Journal of Engineering Science and Technology
Special Issue on ISSC'2016, April (2017) 169 - 180

THE PERFORMANCE EVALUATION OF FEATURE-BASED TECHNIQUE IN TEXT STEGANOGRAPHY

SUNARIYA UTAMA, ROSHIDI DIN*, MASSUDI MAHMUDDIN

School of Computing, UUM College of Arts and Sciences,
Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia
*Corresponding Author: roshidi@uum.edu.my

Abstract

The technique of steganography in text domain is widely employed because of any agreement, report and any other important documents are in form of text, whereas other mediums usually use entertainment environment. One of steganography categories in the medium of text is called text steganography. It conceals hidden message with embedding the text via various methods used could be horizontally shifting, vertically shifting, or based on the letter of text structures. Many researchers highly utilize the feature-based technique to cover hidden message based on uniqueness of letter. This paper concerned about analysed the feature-based text steganography method by using evaluation performance in form of verification and validation through the implementation of feature-based technique in text steganography.

Keywords: Steganography, Text steganography, Feature-based, Verification, and validation.

1. Introduction

The shape of a projectile is generally selected on the basis of combined Steganography is definite as the art and science of hiding the hidden message so that the existence of a message is not detected by human vision. Secure the private information is critical point of steganography in applying performance as a part of information hiding. The practical of idea steganography has been used since the ancient times [1]. It is one of sub-disciplines of information hiding [2] that dependable with some medium as location for hiding the hidden message. The implementation of steganography itself is divided the methods into two categories.

The first is the implementation of steganography in medium of text is named as natural language steganography. The second is implementation steganography in different medium of image, audio, video and other digitally invisible code could be called technical steganography. It is anticipated the steganography approach is able to give some solution for safeguarding the security of information in text media [3].

The implementation of natural language steganography is hiding the hidden message in medium of text that the third party is unable to discover the existences of message in text. In other words, steganography in medium of text can make the secret information invisible and unnoticed for third party to see or detect, and it is directed to the appropriate receivers to apprehend the information.

However, steganography in text domain is the most challenging method implementation than others domain. It has been known based on past research that development steganography in text domain is the most challenging method because of text file has small quantity to hide information [4] and it is dependable on limitation space in text. The analogy of steganography in domain of text is showed in Fig. 1 as follows.

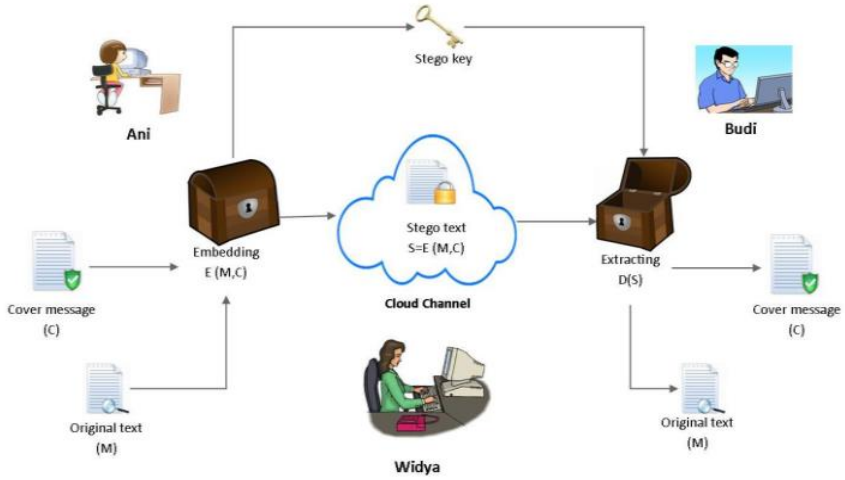

Fig. 1. The analogy process natural language steganography.

