Journal of King Saud University – Computer and Information Sciences 31 (2019) 348–366

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

# A novel steganography method using transliteration of Bengali text

Md Khairullah

*Department of Computer Science and Engineering, Shahjalal University of Science and Technology, Sylhet 3114, Bangladesh*

ARTICLE INFO

*Article history:*
Received 30 October 2017
Revised 22 January 2018
Accepted 23 January 2018
Available online 17 February 2018

*Keywords:*
Steganography
Transliteration
Phonetic keyboard
Capacity
Steganalysis
SVM

ABSTRACT

In this paper, we present a simple and novel approach for steganography through transliteration. A phonetic keyboard layout is very popular for writing languages having non-roman alphabets. Bengali, a language spoken by 230 million people, is a fair example and in this work we utilize Bengali digital text for data hiding by the proposed technique. For several characters in the Bengali alphabet, there are multiple options to represent a character in it's equivalent roman form using a phonetic keyboard layout. The main idea of the proposed method is to exploit this special feature of Bengali phonetic keyboard layouts to hide secret information in form of bits. One of these options can be used to represent the bit '0' and the other option can represent the bit '1' in a document without any risk of understanding by any intermediate user. The results show that the proposed method is very prominent to be a successful steganography technique. Steganalysis results show that the capacity of the method is 1.2%, which is adequate for a text steganography system with very low risk of machine detection. This method can be easily adapted and applied for any other language having non-roman alphabet.

## 1. Introduction

Steganography can be defined as a method of hiding secret data within a cover media so that other individuals fail to realize the existence of the secret data. In other words, steganography is the science of hiding information. It is often confused with cryptography as both are used to protect confidential information. The difference between the two is in the appearance of the processed output; the output of steganography operation is not obviously visible but in cryptography the output is twisted so that it can draw attention. If a nefarious government or Internet service provider (ISP) is looking for encrypted messages, they can easily find them. Whereas the goal of cryptography is to make data unreadable by a third party, the goal of steganography is to hide the data from a third party. Steganalysis is the process to detect the presence of steganography. The key terminologies in the context of steganography are: *plaintext* is the original secret message that needs to be communicated; *covertext* is the larger and harmless looking data which is used as container for the plaintext; and the *stego text* is the data generated after embedding the plaintext into the covertext.

Steganography is the art of hiding information in seemingly harmless carriers without drawing suspicion to the transmission of a hidden message. On the other hand, the art of discovering and rendering such harmless covert messages of hidden information is known as steganalysis. The primary goal of steganalysis is the identification of the existence of a hidden message, and then the identification of hot-spots to look for hidden information (Johnson et al., 2001).

The basic features expected from a steganography method are high embedding capacity, invisibility or perceptual transparency, undetectability, robustness (i.e. the ability of the algorithm to retain the data embedded in the cover), tamper resistance (the capability to prevent modification or deletion or embedding a different message), and the independence of the original cover (Salomon, 2003). Off course, some of these requirements conflict and thus, any specific algorithm can satisfy only one or two of them. More specifically, embedding capacity, robustness, and undetectability are mutually conflicting and cannot all be achieved by one algorithm.

Image, audio and video are some popular media for steganography. On the other hand, text is ideal for steganography due to its ubiquity and smaller size compared to these media. However, text communication channels do not necessarily provide sufficient redundancy for covert communication. Bengali is a native language to Bangladesh, the Indian state of West Bengal, and parts of the Indian states of Tripura and Assam. It is written with the Bengali script. With nearly 230 million total speakers, Bengali is ranked

*E-mail address:* khairullah-cse@sust.edu
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2018.01.008
1319-1578/© 2018 The Author. Production and hosting by Elsevier B.V. on behalf of King Saud University.

as the sixth spoken language in number of speakers in the world. Location of Bengali in Unicode is 0980 09FF.

A keyboard layout is any specific mechanical, visual, or functional arrangement of the keys, legends, or key-meaning associations (respectively) of a computer, typewriter, or other typographic keyboard. A phonetic keyboard layout, or phonetic keyboard for short, is a setup where the letters of one language correspond to the keys in the keyboard layout for another language. It assumes one-to-one correspondence between letters in the languages, often based on their sound. This process is generally called transliteration and the converted text is called the transliterated text. The transliteration should be reversible, i.e. able to automatic and unambiguously recreate the original. As the languages of the world are dominated by roman alphabets, the most commonly used transliteration is to roman.

Like other non-roman languages, Bengali input methods can be grouped into fixed layouts and phonetic layouts. Beside many online Bengali phonetic keyboard tools, there are a number of very desktop applications such as Avro, Onkur, Ekushey Phonetic, Bengali Transliteration by Google, Bengali Transliteration by Microsoft, and Akkhor. Due to the recent surge in social networks through smart mobile phones among the mass, some mobile applications such as Ridmik and Mayabi achieved huge popularity. Table 1 shows the list of Bengali alphabets, and the corresponding roman form(s) used by Avro phonetic keyboard (OmicronLab, 2017). Additionally, a statistics of the percent usage (Sattar et al., 2004) of the Bengali alphabet is included.

From Table 1, we immediately observe that some Bengali characters can be represented in two ways. One of the option can be used to represent the binary bit 0 and the alternative option can represent bit 1. For instance, ক can be represented as ‘k’ or ‘q’ representing bit 0 and 1 respectively. Alternatively, ‘k’ or ‘q’ can represent bit 1 and 0 respectively. However, if we consider case insensitivity of the roman representation of a number of Bengali characters, Table 1 can be extended to include a lot of possible options for other characters too. This expansion is presented in Table 2.

After the introduction of Unicode as well as some phonetic Bengali typewriting software, Bengali computing has raised to a peak. It is reflected in the number of Bengali web pages in the Internet. Similarly more Bengali documents are used in offices and organizations than ever. This vast amount of Bengali documents may be easily used as a covert media for the proposed steganography system. Note that beside the standard Bengali text used in formal documents and newspapers, transliterated/romanized Bengali text is very popular in personal and informal communications using instant messengers and social networks (Hassan et al., 2016; Khan, 2014).

Beside these, text steganography with a regional language have certain other benefits. To suspect hidden information a person must be a native or specialized user of this language and requires specialized knowledge of Bengali Unicode. So, the proposed technique is a safer method of secret data communication over an international network like the Internet.

It is important to mention that the whole encoding and decoding process is done by a software system at the two communication ends. So there is no scope of introduction of any kind of bit errors due to the steganography process except those bit errors introduced by the underlying communication system and the hardware. However, it is possible for an unintended user to eavesdrop and hack to change randomly some of the special characters of the stego text containing the secret bits and corrupt the hidden secret information. Integration of a simple and conventional error detection method such as the checksum can ensure the correctness of the decoded secret data with high probability. Generally, this issue is more related to network security, compared to steganography and out of the current scope of discussion.

One additional and interesting point is that roman representation of the Bengali text takes less disk space due to the fact that the size of the ASCII roman characters is 1 byte, where the Bengali Unicode characters require 1–4 bytes, as specified in the UTF-8 standard for Unicode. Thus, using probability theory, the compression ration is $(4+3+2+1)/4 = 2.5$ and using 200 Bengali text of varying size in the range 8 kilo bytes to 100 kilo bytes we found the average reduction is 2.41. For example, the size of stego text in Appendix is 3.4 kilo byte while the source Bengali text size is 8 kilo byte. Simplifying this we can say that at least half of the required memory or disk space can be saved by using transliteration of Bengali text instead of the original Unicode based Bengali text. Taking this into account, we assume that the size of the resultant stego text is half of the original Bengali text in the later discussion.

Support Vector Machine (SVM) is a supervised learning method primarily used for discriminative classification (Cortes and Vapnik, 1995). It learns from the given labeled training data and outputs an optimal hyperplane which categorizes new examples. However, evaluation and optimization of an SVM model is a challenging problem. Classification Accuracy, Precision and Probability of Detection, Confusion Matrix, etc are used for SVM evaluation, which are greatly affected by imbalanced samples distribution or misclassification cost (Tian et al., 2011). A receiver operating characteristic (ROC) curve is an alternative classification performance metric that overcomes the above issues. In a two-dimensional graph, it plots the number of true positives on the y-axis against the number of false positives on the x-axis, over the range of possible threshold values. The area under the curve (AUC), a quantitative representation, is equal to the probability that a classifier will rank a randomly chosen “true” instance higher than a randomly chosen “false” one (Fawcett, 2006). Accuracy, the proportion of the total number of predictions that were correct to the total number of data points, is an alternative and effective performance metric for a SVM binary classifier.

## 2. Related work

Text steganography can be broadly classified into three types: format-based, random and statistical generations, and linguistic method. Format-based methods use and change the formatting of the cover-text to hide data. They do not change any word or sentence, so it does not harm the ‘value’ of the cover text. Random and statistical generation methods are used to generate cover text automatically according to the statistical properties of language. These methods use example grammars to produce cover text in a certain natural language. The linguistic method considers the linguistic properties of the text to modify it. The method uses linguistic structure of the message as a place to hide information. Following is a list of some major works that have been carried out on hiding information or text steganography. Appropriate available steganalysis methods are included as well.

### *2.1. Using specific characters in a text*

This is an analytical, complicated and time consuming method. Some specific characters from certain words are selected (Morkel et al., 2005). For example, the first words of each paragraph are selected in a manner that by placing the last characters of the selected words side by side forms the secret information. A detection method based on the distribution of first letters of words is discussed in Sui et al. (2006).

