1] 3S. F. Wu, Researches on Information Hiding Technology[D]. Master Thesis. Hefei: China Science and Technology University, 2003(in Chinese).
[2] Dai W, Yu Y, Dai Y, et al. Text Steganography System Using Markov Chain Source Model and DES Algorithm[J]. JSW, 2010, 5(7): 785-792.

[3] Moraldo H H. An Approach for text steganography based on Markov Chains[J]. arXiv preprint arXiv:1409.0915, 2014.

[4] Maas A L, Daly R E, Pham P T, et al. Learning word vectors for sentiment analysis[C]//Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1. Association for Computational Linguistics, 2011: 142-150.

[5] Andrew Thomson. News dataset. DOI= https://www.kaggle.com/snapcrack/all-the-news/data

[6] Stutsman R, Grothoff C, Atallah M, et al. Lost in just the translation[C]//Proceedings of the 2006 ACM symposium on Applied computing. ACM, 2006: 338-345.

[7] Chen X, Sun H, Tobe Y, et al. Coverless information hiding method based on the chinese mathematical expression[C]//International Conference on Cloud Computing and Security. Springer, Cham, 2015: 13-143.