In Fig. 1 illustrated the process steganography in text domain analogically can be illustrated using Prisoner's Problem [5]. It signified, Ani is distributing an original text (*M*) along with a cover message (*C*) in order to process embedding known as stego text (*S)* covering a stego key (*K*). Primarily, apply the invertible function *e: {M, C}* $\rightarrow$ *S*. Ani able to plan an original text (*M*) using a stego key (*K*) through *e* (*M, C*) = *S*. Therefore, *S* as stego object and it is invertible function, Widya will not able discover it suspicious thing. Next, Budi will figure out *e - 1(S) = {M, C}* in order to retrieve original text *M* and cover message *C* with a stego key *K* for decoding the process use function.

Natural language steganography conceals message in medium of text. It can be divided into two main categories; text steganography and linguistic steganography. Text steganography covers messages which manipulating the component in text such as, word, space, line and any other characters in sentence of text [6-7]. Whereas, linguistic steganography is covering messages by modifying the

information that encoded based on linguistic order [8]. In Fig. 2 shows the category of natural language steganography techniques as follows.

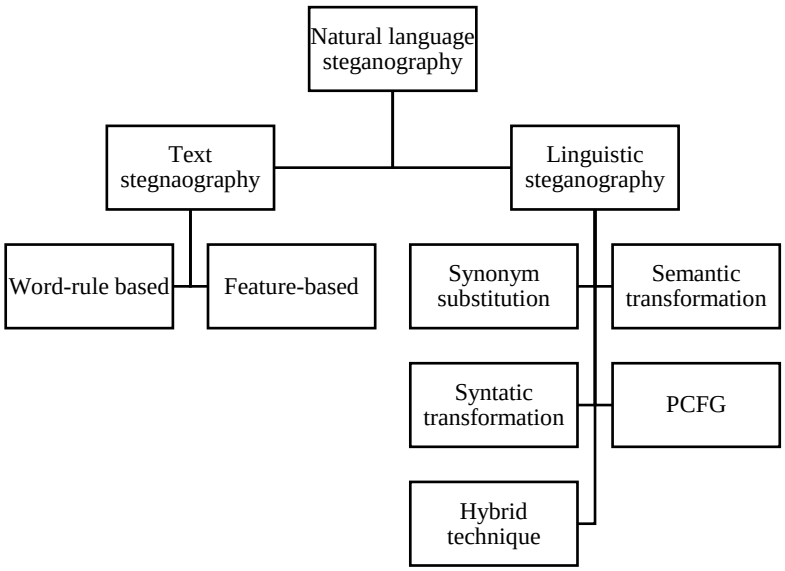

Fig. 2. The category of natural language steganography.

Based on Fig. 2 presented the natural language generally consist with categories; Text steganography and Linguistic Steganography. For text Steganography hide the messages which manipulating the attributes in text such as, word, space, line and any other characteristic in sentence of text [9]. Meanwhile, linguistic steganography hides the message based on adjusting the information that arranged based on linguistic order [8] that consist of the several methods such as synonym substitution, semantic transformation, syntactic transformation, PCFG and hybrid technique.

The main category of this paper is clarifying about technique in text steganography that consist with two techniques such as word-rule based and feature-based methods. The focus in this paper is about feature-based methods that covering the hidden messages feature of letter by manipulating in the shape, size, and position of font in the text. It chooses the three techniques in feature-based which are Change Alphabet Language Pattern (CALP) [10] Curve Subheading (CURVE) and vertical straight line based (VERT) [11].

2. Research Design

In this study involved several stages in order to develop the research design that shown in Fig. 3 as follow.

According to Fig. 3, the three general stages for developing the process in order to achieve the objective the study. Those are the stage of research design such as establish input stage, process analyzed stage, and output stage.