**Table 1**
List of Bengali characters and the roman equivalent form and alternatives along with the usage statistics.

| Bengali | Percentage | Roman |
|---|---|---|
| অ | 2.39 | o |
| আ | 12.97 | a |
| ই | 5.22 | i |
| ঈ | 1.19 | I or ee |
| উ | 1.26 | u or oo |
| ঊ | 0.43 | U |
| ঋ | 0.00 | rri |
| এ | 7.81 | e |
| ঐ | 0.05 | OI |
| ও | 0.80 | O |
| ঔ | 0.08 | OU |
| ক | 4.34 | k or q |
| খ | 0.68 | kh or qh |
| গ | 0.81 | g |
| ঘ | 0.11 | gh |
| ঙ | 0.01 | Ng |
| চ | 0.41 | c |
| ছ | 0.86 | ch |
| জ | 1.02 | j |
| ঝ | 0.04 | jh |
| ঞ | 0.09 | NG |
| ট | 0.57 | T |
| ঠ | 0.13 | Th |
| ড | 0.14 | D |
| ঢ | 0.03 | Dh |
| ণ | 0.43 | N |
| ত | 4.36 | t |
| থ | 0.52 | th |
| দ | 2.36 | d |
| ধ | 0.60 | dh |
| ন | 5.05 | n |
| প | 2.91 | p |
| ফ | 0.28 | ph or f |
| ব | 3.81 | b |
| ভ | 0.44 | bh or v |
| ম | 3.19 | m |
| য | 1.18 | z |
| র | 6.63 | r |
| ল | 3.09 | l |
| শ | 0.91 | sh or S |
| ষ | 0.55 | Sh |

### 2.2. Shifting methods

In printed document the lines of the text are vertically shifted to some degrees to hide secret data (Low et al., 1995; Alattar and Alattar, 2004). In a similar approach, information is hidden in the text by shifting words horizontally and by changing distance between words (Low et al., 1995; Kim et al., 2003). This method is acceptable for texts where the distance between words is varying. This method can be identified less, because change of distance between words to fill a line is quite common. If the text is retyped or if character recognition programs (OCR) are used, the hidden information would get destroyed. Steganalysis for word-shifting method is discussed in Li et al. (2008).

### 2.3. Exploiting punctuation signs

Appropriate placement of some punctuation signs such as full stop (.) and comma (,) can hide information in a text file (Morkel et al., 2005). This method requires identifying proper places for putting punctuation signs.

### 2.4. Using alternative words

By using the synonym of words for certain words one can hide information in the text (Alattar and Alattar, 2004; Niimi et al., 2003). A major advantage of this method is the protection of information in case of retyping or using OCR programs. However, this method may alter the meaning of the text sometimes. Similarly, by using abbreviations some information can be hidden, though with very less capacity (Morkel et al., 2005). For example, only a few bits can be hidden in a file of several kilobytes. The statistical method (Yu et al., 2009) using context information can detect this type of steganography with a very high success rate.

### 2.5. Feature coding method

Some of the features of the text can be altered to hide some information (Rabah, 2004). For example, the end part of some characters such as h, d, b or so on, is elongated or shortened a little thereby hiding information in the text. Retyping the text or using OCR program destroys the hidden information. Additionally, data can be hidden by displacing letter points and diacritics. Arabic, Persian and Urdu texts are used for this technique (Shirali-Shahreza and Shirali-Shahreza, 2006; Memon et al., 2005; Aabed et al., 2007; Gutub et al., 2008). One of the characteristics of these languages is plenty of points in its letter. One point letters are used to hide the information by shifting position of point a little bit vertically high with respect to the standard point position in the text. The same technique is applied for the vowel signs to hide information.

### 2.6. White space strategy

We can add some extra white spaces in the text (Morkel et al., 2005; Huang and Yan, 2001) to hide some information. These white spaces can be placed at the end of each line, at the end of each paragraph or between the words. However, some text editor programs automatically delete extra white spaces and thus destroy the hidden information. In Rose et al. (2014) a text 'visualization' method can easily detect the stego text produced by this technique.

**Table 2**
Extended list of Bengali characters with the equivalent roman representations.

| Bengali | Roman | Alternative Roman |
|---|---|---|
| অ | o | |
| আ | a | A |
| ই | i | |
| ঈ | I | ee, eE, Ee, EE |
| উ | u | oo |
| ঊ | U | |
| ঋ | rri | |
| এ | e | E |
| ঐ | OI | |
| ও | O | |
| ঔ | OU | |
| ক | k | K, q, Q |
| খ | kh | Kh, kH, KH, qh, qH, Qh, QH |
| গ | g | G |
| ঘ | gh | gH, Gh, GH |
| ঙ | Ng | |
| চ | c | C |
| ছ | ch | cH, Ch, CH |
| জ | j | J |
| ঝ | jh | jH, Jh, JH |
| ঞ | NG | |
| ট | T | |
| ঠ | Th | TH |
| ড | D | |
| ঢ | Dh | DH |
| ণ | N | |
| ত | t | |
| থ | th | tH |
| দ | d | |
| ধ | dh | dH |
| ন | n | |
| প | p | P |
| ফ | ph | f, pH, Ph, PH, F |
| ব | b | B |
| ভ | bh | v, bH, Bh, BH, V |
| ম | m | M |
| য | z | Z |
| র | r | |
| ল | l | L |
| শ | sh | S, sH |
| ষ | Sh | SH |
| স | s | |
| হ | h | H |
| ড় | R | |
| ঢ় | Rh | RH |
| য় | y | Y |
| ৎ | t“ | |
| ◌ং | ng | |
| ◌ঃ | : | |
| ◌ঁ | ^ | |

### 2.7. By extension letter

Text steganography is applied on Arabic text Gutub and Fattani (2007) for this algorithm. Arabic language has a special extension character, which can be arbitrarily inserted between characters for formatting purposes.

### 2.8. Through office suite documents

In the proposed technique in Khairullah (2009) three bytes of secret information can be hidden as RGB color values of each invisible character such as the space, the tab and the new line characters in an MS Word document. A novel technique presented in Mahato et al. (2017) utilizes the *track changes* feature of Microsoft Word for hiding the secret message. In Kumar et al. (2016) secret data bits are hidden in Microsoft Word document by changing the font type and font style of white space characters without raising suspicion by the user. Various techniques of steganography in MS Excel documents and relative benefits have been discussed in Bin et al. (2011), Tiwari and Sahoo (2011). For example, the text direction in a cell can slightly be rotated based on the intended bits to hide.

### *2.9. Utilizing place value in decimal numbers*

In this method, data is hidden in a cricket match scorecard by adding a meaningless zero before a number to represent bit 1 and leaving the number as it is, to represent bit 0 (Khairullah, 2011). Financial documents, e.g. a balance sheet, can be also exploited in a similar way to utilize for steganography (Khairullah, 2014).

### *2.10. Through computing technologies*

Some markup language feature can be used to hide information (Bennett, 2004). For instance, case insensitivity of HTML tags can be exploited. For example, the tag <BR> can be also used as <Br> and <br>. As a result one can do text steganography in HTML documents by changing the small or large case of letters in document tags. In some cases the positions of tags are also used. For example <B><U> </B></U> may represent bit 0 and the alternative <U><B> </U></B> can represent bit 1. Information can be extracted by comparing the tags positions. A corresponding detection method is presented in Huang et al. (2009). A CSS (Cascading style sheet) based steganography is discussed in Kabetta and Dwiandiyanta (2011). This technique encrypts a message using RSA public key cryptosystem and cipher text is then embedded in a Cascading Style Sheet (CSS) by using End of Line on each CSS style properties, exactly after a semicolon. A space after a semicolon embeds bit 0 and a tab after a semicolon embeds bit 1. The proposed method in Bassil (2012) encodes input text messages into SQL carriers made up of SELECT queries. In effect, the output SQL carrier is dynamically generated out of the input message using a dictionary of words implemented as a hash table and organized into 65 categories, each of which represents a particular character in the language. Under some assumptions from the perspective of information theory and practice, a text steganography framework based on searching a webpage has been proposed in Shi et al. (2016).

### *2.11. Compression algorithm methods*

In Satir and Isik (2012) the LZW compression algorithm is used to hide secret information. This method hides the secret data into the email addresses which are listed in the 'Cc' field. For each character in the secret message a relative 'distance' of the same character in a text is calculated, thus a 'distance vector' is derived for the secret message and a 'distance matrix' is generated for each of the text of a text-base. The optimal text which gives the highest repeatability of the distance values is ultimately chosen from the text-base as a cover text as well as the stego key. The LZW code is calculated for this optimal distance matrix and resulting bits are concatenated and divided into blocks of 12 bits with partition of 9 and 3 bits. These partitions are used to select the user-name and the domain-name from some available options to form a valid email address. Although the algorithm is complicated due to large number of components, the capacity is up to 7%. Similar methods based on Huffman compression is presented in Satir and Isik (2014), Malik et al. (2017), Rahman et al. (2017). A simplified extension is presented in Malik et al. (2017) that directly applies the LZW algorithm on the secret message and the obtained bit stream is hidden into email addresses and also in the message of the email, that rises the capacity up to 14%. Colors are used to hide secret bits in the email text following some color coding, which is a primary disadvantage.

### *2.12. Using mathematical tools*

The well variety of the list even includes some mathematical theorems and formulas to hide information in text. A PDF file based steganography is presented in Ekodeck and Ndoundam (2016), which applies the Chinese Remainder Theorem to generate the stego text. The fact that the non-breaking space character (i.e. ASCII code A0) becomes invisible to common PDF readers is utilized to hide secret data at between-word or between-character locations in a PDF file. The disadvantage of the increase in the resulting PDF file size is mitigated through the Chinese Remainder Theorem, which adds some randomness as well. The fundamental idea of the theorem is to reduce modular calculations with large moduli to similar calculations for each of the (mutually co-prime) factors of the modulus. This concept is imported in the method to reduce number of A0s to be inserted. The authors in Mandal et al. (2014) propose a new kind of number system for text steganography through a directed weighted graph cover media. Each character in the secret data is converted to a 2D coordinate value and is presented as a node at that coordinate in planar graph. The character sequence in the secret message is guaranteed by using smaller edge weights between the character pairs. Recently, a method based on Markov chains of different orders is proposed (Shniperov and Nikitina, 2016). The basic idea of the method is generation of stego text on the basis of a Markov chain, preconstructed using a text pattern which is composed in a natural language. The generated stego text normally reflects a common meaning, at the same time each of its sentences will quite reliably repeat syntactically and grammatically some blocks of the text pattern. The claimed capacity is up to 9%. A new key-based model of text steganography is proposed in Acharjee et al. (2016) that uses the XOR operation on the start and end letter of words of the cover. Based on this result and the bit to be hidden a key is stored in a key file which along with the cover file is sent to the receiver.

### *2.13. Using machine generated Chinese text*

A novel text steganography method is proposed in Luo and Huang (2017) which uses Recurrent Neural Networks (RNN) Encoder-Decoder structure to generate a certain genre of Chinese poetry. The proposed method ensures high capacity as well as better poetry quality. A similar work is presented in Liu et al. (2016) which generates song poetry that is utilized in secret data hiding. The basic idea is to generate a large amount of redundancy which makes enough space for hiding a large amount of data, for example the capacity of the two methods is 35% and 27% respectively. The proposed method in Qi and Guo (2015) generates a text carrier from a massive targeted corpus through natural language processing technology. Additionally, a repository of the right words and the wrong words is built and the secret message is embedded through substitution of error words for candidate correct words after word segmentation of the text carrier. The pairs of error word and correct word is located using the Chinese text automatic proofread technology to extract the embedded secret message in the receiver.

## 3. Steganography in Bengali

Most of the techniques described above are applicable for Bengali text also along with their inherent advantages and disadvantages. Methods in Sections 2.1,2.2,2.3,2.4,2.5,2.6,2.7,2.8,2.9 can be easily implemented for steganography in Bengali with a little modification or no modification at all.

A feature coding based approach is implemented to hide secret message in the text by shifting the specific 'matra' towards left, by shifting the points of some characters and by shifting the character 'ref' respectively (Changder and Debnath, 2009). A new linguistic approach for steganography through Indian languages such as Bengali by considering the flexible grammar structure of the languages is presented in Changder et al. (2010). In this work, the bits of the binary stream are encoded to some part-of-speech and create meaningful sentences starting with a suitable word belonging to the mapped part-of-speech. A quantum approach based text steganography technique has been proposed with the help of Bengali language (Banerjee et al., 2012). This approach uses two specific characters and two special characters like inverted commas (opening and closing) in Bengali language and the mapping technique of quantum gate truth table. Another text steganography technique has been proposed considering the structure of Bengali alphabet, which hides secret message through changing the pattern of Bengali alphabet letters (Bhattacharyya et al., 2011). Considering the availability of more characters and flexible grammar structure of Indian Languages including Bengali, another steganography method hides the secret message in the text by creating meaningful sentences after finding the longest common subsequence of two binary string among which one is the secret message and another may be any binary string (Changder et al., 2010).

The proposed method in Khairullah (2018) utilizes the composite form of certain Bengali characters in Unicode to hide bits of a secret code. These characters defined in Unicode also have composite forms, which means those can be written using two different codes in Unicode. In other words, these characters have a single form as well as a composite form. For example, the Bengali diacritic 'ো' can be written using either Unicode 09CB or as the combination of Unicode 09C7 and 09BE, i.e.'ে' + 'া'. One of these two forms can be used to represent the bit 0 and the other form can represent the bit 1. A similar method is presented in Xinmei et al. (2010) that hides secret data in Chinese text.

As with other languages Bengali language is rich with large set of synonyms. Flower can be represented by at least 4 Bengali words: ফুল, পুষ্প, কুসুম, প্রসূন and sun can be translated to the distinct words সূর্য, রবি, সূর, অরূণ, অঁজিষ্ণু, অর্ক, ভানু, ভাস্কর, অংশুমালী. So a synonym based text steganography system in Bengali has high capacity. Additionally, the language has a large number of synonyms due to the inheritance from Sanskrit. The words চন্দ্র (originally Sanskrit) and চাঁদ (in pure Bengali) represent the word moon. Another source of availability of large number of synonyms are the Shadhu form and the Cholito form of the verbs in Bengali, though mixing of both the forms in the same document or speech is highly discouraged. For example, খাইয়াছি and খেয়েছি are the two forms to mean the finished work of eating in the first person.

Bengali speaking people use abbreviated forms of some commonly used word combinations in their daily lives and particularly in office environments. কাবিখা (কাজের বিনিময়ে খাদ্য), মূসক (মূল্য সংযোজন কর), শাবি (শাহজালাল বিশ্ববিদ্যালয়), গসাগু (গরিষ্ঠ সাধারণ গুণনীয়ক) are a few examples of common abbreviations in Bengali. This indicates that the abbreviation based Bengali text steganography will be also strong.

By displacing letter points in the characters ড়, ঢ়, য়, and র; and the diacritics ু, ূ, and ৃ (for example in the conjuncts কু, কূ, and কৃ)) we can implement a powerful Bengali text steganography system.

## 4. The proposed technique

Based on the discussion in Section 1, a novel text steganography method that utilizes the transliteration of Bengali text can be implemented in following two ways.

### 4.1. Based on different roman representation

A very simple approach of hiding binary bits is to exploit the basic alternative roman forms in Table 1. A code-map table can be constructed to map the special Bengali characters to a desired roman representation. Table 3 is such an example. Note that the roman representations can be permuted to represent different bits. For instance, in the 3rd row of Table 3, the two options for ক can be swapped, e.g. ‘q’ for bit 0 and ‘k’ for bit 1. Hence, this table can be constructed in $2^7$ or 128 ways.

**Table 3**
List of Bengali characters which can be written with different roman character(s).

| Bengali | Roman (bit 0) | Alternative Roman (bit 1) |
|---|---|---|
| ঈ | I | ee |
| ঊ | u | oo |
| ক | k | q |
| খ | kh | qh |
| ফ | ph | f |
| ভ | bh | v |
| শ | sh | S |

For example, the Bengali text আজ যেমন ক'রে গাইছে আকাশ is represented as “aj zemon *q*‘ore gaiche aka*S*” to hide the 3-bit stream “101”. In the transliterated text, the roman character representing bit 1 is italicized and the character representing bit 0 is underlined for easy understanding of the reader.

The capacity of the proposed method is quite low. Based on the statistics of usage in Table 1, the characters in Table 3 is used around 9% in a Bengali text. So, we can assume that in every 4000 characters we are able to hide 360 bits. More precisely, 45 bytes of secret information can be hidden in a 8 Kbyte document. On an average, a standard and average Bengali news paper article’s size is around 8 KBytes and a password or a pin number or a short secret message requires less than the mentioned 45 bytes.

### 4.2. Based on case insensitivity

#### 4.2.1. Using Bengali text as input

Based on the case insensitivity, Table 2 can be exploited to dramatically extend the capability of data hiding method. An example is provided in Table 4, which can be constructed in $2^{37}$ ways. Notice that all available roman alternatives for a Bengali characters is not used as the usable number of options is power of 2.

For example, the Bengali text আজ যেমন ক'রে গাইছে আকাশ is represented as “*aJ* *z*e*m*on **q**‘ore *g*ai**Ch***e* a*kaS*” to hide the 18-bit stream “10 101 100 10100 00011”. For the convenience of the reader the roman characters in the transliterated roman text are formatted according to the following bit representations: bit 1 italic, bit 0 underline, bit pair 10 bold, and the bit pair 00 italic and underline together.

On contrary to the low capability of the method using different roman representation, the capacity of the method that utilizes case insensitivities is high. According to Table 1, the Bengali characters applicable for multiple roman representation (see Table 4) are used almost 63% in Bengali text. Thus, a 8 KB Bengali document having 4000 characters can hide 2520 bits (315 bytes). Notice that this capability is almost double of the standard SMS length limit, which is 160 bytes. This is clearly observed in the above example. The same amount of stego text now can hide 18 bits, which is only 3 bits for the method based on different roman representation.

#### 4.2.2. Using the roman representation as input

By a close observation of Table 4 we find that the characters “abcefhjklmpqvyz” from the roman alphabet are case insensitive for transliteration and they can be directly utilized in transliteration. The corresponding code-map table can be formed in $2^{15}$ or 32,768 ways. As the encoding and decoding step will be character

**Table 4**
List of Bengali characters which can be used to represent binary bits by using alternative forms in a phonetic keyboard layout. Bit combinations are in the parenthesis.

| Character | Option 1 | Option 2 | Option 3 | Option 4 | Option 5 | Option 6 | Option 7 | Option 8 | Capacity |
|---|---|---|---|---|---|---|---|---|---|
| আ | a (0) | A (1) | | | | | | | 1 |
| ঈ | I (00) | ee (01) | eE (10) | Ee (11) | | | | | 2 |
| উ | u (0) | oo (1) | | | | | | | 1 |
| এ | e (0) | E (1) | | | | | | | 1 |
| ক | k (00) | K (01) | q (10) | Q (11) | | | | | 2 |
| খ | kh (000) | Kh (001) | kH (010) | KH (011) | qh (100) | qH (101) | Qh (110) | QH (111) | 3 |
| গ | g (0) | G (1) | | | | | | | 1 |
| ঘ | gh (00) | gH (01) | Gh (10) | GH (11) | | | | | 2 |
| চ | c (0) | C (1) | | | | | | | 1 |
| ছ | ch (00) | cH (01) | Ch (10) | CH (11) | | | | | 2 |
| জ | j (0) | J (1) | | | | | | | 1 |
| ঝ | jh (00) | jH (01) | Jh (10) | JH (11) | | | | | 2 |
| ঠ | Th (0) | TH (1) | | | | | | | 1 |
| ঢ | Dh (0) | DH (1) | | | | | | | 1 |
| থ | th (0) | tH (1) | | | | | | | 1 |
| ধ | dh (0) | dH (1) | | | | | | | 1 |
| প | p (0) | P (1) | | | | | | | 1 |
| ফ | ph (00) | f (01) | pH (10) | Ph (11) | | | | | 2 |
| ব | b (0) | B (1) | | | | | | | 1 |
| ভ | bh (00) | v (01) | bH (10) | Bh (11) | | | | | 2 |
| ম | m (0) | M (1) | | | | | | | 1 |
| য | z (0) | Z (1) | | | | | | | 1 |
| ল | l (0) | L (1) | | | | | | | 1 |
| শ | sh (0) | S (1) | | | | | | | 1 |
| ষ | Sh (0) | SH (1) | | | | | | | 1 |
| হ | h (0) | H (1) | | | | | | | 1 |
| ঢ় | Rh (0) | RH (1) | | | | | | | 1 |
| য় | y (0) | Y (1) | | | | | | | 1 |

by character, this approach is very simple to implement and understand. This approach is particularly useful if the available input text is already transliterated.