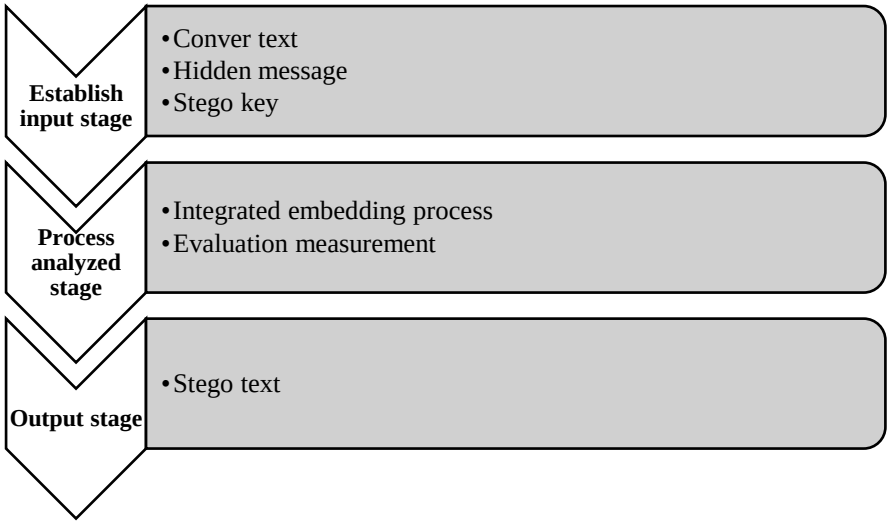

Fig. 3. The phase model of research design.

2.1. Establish input stage

In this stage, the necessary requirement has prepared as the establish input such as cover text, hidden message and stego key.

The cover text used in this study, it used as original text as the medium that would be embedded in hidden message in process to hide information in technique text steganography. Then, the hidden message as the information which would be chosen to be embedded in cover text based of each technique in text steganography in the form of binary bits. Meanwhile, the stego key is the three techniques chosen which are CALP, CURVE and VERT as schemes that encipher the encrypted the embedding algorithm and deciphers the extract message technique [12]. Therefore, with three requirements as the input data could be developed to continue develop process analyzed stage.

2.2. Process analyzed stage

In this stage, process analyzed is the main focus in the experimental design which consists of several algorithms based on stego key as the technique of feature-based. This stage as the formula the technique that will chose from one of several stego key algorithm. It is to convert the hidden message into binary bits and embed it into cover text to be the stego text. This process will receive input texts establish from establish stage, and determine the algorithm used based on what kind of stego key has been chosen. The first process in this stage with Integration embedding algorithm uses the three of technique stego key that can be chosen by users. Each stego key has the algorithm for embedding the hidden message in process analyzed environment. The second stage of analysed process on this stage is evaluation measurement as the main objective in this paper. There are two processes in this phases that will be considered for verification process and

validation process. Both of processes are the principal process in assessing accuracy result in computational condition.

2.3. Output environment

In this stage, the output environment obtains the data through input design through to get the similar data with input data. This environment is generated by process of verification and validation is stego output environment. The stego text output is the final output which is obtained from hiding input environment is embedded. And undergone verification and validation process with feature-based technique based on the stego key. Based on this stage will discover the comparison of three technique evaluation through verification and validation.

3. Performance Evaluation

The performance evaluation that analyzed in this study consists of 15 hidden messages that used in cover text on each technique of feature-based in text steganography. The capacity size bit of hidden message showed in Table 1 as follows.

Table 1. Capacity size bit of hidden message.

No.	Capacity size bit (kb)
1.	0.126
2.	0.202
3.	0.280
4.	0.356
5.	0.430
6.	0.504
7.	0.576
8.	0.650
9.	0.726
10.	0.804
11.	0.882
12.	0.962
13.	1.038
14.	1.114
15.	1.188

Based on the Table 1, the hidden message with that capacity of size bit that could embeds into cover text in order to generate stego text.

3.1 Verification performance

The verification process is determining the input variables, when the system due process or stop, and the output [13]. The result of the verification in some system discovered based on parameter metric as tools in order measure the process system. In this paper, the parameter metric uses the verification evaluation is size bit of stego text. This parameter measured the capacity of the text after embedded with hidden message.

3.1.1 Size bit of stego text CALP technique

In Fig. 4 presents the size bit of stego text used CALP technique as follows.