For example, the equivalent transliterated text “aj zemon ko‘re gaiche akash” is converted to “*A*j *Z*e*M*on *K*‘ore gai*C*h*E* ak*AsH* to hide the 15-bit stream “10101100 1010000”. Here, the characters hiding bit 1 are italicized and the characters hiding bit 0 are underlined for the convenience of the reader. Notice that, now the each character hides maximum of one bit. For example, the roman characters ‘q’ and ‘k’ now hides one bit instead of two bits, and ‘g’ hides nothing, which is a contrary to the last method.

Table 6 summarizes the usage statistics of the roman characters to represent Bengali text. This statistics is derived from the usage statistics of the Bengali characters presented in Table 1. The usage percentage of each roman character is calculated by summing up the percent usage of the Bengali characters which utilize this. For example, the usage of ‘b’ is 4.25% as it is used to represent both ব and ভ, which are used 3.81% and 0.44% respectively in Bengali text according to Table 1.

According to Table 6, these roman characters are used almost 60% in transliteration of Bengali text. Thus, in 4000 chars 2400 bits (300 bytes) can be hidden in a 8 KB Bengali document. Again, this capability is almost double of the standard SMS length limit. Here, the same amount of stego text now can hide 15 bits, which is 18 bits for the previous method and only 3 bits for the method based on different roman representation.

As the two techniques utilizing the case insensitivity (see Section 4.2) have a significantly higher capacity, we consider these two methods in the proposed steganography technique. As the generated stego text by the two methods slightly differ, we propose to combine both methods in our steganography method and it dynamically chooses the method to use based on the version of the input text or by an agreement between the sender and the receiver or by a fixed protocol.

Let we have some secret information to hide in the document. We have to convert the secret information to its equivalent secret bit stream consisting of only 0 and 1. Let the selected method of steganography is by using the original Bengali text. We start by looking for an occurrence of the special characters included in the Table 4 (for example আ, Unicode U + 0986) in the document. As this has the capacity of hiding 1 bit, we take the first bit of the bit stream. If the bit is 1 we will use the option 1 of the corresponding roman character (for example ‘a’) in the output text and option 2 (for example ‘A’) otherwise. On the other hand, if the found special character has higher capacity (for example ক, Unicode U + 0995), we take multiple bits from the bit stream and use option 1 (for example ‘k’) for bits 00 or option 2 (for example ‘K’) for bits 01 or option 3 (for example ‘q’) for bits 10 or option 4 (for example ‘Q’) for bits 11. On the other hand, if the selected method for steganography is by using the transliterated text, i.e. roman text; we incrementally find the presence of any special character in Table 5. We take the next bit from the bit stream and select the corresponding form of the special character based on the bit. The selected form is appended with the output stego text and the search for the next special character in the input text continues and all other steps are repeated.

In the receiver side we do the reverse to decode the original text and of course, to generate the secret message bits. Let the selected method of steganography is by using the original Bengali text. We look for an occurrence of the roman representation of any special characters in the Table 4 (for example ‘a’ or ‘A’) in the document. This has the capacity of hiding 1 bit and we put আ in the output text and place bit 0 on the output bit stream if the character is ‘a’ or place 1 if the character is ‘A’. On the other hand, if the found special character has higher capacity (for example ‘k’ or ‘K’ or ‘q’ or ‘Q’), the corresponding character in the code-map is appended with the output text (for example ক) and the capacity amount of bits are placed on the secret bit stream (for Example 00 for ‘k’, 01 for ‘K’, 10 for ‘q’, and 11 for ‘Q’). On the other hand, if the selected method for steganography is by using the transliterated text, i.e. roman text, we incrementally find the presence of any special character in Table 5. The corresponding bit for the special character in Table 5 is appended with the secret bit stream. Then the search for a special character in the input text continues and all other steps are repeated.

These alternation of characters via transliteration are very rarely noticeable and differentiable by any human reader. Any unintended user can never realize and detect this alteration of these special characters in the document. In any case, if someone can realize the alternation, it is not simple to identify whether the bit is 1 or 0 for an alternation as the code-map is built only prior to a communication step and for each special character the corresponding bit(s) change and also the form of the output character in the stego text. For example, in roman representation of আ in one communication (‘A’, ‘a’) may represent bits (0, 1) and in the next they may represent (1, 0). Hence, the extraction of the secret message is far more difficult for an unintended user. Algorithm 1 and Algorithm 2 summarizes the encoding and decoding schemes of the proposed steganography technique.

A Java application is developed, which can switch between the encoder and the decoder functionalities of the proposed data-hiding scheme. The user need to choose between the two proposed method and also a code-map out of number of possibilities. In the encoder, a secret message is taken as an input from a text field. The input or carrier text can be written in the text area or can be loaded from a saved file in the disk. The output stego text is given on a separate text area, which can also be saved to the disk. The sender either can copy the stego text from the text area and paste it in the desired medium of communication, for example in an e-mail, or can directly send the saved file to the receiver. Similarly, in the decoder the stego text can be paste on a text area or can be loaded from a file. Fig. 1 and Fig. 2 respectively show the screen shots of the developed encoder and decoder functionality of the application.

For interested readers, a practical case is included in Appendix. It includes an original Bengali text used as the input of the steganography algorithm, the corresponding transliterated text

**Algorithm 1:** Hiding secret bits

```
Data: Cover digital document, Secret bit stream
Parameter: Method, Code-map
Result: Stego digital document
$i = 1$;
foreach character in the input text do
    if the character is a special character in the code-map then
        if method == Bengali then
            $n$ = capacity of the character according to Table 4;
        else
            $n = 1$;
        end
        if $i \leq$ length of bit stream then
            bits = $n$ bits from the secret bit stream starting at $i$;
        else
            bits = take $n$ 0s;
        end
        $i = i + n$;
        add the roman character(s) corresponding to bits according to the code-map
          to the stego text;
    else
        add the standard roman character(s) corresponding to the input character
          according to Table 1 to the stego text;
    end
end
```

**Algorithm 2:** Extracting bits from the received digital document

```
Data: Stego digital document
Parameter: Method, Code-map
Result: Cover digital document, Secret bit stream
$i = 1$;
while the end of the input text is not reached do
    $s$ = the next roman characters combination, starting at $i$, corresponding to a
      special character in the code-map;
    $k$ = index of $s$ in the input text;
    while $i < k$ do
        $c$ = the character at position $i$ in the input text;
        if method == Bengali then
            add the corresponding Bengali character of $c$ according to Table 1 to the
              output text;
        else
            add $c$ to the output text;
        end
        $i = i + 1$;
    end
    if method == Bengali then
        add the corresponding Bengali character corresponding to $s$ according to
          code-map to the output text;
    else
        add the corresponding roman character corresponding to $s$ according to
          code-map to the output text;
    end
    add the bit(s) corresponding to $s$ according to code-map to the secret bit stream;
    $i = k$ + length of $s$;
end
Delete the consecutive 0s at the end of the decoded secret message, which fail to
  construct a byte or construct a byte with all bits 0;
```

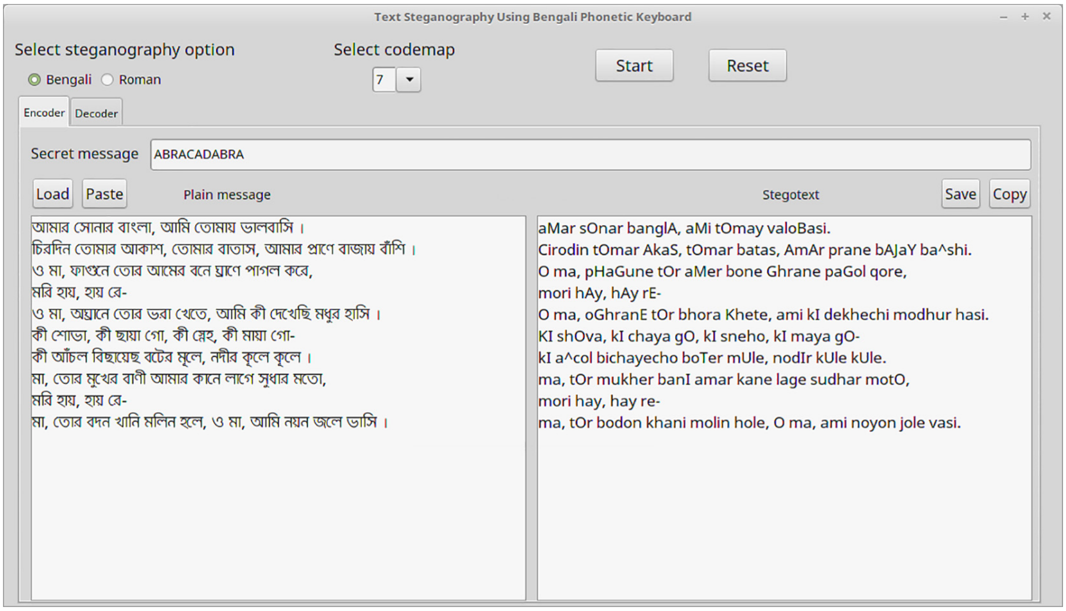
**Table 5**
An example code-map for hiding binary bits in transliterated text.

| Roman | Option 1 (bit 0) | Option 2 (bit 1) |
|---|---|---|
| a | a | A |
| b | b | B |
| c | C | c |
| e | e | E |
| f | f | F |
| h | H | h |
| j | J | j |
| k | K | k |
| l | l | L |
| m | M | m |
| p | p | P |
| q | q | Q |
| v | v | V |
| y | Y | y |
| z | z | Z |

that hides no secret data, a secret message of 40 bytes which is converted to it's equivalent bit stream for steganography, the particular codemap used during this steganography, and the corresponding stego text.

### 4.3. Steganalysis

Two basic approaches for detection of steganography are visual attacks and statistical attacks (Westfeld and Pfitzmann, 2000). A visual attack approach makes use of the human ability to clearly differentiate between a normal text and some obviously uncommon patterns in a text. On the other hand, statistical attacks analyze data, recognize patterns, and classify the texts using some mathematical and statistical theories. Often, these rely on differ-



Text Steganography Using Bengali Phonetic Keyboard

Select steganography option: ● Bengali ○ Roman

Select codemap: 7

Secret message: ABRACADABRA

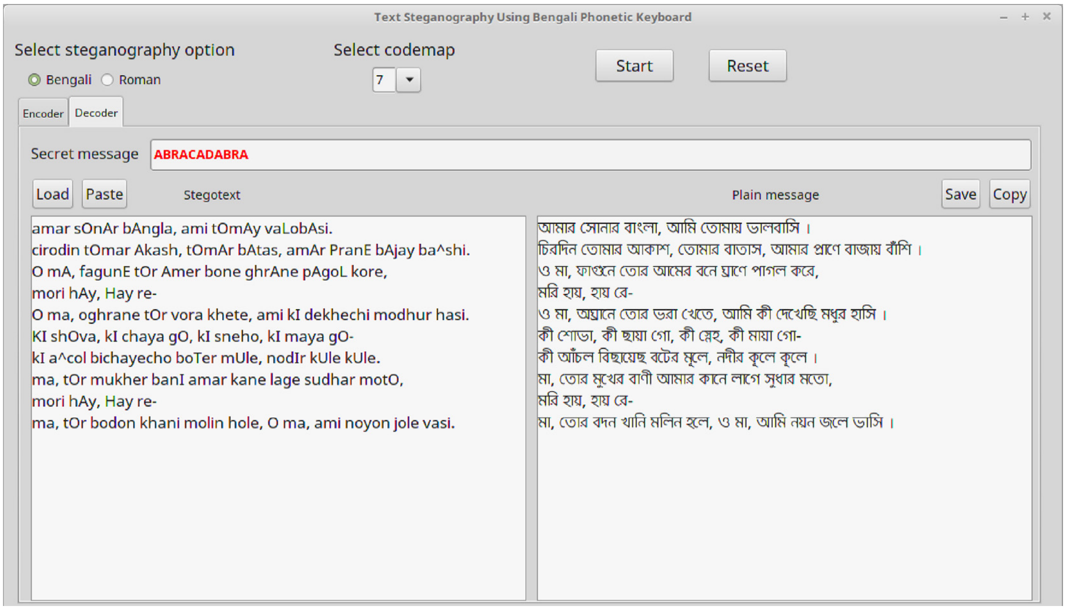
Plain message

আমার সোনার বাংলা, আমি তোমায় ভালবাসি ।
চিরদিন তোমার আকাশ, তোমার বাতাস, আমার প্রাণে বাজায় বাঁশি ।
ও মা, ফাগুনে তোর আমের বনে ঘ্রাণে পাগল করে,
মরি হায়, হায় রে-
ও মা, অঘ্রানে তোর ভরা খেতে, আমি কী দেখেছি মধুর হাসি ।
কী শোভা, কী ছায়া গো, কী স্নেহ, কী মায়া গো-
কী আঁচল বিছায়েছ বটের মূলে, নদীর কূলে কূলে ।
মা, তোর মুখের বাণী আমার কানে লাগে সুধার মতো,
মরি হায়, হায় রে-
মা, তোর বদন খানি মলিন হলে, ও মা, আমি নয়ন জলে ভাসি ।

Stegotext

aMar sOnar banglA, aMi tOmay valoBasi.
Cirodin tOmar AkaS, tOmar batas, AmAr prane bAJaY ba^shi.
O ma, pHaGune tOr aMer bone Ghrane paGol qore,
mori hAy, hAy rE-
O ma, oGhranE tOr bhora Khete, ami kI dekhechi modhur hasi.
KI shOva, kI chaya gO, kI sneho, kI maya gO-
kI a^col bichayecho boTer mUle, nodIr kUle kUle.
ma, tOr mukher banI amar kane lage sudhar motO,
mori hay, hay re-
ma, tOr bodon khani molin hole, O ma, ami noyon jole vasi.

**Fig. 1.** Encoder of the data hiding application.



Text Steganography Using Bengali Phonetic Keyboard

Select steganography option: ● Bengali ○ Roman

Select codemap: 7

Secret message: ABRACADABRA

Stegotext

amar sOnAr bAngla, ami tOmAy vaLobAsi.
cirodin tOmar Akash, tOmAr bAtas, amAr PranE bAjay ba^shi.
O mA, fagunE tOr Amer bone ghrAne pAgoL kore,
mori hAy, Hay re-
O ma, oghrane tOr vora khete, ami kI dekhechi modhur hasi.
KI shOva, kI chaya gO, kI sneho, kI maya gO-
kI a^col bichayecho boTer mUle, nodIr kUle kUle.
ma, tOr mukher banI amar kane lage sudhar motO,
mori hAy, Hay re-
ma, tOr bodon khani molin hole, O ma, ami noyon jole vasi.

Plain message

আমার সোনার বাংলা, আমি তোমায় ভালবাসি ।
চিরদিন তোমার আকাশ, তোমার বাতাস, আমার প্রাণে বাজায় বাঁশি ।
ও মা, ফাগুনে তোর আমের বনে ঘ্রাণে পাগল করে,
মরি হায়, হায় রে-
ও মা, অঘ্রানে তোর ভরা খেতে, আমি কী দেখেছি মধুর হাসি ।
কী শোভা, কী ছায়া গো, কী স্নেহ, কী মায়া গো-
কী আঁচল বিছায়েছ বটের মূলে, নদীর কূলে কূলে ।
মা, তোর মুখের বাণী আমার কানে লাগে সুধার মতো,
মরি হায়, হায় রে-
মা, তোর বদন খানি মলিন হলে, ও মা, আমি নয়ন জলে ভাসি ।

**Fig. 2.** Decoder of the data hiding application.

ences between expected number of some event occurrences and number of its real occurrences. As the statistical approach is basically implemented into computer software, it is more effective and applied more frequently.

As transliteration is a method of substitution of the letters in one language by letters in another in predictable ways, the proposed technique is a substitution based steganography method. The fact that there are fewer high-frequency words in stego texts than in normal texts is utilized in Meng et al. (2010) to detect translation based text steganography. A Support Vector Machine (SVM) classifier is used to classify given texts to normal texts and stego texts based on the frequency differences between normal texts and stego texts. The transliteration scheme can be considered as a character level translation in a narrow sense and we can take into account the frequency of characters instead of words for a similar steganalysis for our approach. In Zhao et al. (2009) a steganalysis method is proposed to detect the existence of hidden information using character substitution in mixed texts. This also utilizes SVM as a classifier to classify the alphabet characteristic vector inputs. It also considers that the steganographic process alters the ratio of abnormal characters to normal characters. However, this considers only the comma (,) characters (the English version or the Chinese version), where our approach considers the complete set of the roman alphabet.

Linguistic steganography utilizes rules and grammars of natural languages, while preserving the syntactic and semantic correctness of cover texts. Due to shortcomings of computer natural language processing capability, linguistic steganalysis is a challenging task. Steganalysis for synonym substitution steganography methods takes into account the synonym pair attributes and detects stego text based on the frequency (Xiang et al., 2014) or relative frequency (Chen et al., 2011). The scheme in Chen et al. (2011) introduces context clusters to estimate the context fitness and show how to use the statistics of context fitness values to distinguish between normal texts and stego texts. Although our method utilizes some features of a natural language, it should not be classified as a linguistic approach. For example, the choice of a particular substituting character in the stego text has no contextual side effects, which is the case for choice of a particular substituting word in synonym steganography. Also, as our method does not utilize statistical generation, it is free of corresponding statistical attacks as well.

Secret bits are encoded by choosing a particular character from a number of alternatives during transliteration of a Bengali character. Mostly, the case of the Roman characters are utilized to construct the set of alternatives for a Bengali character. Hence, the steganalysis for the proposed method should focus on the statistics of the case of used letters in normal texts that contain no secret data as well as stego text. It should be noted that neither semantics nor context apply for transliteration to consider a transliterated text to be considered more standard and subsequently a normal text. One may expect that a normal text should be in ‘sentence case’, i.e. the first character of a sentence is in capital and the others in small letters. Notice that this can not be fulfilled by transliteration procedure. However, in possible cases choosing a capital letter for the first character of a sentence may be considered more natural and the definition of normal text may change. Nevertheless, we consider transliteration that uses the first alternative for each Bengali character generates the normal text.

Let $n_u$ and $n_l$ respectively represent the number of capital letters and small letters in a transliterated text. We define the ratio of capital to small letters $r$, which is a length independent quantity, as

$$r = \frac{n_u}{n_l}. \tag{1}$$

In typical situations we expect $0 \leqslant r \leqslant 1$. As the proposed method deliberately alters the case for many characters in the transliterated text, a primary steganalysis approach should take into account the frequency distribution of $r$ in normal text and in stego text. As in the analysis methods discussed above, if the distributions of $r$ in normal texts and in stego texts vary significantly we can conclude that the proposed method is detectable and undetectable otherwise. A complementary approach may be to look into such statistics of each character in the Roman alphabet. For this we compute the % *occurrence* of a character $i$ as

$$occurrence = \frac{k_i}{N} \times 100\%, \tag{2}$$

where $k_i$ is the number of occurrence of character $i$ in a transliterated text and $N$ is the total length of that text.

Notice that the mixed use of capital and small letters is common in transliteration (e.g. the transliterated plain text in Appendix which contains no secret data) and also for some cases in normal English text (e.g. the first letter of a proper noun, the first letter of a sentence, abbreviations). Hence, the mixed use of capital and small letters should not raise an additional suspicion of hidden data.