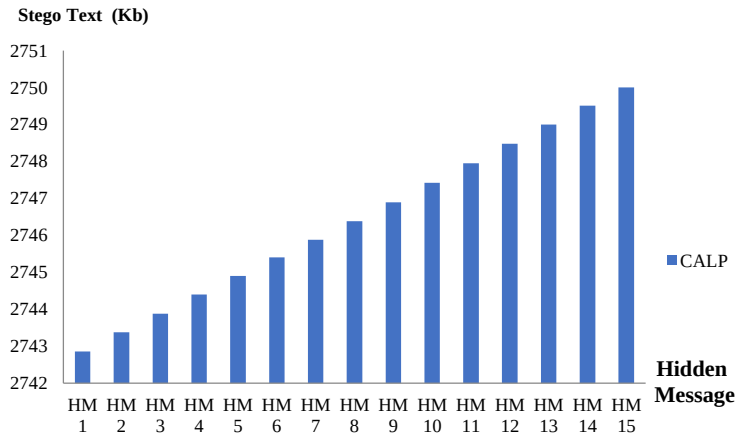

Fig. 4. The size bit of stego text using CALP technique.

According to Fig. 4, the capacity of size bit of text after embedded with hidden message using CALP technique. The capacity of size bit of text start with almost 2743 kb in Hidden message 1 until reach 2750 in hidden message 15.

3.1.2 Size bit of stego text CURVE technique

The size bit of stego text used CURVE technique is showed in Fig. 5 as follows.

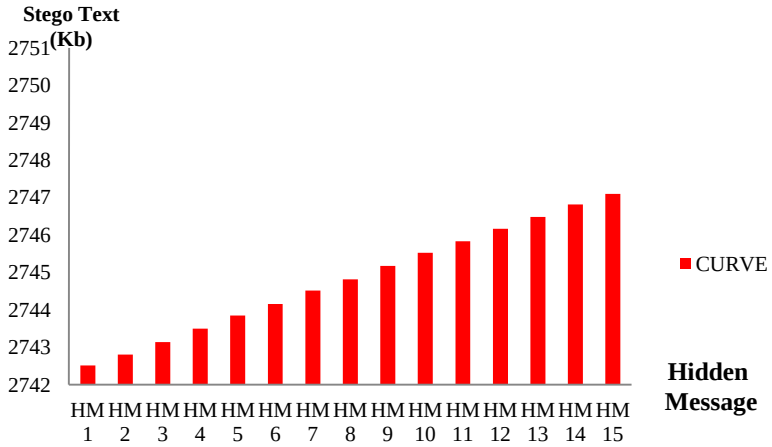

Fig. 5. The size bit of stego text using CURVE technique.

In Fig. 5 showed the capacity of size bit of text after embedded with hidden

message using CURVE technique. The capacity of size bit of text start with around 2742.5 kb in Hidden message 1 until reach 2747 with last hidden message embedded.

3.1.3 Size bit of stego text VERT Technique

Fig. 6 presents the size bit of stego text used VERT technique as follows.

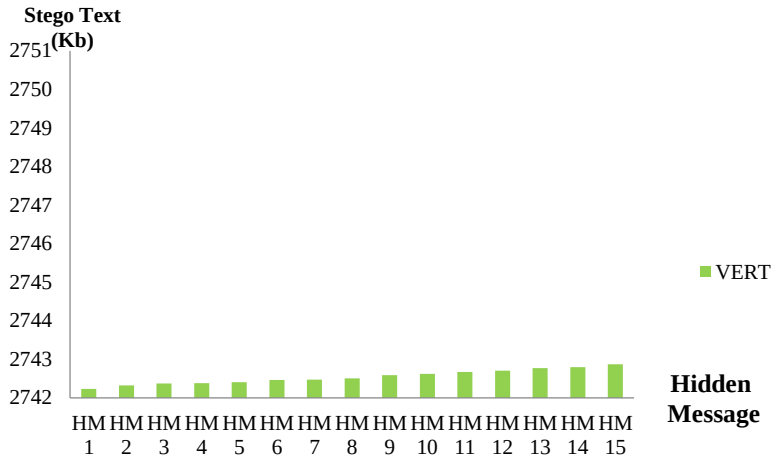

Fig. 6. The size bit of stego text using VERT technique.

Different with other techniques, VERT technique increases the capacity size bit of stego text embedded is slightly insignificants. It clearly seems for the first until the end of embedded hidden message which not even reach 2743 kb.