## 5. Results and discussions

To assess the performance of the proposed method, several test runs were performed with the developed application software. Input texts were typed directly with the help of phonetic Bengali typing software, as well as copied from Bengali on-line newspapers. The developed applications hide and extract secret information with 100% accuracy in all of the performed test cases. In total 200 tests were performed.

We proposed two attack models for steganalysis of the transliteration based text steganography method in the previous section. The outcomes are detailed below.

### 5.1. Visual attack

Output of a sample test was saved for an open survey. The participants were requested to read the stego text and asked what they think about the text. The survey question was intentionally made rather subjective to extract the actual feeling and emotion of the participants. The main goal is to analyze the possibility of seeing or feeling anything wrong with the supplied text or document, which is a primary sign ineffectiveness of a steganography method. The size of the population for the survey was 28. The population consists of a mix of people who vary from the basic to advanced knowledge about Unicode and phonetic keyboard, but all with significant computer literacy. Table 7 summarizes the outcome of the survey.

We analyzed and categorized the survey out come as the primary groups: i) whether interested in the content, ii) identification of roman representation, iii) special attention, and iv) emotion. It is observed that indeed the participants focused on the content of the text (86%) and 64% only looked at the content only. Only 14% mentioned about the transliteration of the text. Around 21% of the participants noticed some twisting of the text by mixing small and capital letters, or by the vowels but did not suspect any special reason for this. The expressed emotions make this evident, where they mostly specified their disliking and annoyance to read transliterated text. Only one participant specified that some sort of user experience is being evaluated through the survey. Thus, the conclusion of the survey is that the generated stego text was not suspected for secret message by any participant and hence, the

**Table 6**
The derived usage statistics of roman characters in transliterated Bengali text.

| Char | % | Influenced Bengali character |
|---|---|---|
| a | 12.97 | আ |
| b | 4.25 | ব, ভ |
| c | 1.27 | চ, ছ |
| e | 9.00 | ঈ, এ |
| f | 0.28 | ফ |
| h | 7.81 | খ, ঘ, ছ, ঝ, ঠ, ঢ, থ, ধ, ফ, ভ, শ, ষ, হ, ঢ় |
| j | 1.06 | জ, ঝ |
| k | 5.02 | ক, খ |
| l | 3.09 | ল |
| m | 3.19 | ম |
| p | 3.19 | প, ফ |
| q | 5.02 | ক, খ |
| v | 0.44 | ভ |
| y | 1.90 | য় |
| z | 1.18 | য |

**Table 7**
Outcome of the user survey about a sample text with hidden information.

| Participant No. | Interest in the content? | Identifies roman representation? | Special attention | Expressed emotion |
|---|---|---|---|---|
| 1 | | | vowels | |
| 2 | Y | | | |
| 3 | Y | | | |
| 4 | Y | Y | | annoying |
| 5 | | | vowel O | |
| 6 | Y | | | |
| 7 | Y | | | |
| 8 | Y | | | |
| 9 | Y | Y | | |
| 10 | | | mixing small/capital letters | difficulty to read |
| 11 | Y | | | |
| 12 | Y | | capital letters | keyboard/software issue |
| 13 | Y | | | |
| 14 | Y | | | |
| 15 | Y | | | |
| 16 | Y | | | |
| 17 | Y | Y | capital letters | user evaluation |
| 18 | Y | | | |
| 19 | Y | | | |
| 20 | Y | | | |
| 21 | Y | Y | | dislike |
| 22 | Y | | | |
| 23 | Y | | | |
| 24 | Y | | | |
| 25 | Y | | | |
| 26 | Y | | | |
| 27 | Y | | | |
| 28 | | | emphasis on particular node | |
| Total | 86% | 14% | 21% | 18% |

proposed method can be reliably used as a steganography technique.

### 5.2. Statistical attack

We can evaluate the proposed steganography method to be easily detectable if the statistics of capital to small letters in stego texts differs significantly from that of normal texts. Otherwise, we can grant the steganography to be undetectable.100 Bengali texts of size around 8 Kilo Bytes are used for experiments. These texts are from a Bengali corpus dataset (http://scdnlab.com/corpus) and covers the full range of types, namely *accident*, *art*, *crime*, *economics*, *education*, *entertainment*, *environment*, *international affairs*, *opinions*, *politics*, *science and technology*, and *sports*.

### *5.2.1. Distribution of capital to small letter ratio*

Fig. 3 presents the capital to small letter ratio of normal text (hiding no secret data) and for stego text hiding different amount of secret data. Notice that although the capital to small letter ratio fluctuates in different texts, the distribution is quite uniform. The solid lines are for stego text by method 1 and the dashed lines for stego text by method 2. Notice that method 1 has slightly lower capital to small letter ratio. This can be attributed to the fact that method 1 additionally use different Roman character alternatives other than swapping the case of a character. It is evident that the ratio in stego text hiding 80 bytes or 160 bytes is very much higher than that in normal texts and we may consider the method unsuitable with so high capacity. Of course, this may be considered apparently safe for lower capacity (e.g. 40 bytes) as we observe considerable overlap of the distribution with that of the normal text.

Now let us consider the distribution of the capital to small letter ratio among the 100 texts. Fig. 4 demonstrates the probability distribution function derived using Gaussian kernel density estimation, which are alternatives to histograms that have disadvantages such as the non-smoothness, dependence on the width of the bins as well as the end points of the bins (Silverman, 1986). Again, the solid lines are for stego text by method 1 and the dashed lines are for stego text by method 2. The same observations as in Fig. 3 stated in the previous paragraph also apply here. It indicates that the stego text hiding 80 bytes or 160 bytes can be easily detected and lower capacity data hiding may be considered safe due to the considerable overlap of the distributions.

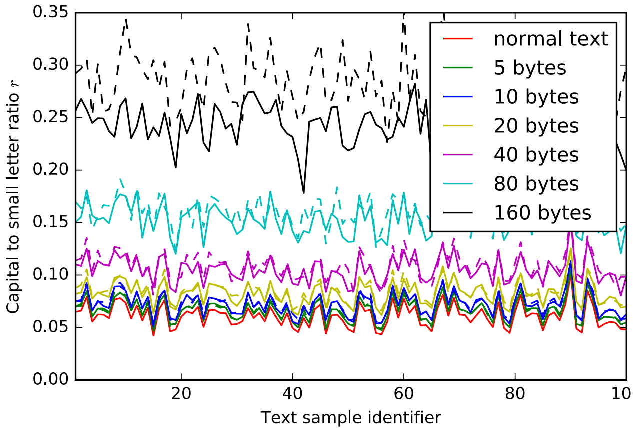


**Fig. 3.** Capital to small letter ratio in different transliterated text with varying data hiding capacity.

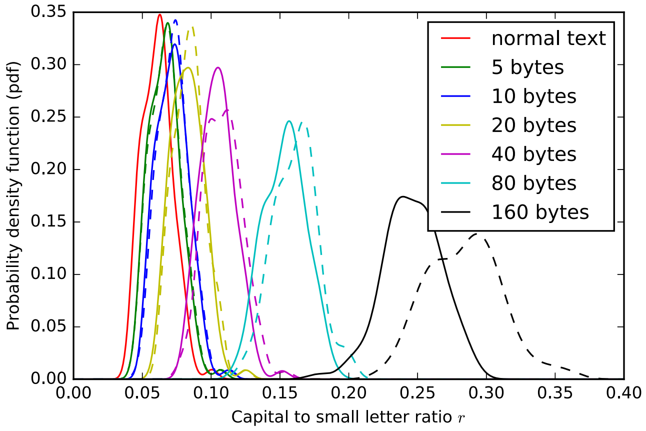


**Fig. 4.** Distribution of capital to small letter ratio in different transliterated text with varying data hiding capacity.

### *5.2.2. SVM classification*

A support vector machine (SVM) implemented in Chang and Lin (2011) is used to classify a set of transliterated text consisting of normal text and stego text. The training set consists of 40 normal text and 40 stego text with varying amount of hidden data. Specifically, steganography with hiding capacity 20, 30, 40, 50, 60, 70, 80, and 90 bytes were performed and 5 stego text from each group was taken in the training set. The remaining 20 texts were used for testing. To apply SVM, either no data was hidden (normal text) or some bytes were hidden (stego text) in these texts.

The capital to small letter ratio $r$ (see (1)) is used as a scalar training vector. ROC curves in Fig. 5 demonstrates the capability of the SVM to detect the steganography. We observe that hiding 20 bytes is almost unlikely to be detected and the performance of the SVM is almost equal to a random classifier. On the other hand, hiding 70 bytes or more is instantaneously detected by the SVM. Up to 40 bytes the accuracy of the SVM is below 70% which is considered as a poor performance for a classifier and we can consider that our steganography has low risk to be detected. Note that this also agrees with the findings of distribution analysis.

The character occurrence (see (2)) is used as a multi-dimensional training vector. The dimension is 52, i.e. 26 for lower letters and 26 for capital letters. ROC curves in Fig. 6 presents the

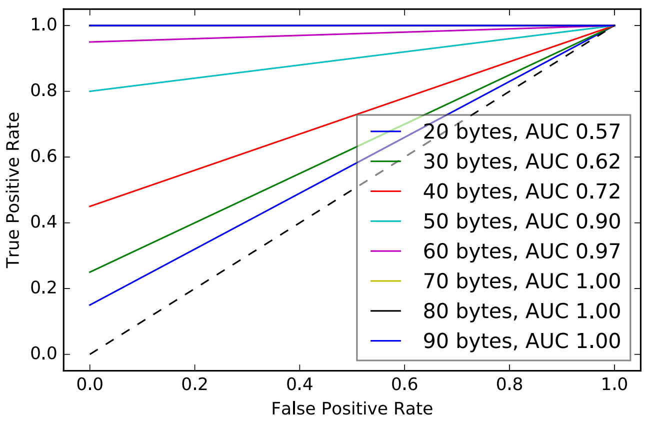


**Fig. 5.** ROC of the SVM classifier considering capital to small letter ratio in transliteration, hiding different number of secret bytes. The dotted line indicates a purely random classification.