3.2. Validation performance

The Validation process is generating the expected output from testing process that can prevent problem in systems or applications [14-15]. The validation also has the parameter metric in order to obtain the validated value of result in the system. One of parameter metric in validation is running time [16-18]. However, the parameter metric running time in text steganography in this paper named embedding time process.

3.2.1. Embedding time process in CALP technique

The embedding time process of CALP technique as validation tools measurement presents in Fig. 7 as follows**.**

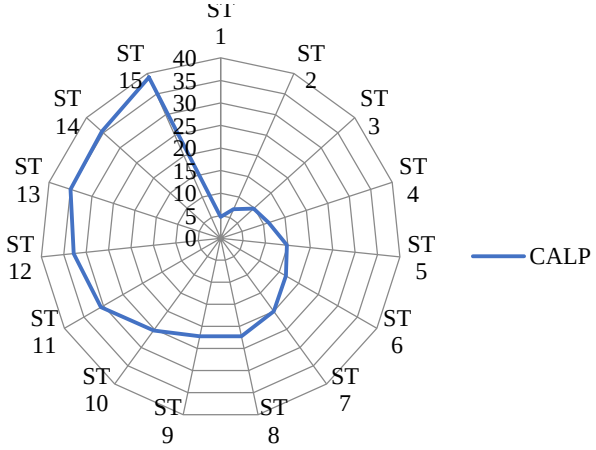

Fig. 7. Embedding time process in CALP technique.

The embedding process in CALP technique showed in Fig. 7 that described higher capacity of size bit would need more time in order to embedding process. Where the generates stego text 1 need 3 second in embedding time process and the last stego text need around 23 seconds in embedding time process.

3.2.2. Embedding time process in CURVE Technique

In Fig. 8, embedding time process of CURVE technique is showed as follows.

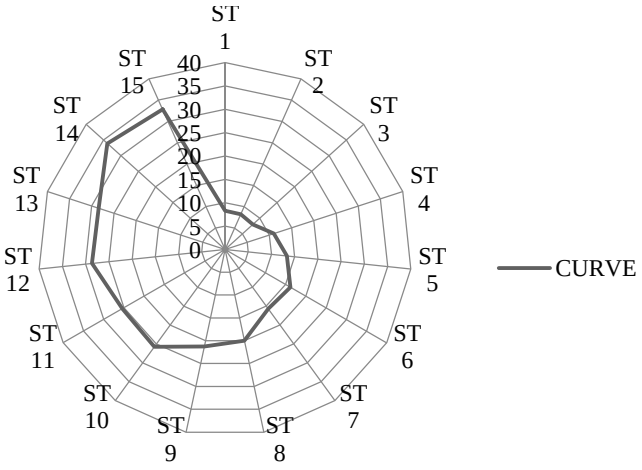

Fig. 8. Embedding time process in CURVE technique.

Based on Fig. 8 showed the CURVE technique also need several times in order to embed the hidden message with highest capacity size bit. However, some of higher size bit hidden message faster than the lower size bit. It showed in stego text 11 that faster that reach 18 seconds than stego text 10 and 9 that need more time in embedding process.

3.2.3. Embedding time process in VERT technique

The embedding time process of VERT techniques shows in Fig. 9 as follows.

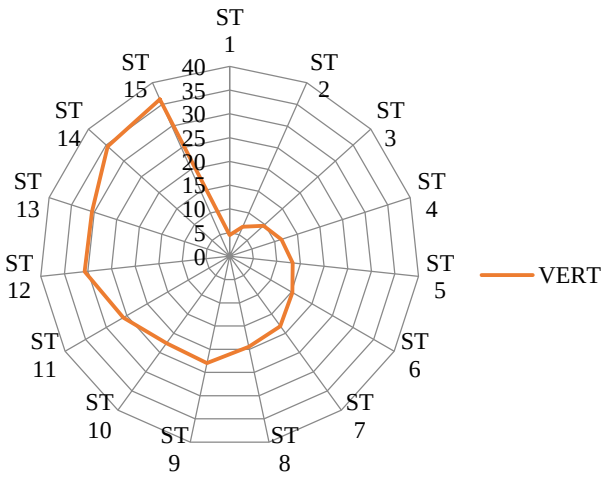

Fig. 9. Embedding time process in VERT technique.

In Fig. 9 showed the embedding time process did not quite stable. It showed in stego text 8 embedding time process is slower than the stego text 9 and 10 that have higher size bit. However, the last stego text that has highest size bit reach 35 as the longest time in embedding process.

4. Discussion

Based on performance evaluation that obtained the comparison oh three techniques of feature-based. The comparison of size bit stego text as the verification evaluation as a parameter in feature-based text steganography showed in Fig. 10 as follows.

In Fig. 10 shown the CALP technique has a higher value from stego text that embedded with hidden message 1 until in stego text15 that reach around 2,750 kb which significantly stable in increased the value of size bit. It followed by CURVE technique that has a lower value size bit value than CALP that only reach around 2,747 kb. The lowest value in Fig. 10 is also VERT technique which the highest size bit of stego text of hidden message that reach around 2,472 kb that even lower than stego text in CALP technique.

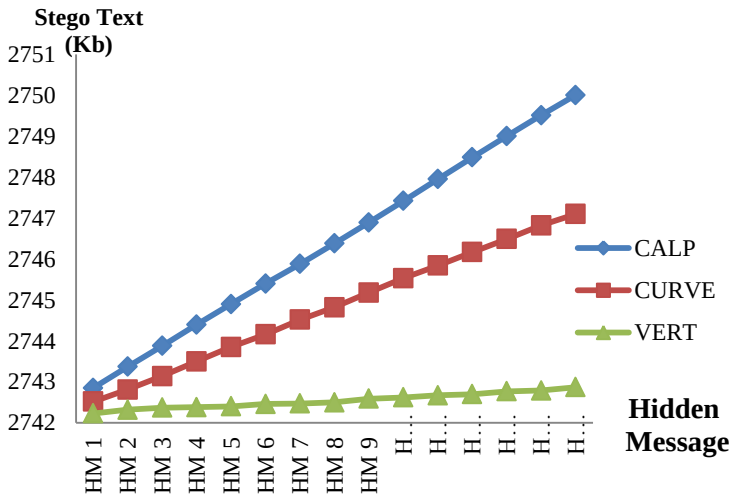

Fig. 10. The comparison of stego text size bits.

Next, the comparison of embedding time text as the validation evaluation as the parameter in feature-based text steganography showed in Fig. 11 as follows.

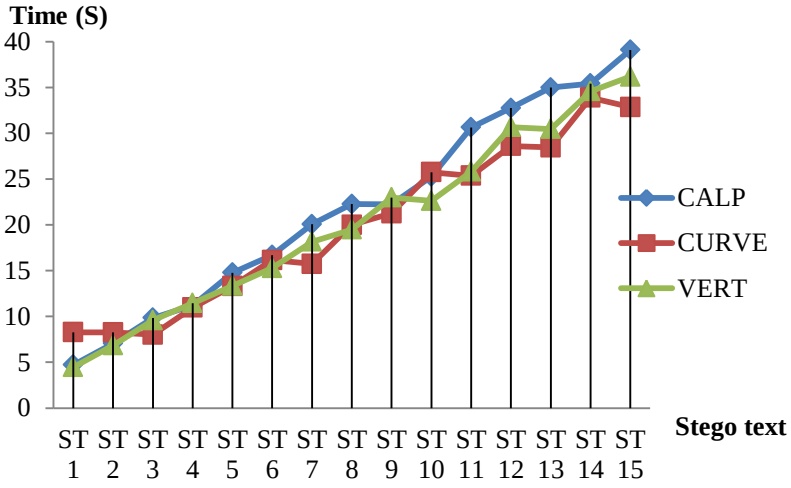

Fig. 11. The comparison of embedding time process.

Based on Fig. 11 illustrates the embedding time of stego text in CALP technique is the fastest than other technique and quite stable from first until the last stego text. It only needs less than 25 seconds embedding time in last stego text. This followed by CURVE that reaches around 30 seconds and VERT technique which needs around 35 seconds in embedding the last stego text.