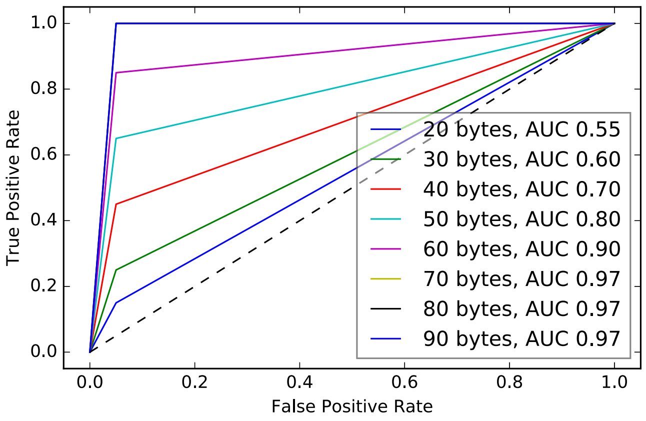


**Fig. 6.** ROC of the SVM classifier considering the occurrence of each letter in transliteration, hiding different number of secret bytes. The dotted line indicates a purely random classification.

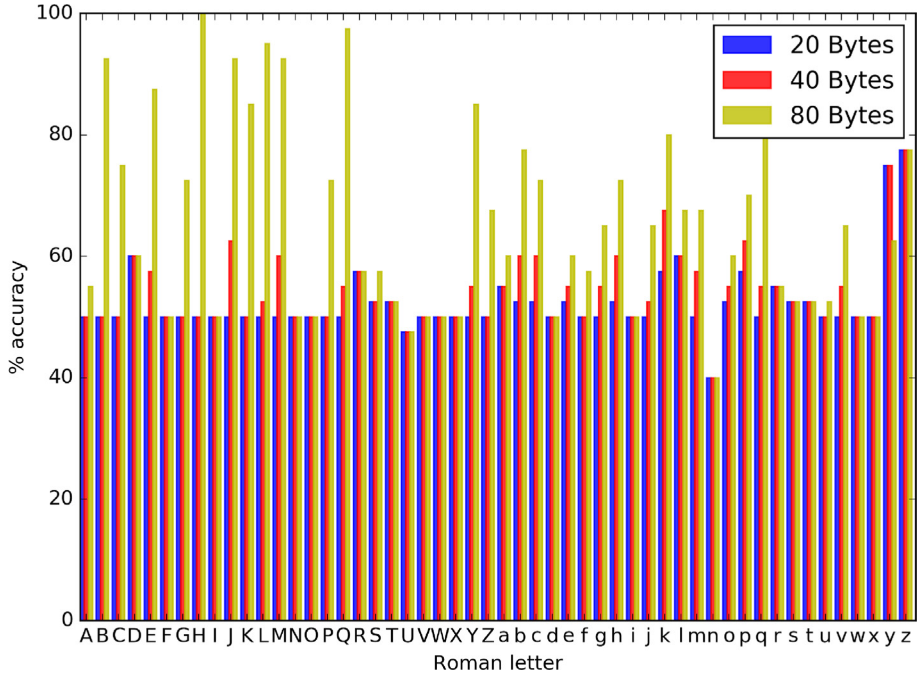


**Fig. 7.** Accuracy of the SVM classifier considering the occurrence of each individual letter in transliteration, hiding different number of secret bytes. A 50% accuracy indicates a purely random classification.

capability of the SVM to detect the steganography. We have exactly the same observations as for the SVM using the capital to small letter ratio. Thus the steganalysis using more information with finer features does not enhance the SVM capability in this context.

Considering each individual roman character does not help an SVM in detecting the presence of hidden data in the stego text when the amount of hidden data is 40 bytes in a 4 KB text, which is evident through the accuracy bars in Fig. 7. SVM classification was used for each roman character and the occurrence of each character was used as a scalar training vector. Note that the maximum accuracy for 40 bytes hidden data is around 75% while with 80 bytes this reaches 100%.

Fig. 8 compares the capital to small letter ratio and the character occurrence as the SVM input data. It presents the obtained AUC values for varying data hiding capacity. We observe that SVM with character level information is slightly worse than the SVM with overall capital to small letter ratio information, which is a contrary to the expectation. In fact, SVM with character level information deals with 52-dimensional data and may suffer from the drawbacks of processing high dimensional data and consequently result in poor classification performance.

Above classification experiments were repeated for another 100 Bengali texts of varying size in the range 10 kilo bytes to 100 kilo bytes. Again, different amount of secret data was hidden resulting in 0.5%, 1%, 2%, and 4% of capacity. The training set consists of 40 texts containing no secret data and 10 texts from each of the group of stego texts with different capacity, where 20 texts from each

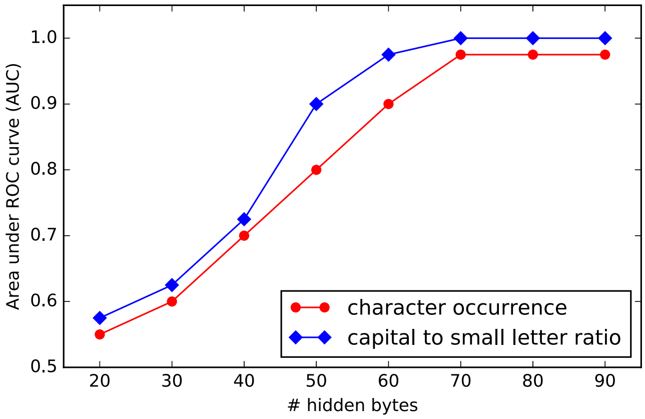


**Fig. 8.** Area under ROC curves (AUC) in Fig. 5 and Fig. 6 for varying amount of hidden data in stego text.

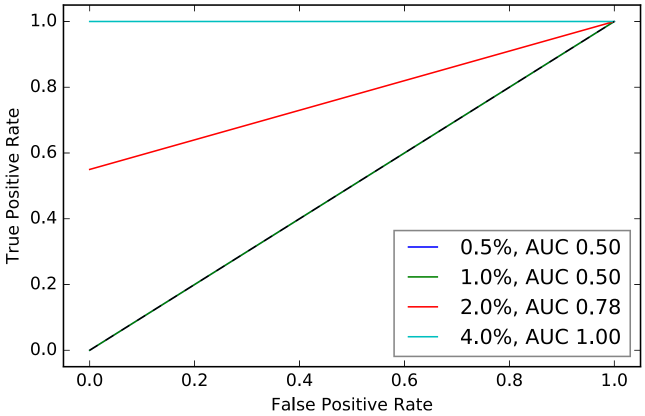


**Fig. 9.** ROC of the SVM classifier considering capital to small letter ratio in transliteration, hiding different number of secret bytes. The dotted line indicates a purely random classification. Varying length texts are used as the cover text. Note that the ROC for data hiding capacity of 1% as well as 0.5%, and purely random classification (all having AUC 0.50) cannot be distinguished in the figure.

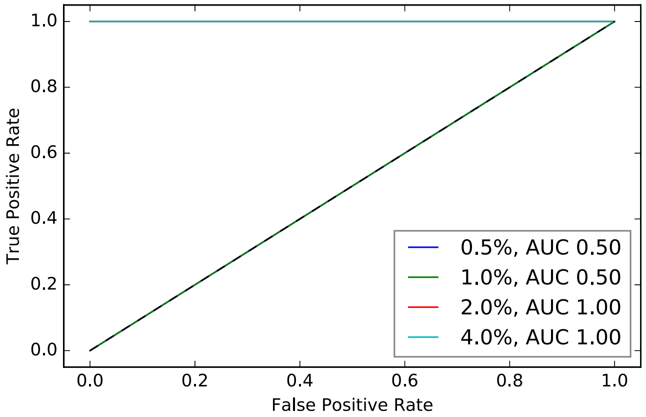


**Fig. 10.** ROC of the SVM classifier considering the occurrence of each letter in transliteration, hiding different number of secret bytes. The dotted line indicates a purely random classification. Varying length texts are used as the cover text. Note that the ROC for data hiding capacity of 1% as well as 0.5%, and purely random classification (all having AUC 0.50) cannot be distinguished in the figure. Similarly, the ROC for data hiding capacity of 4% and 2% (both having AUC 1.00) cannot be distinguished.

**Table 9**
Comparison of the average size of stego text with varying hiding capacity and the cover text. Here, variable size Bengali text is used.

| Cover text | 0.5% | 1% | 2% | 4% |
|---|---|---|---|---|
| 6142 | 6137 | 6132 | 6123 | 6108 |

group is used for testing. Fig. 9, Fig. 10, and Fig. 11 present the SVM classification performance using the overall capital to small letter ratio, the character occurrence vector, and the occurrence of each individual roman character respectively. Note that these correspond to Fig. 5, Fig. 6, and Fig. 7 respectively. We observe that SVM classification performance is poor for hiding capacity of 1% or less, while data hiding with higher capacity is detected more accurately. Note that the data hiding capacity of 1% is equivalent to hiding 40 bytes data in a 4 kilo bytes cover text (8 kilo bytes Bengali text).

Hence, we can summarize that the maximum capacity of the proposed steganography ensuring very low risk of detectability is 40 bytes in a 4 Kilo Bytes stego text, i.e. 1 byte per 100 bytes. To be more exact, 1.2 bytes can be hidden per 100 bytes in the transliterated stego text. As discussed in Section 1 the compression ratio is around 2.4 by transliteration and hence, a Bengali text of 8 kilo bytes is transliterated into a 3.4 kilo byte roman text. Although the reader may consider it as a poor capacity, such low capacity is very common in text steganography systems and is considered fair, particularly if undetectability is guaranteed.