5. Conclusions

This paper evaluated the three techniques of feature-based in text steganography

such as CALP, CURVE and VERT technique through the verification and validation measurement. The verification parameter in this study used is size bit stego text that measured the capacity size bit of CALP is highest than other techniques, while the VERT technique has lowest capacity size bit than other techniques. Then, in validation measurement used embedding time process as tools which obtained even the CALP has highest capacity size bit but have fastest time of embedding and stable. Meanwhile, CURVE technique is slower embedding time than CALP and rather unstable. Furthermore, VERT technique is slowest embedding time even though has lowest size bit and unstable process embedding time compare the other technique.

Acknowledgement

We would like thank to Director of Awang Had Salleh Graduate School (AHSGS) for their moral support for the realization of this work. This research was financially supported by the Non-PI Grant (No: 13437/2016) under RIMC Grants, Universiti Utara Malaysia (UUM).

References

1. Iyer, S.S.; and Laktaria, K. (2016). New robust and secure alphabet pairing text steganography algorithm. *International Journal of Current Trends in Engineering & Research*, 2(7), 15–21.
2. Por, L.Y.; and Delina, B. (2008). Information hiding: A new approach in text steganography. *7th WSEAS International Conference on Applied Computer & Applied Computational Science*. Hangzhou, China, 689-695.
3. Odeh, A; Khaled, E.; and Feazipour. (2013). Text steganography using language remark. *ASEE Northeast Section Conference*. 1-7.
4. Nasab, M.V.; and Shafiei, B.M. (2011). Steganography in programming. *Australian Journal of Basic and Applied Sciences*, 5(12), 1496-1499.
5. Simmons, G.J. 1984. Prisoners problem and the subliminal channel. *Proc.CRYPTO83 - Advances in Cryptology*, 1(1), 51-67.
6. Liu, M.; Guo, Y.; and Zhou, L. (2009). Text steganography based on online chat. *5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 807-811.
7. Rabah, K. (2004). Steganography-The art of hiding data. *Information Technology Journal*, 3(3), 245-269.
8. Chang, C.Y.; and Clark, S. (2010). Linguistic steganography using automatically generated paraphrases. *2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 591–599.
9. Lee, C.F.; and Chen, H.L. (2013). Lossless text steganography in compression coding. Recent Advances in Information Hiding and Applications. *Springer Berlin Heidelberg*, 155-179.
10. Bhattacharya, S.; Indu, P.; Duta, Biswas, S.A.; and Sanyal, G. (2011). Hiding data in text through in alphabet letter patterns (CALP). *Journal of Global Research in Computer Science*, 2(3), 33-39.

11. Dulera, S.; Jinwala, D.; and Dasgupta, A. (2011). Experimenting with the novel approaches in text steganography. *International Journal of Network Security & Its Applications*, 3(6), 213-225.
12. Agarwal, M. (2013). Text steganography approaches: a comparison. *International Journal of Network Security & Its Applications*, 5(1), 91-106.
13. Das, H.; Jarfour, A.; Orlitsky, A.; Pan, S.; and Suresh, A.T. (2012). On the query computation and verification of functions. *IEEE International Symposium on Information Theory Proceedings*, 2711-2715.
14. Ling, J.; Chen, J.; and Liu, C. (2008). An automatic mechanism for adjusting validation function. *22nd International Conference on Advanced Information Networking and Applications*, 602-607.
15. Marincic, J.; Mader, A.; and Wieringa, R. (2011). Validation of embedded system verification models. *IEEE Journal Systems*, 1(1) 48-54.
16. Dasso A.; and Funes, A. (2007). *Verification, validation, and testing in software engineering.* Idea Group Publishing, London.
17. Catal, C. (2012). Performance evaluation metrics for software fault prediction studies. *Acta Polytechnica Hungarica*, 9(4), 193-206.
18. Oberkampf, W.L.; and Roy, C.J. (2010). *Verification and validation in scientific computing*. New York: Cambridge University Press.