The proposed method uses alternative roman character combination to represent the same Bengali character. Hence, no extra

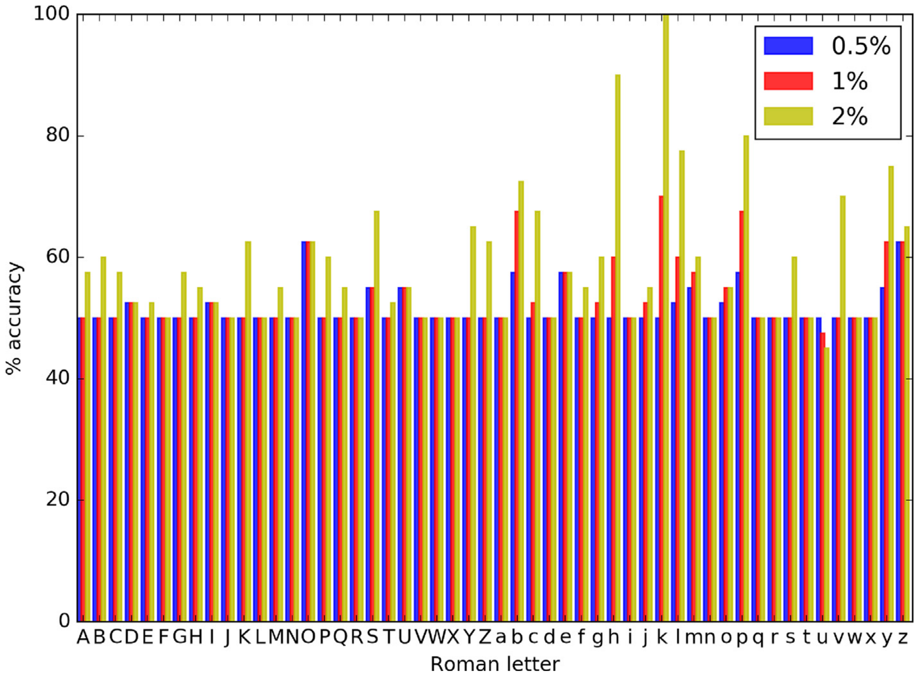


**Fig. 11.** Accuracy of the SVM classifier considering the occurrence of each individual letter in transliteration, hiding different amount of secret bytes in varying length transliterated text. A 50% accuracy indicates a purely random classification. Varying length texts are used as the cover text.

**Table 8**
Comparison of the average size of stego text with varying hiding capacity different and the cover text. Size of all Bengali text is 8 kilo bytes.

| Cover text | 20 B | 30 B | 40 B | 50 B | 60 B | 70 B | 80 B | 90 B |
|---|---|---|---|---|---|---|---|---|
| 3318 | 3315 | 3314 | 3312 | 3311 | 3310 | 3308 | 3308 | 3307 |

character is added in the stego text and the length of the cover text and the stego text should not differ significantly. This is evident in Table 8 and Table 9, which compare the lengths of cover text and the corresponding stego text. It is not surprising that with higher capacity the length of stego text can decrease a little bit, as it uses more single-letter version instead of two-letter version of roman letters for some Bengali characters (e.g. ফ, ভ, শ).

Finally, in Table 10 we compare the proposed method with other text steganography methods based on various languages having non-roman alphabets which are discussed in this paper. Observe that only our method is supported by a detailed statistical steganalysis procedure and hence the reported capacity is guaranteed. A higher capacity without such an analysis may not be useful. For example, the capacity of our method is 7.88% (see Section 4.2), if steganalysis is ignored. Thus, the proposed method is the most reliable among the methods in it's category. Table 11.

## 6. Conclusions

We discussed a novel technique of hiding information in Bengali digital document. These documents are very common to any organization or on the Internet. The initial results obtained for the proposed system are very appealing and we are confident that our method can be a good choice for text steganography.

Other interesting features of this language or the alphabet can be exploited efficiently for steganography purposes and deserves significant research. The proposed method can be further enhanced. For example, by incorporating all other available transliteration systems for Bengali, the possible number of code-maps can be almost infinity.

Bengali is used as an example medium in this paper for the proposed technique. As the main idea is quite simple and straight forward, it can easily adapted and extended to use any of the language having non-roman alphabet. A generic and language-independent framework may be developed, which can lead to an industry standard steganography system.

## Acknowledgements

The author thankfully acknowledges Md. Abu Shahriar Ratul and Farhana Rumzum Bhuiyan for conducting the survey, used as the visual attack in steganalysis.

**Table 10**
Comparison of text steganography methods using different languages having non-roman alphabets.

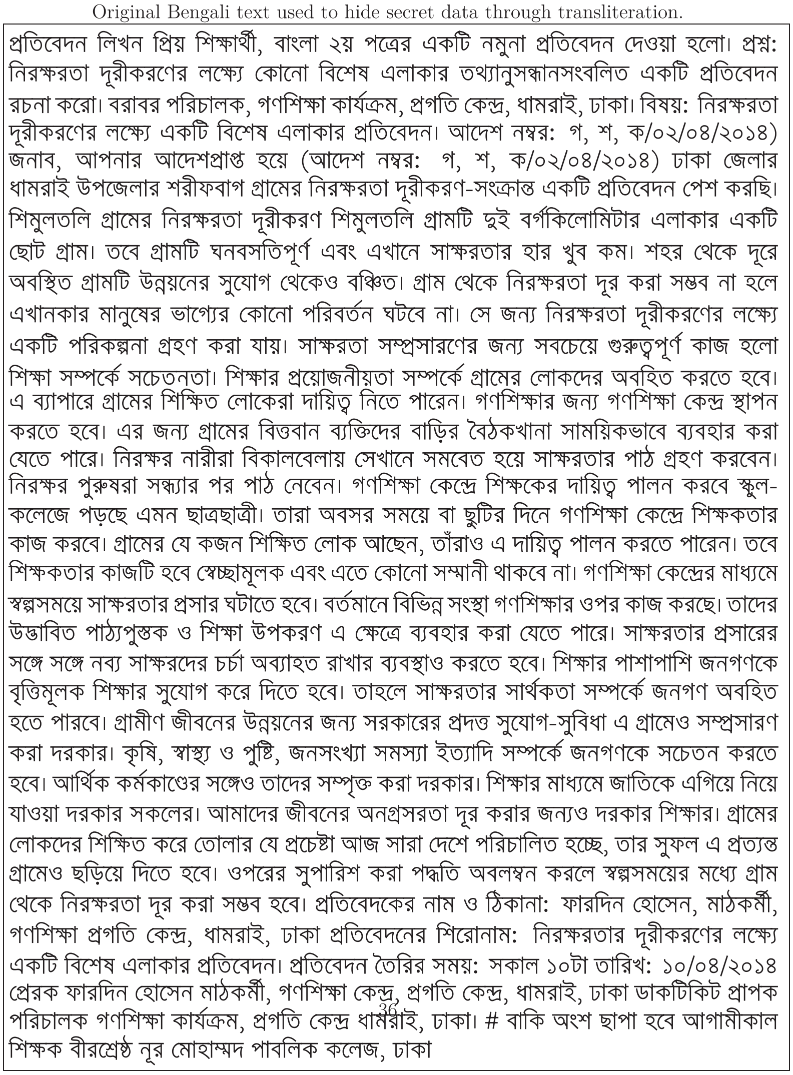
| Method (Language) | Capacity (%) | Robustness | Steganalysis |
|---|---|---|---|
| Shirali-Shahreza and Shirali-Shahreza (2006) (Arabic/Persian) | 1.41 | fails in retyping, OCR, font change | no |
| Memon et al. (2005) (Arabic/Urdu) | not reported | fails in retyping, OCR, font change | no |
| Aabed et al. (2007) (Arabic) | 3.27, 1.22 | easily detectable by visual attack | no |
| Gutub et al. (2008) (Arabic) | 4.5 | easily detectable by visual attack | no |
| Khairullah (2018) (Bengali) | 1.25 | passed visual attacks | no |
| Xinmei et al. (2010) (Chinese) | 2.1, 1.7 | low detectability by visual attacks | no |
| This work (Bengali) | 1.2 | passed visual attacks | yes |

**Table 11**
Used codemap with bit combinations.

| Character | Option 1 | Option 2 | Option 3 | Option 4 | Option 5 | Option 6 | Option 7 | Option 8 | Capacity |
|---|---|---|---|---|---|---|---|---|---|
| আ | a (0) | A (1) | | | | | | | 1 |
| ঈ | I (00) | eE (01) | Ee (10) | ee (11) | | | | | 2 |
| উ | u (0) | oo (1) | | | | | | | 1 |
| এ | e (0) | E (1) | | | | | | | 1 |
| ক | k (00) | K (01) | q (10) | Q (11) | | | | | 2 |
| খ | kh (000) | Kh (001) | kH (010) | KH (011) | qh (100) | qH (101) | Qh (110) | QH (111) | 3 |
| গ | g (0) | G (1) | | | | | | | 1 |
| ঘ | gh (00) | Gh (01) | gH (10) | GH (11) | | | | | 2 |
| চ | c (0) | C (1) | | | | | | | 1 |
| ছ | ch (00) | Ch (01) | CH (10) | cH (11) | | | | | 2 |
| জ | J (0) | j (1) | | | | | | | 1 |
| ঝ | jh (00) | jH (01) | Jh (10) | JH (11) | | | | | 2 |
| ঠ | TH (0) | Th (1) | | | | | | | 1 |
| ঢ | DH (0) | Dh (1) | | | | | | | 1 |
| থ | th (0) | tH (1) | | | | | | | 1 |
| ধ | dh (0) | dH (1) | | | | | | | 1 |
| প | P (0) | p (1) | | | | | | | 1 |
| ফ | ph (00) | pH (01) | Ph (10) | f (11) | | | | | 2 |
| ব | B (0) | b (1) | | | | | | | 1 |
| ভ | bh (00) | v (01) | Bh (10) | bH (11) | | | | | 2 |
| ম | m (0) | M (1) | | | | | | | 1 |
| য | Z (0) | z (1) | | | | | | | 1 |
| ল | l (0) | L (1) | | | | | | | 1 |
| শ | sh (0) | S (1) | | | | | | | 1 |
| ষ | SH (0) | Sh (1) | | | | | | | 1 |
| হ | h (0) | H (1) | | | | | | | 1 |
| ঢ় | RH (0) | Rh (1) | | | | | | | 1 |
| য় | Y (0) | y (1) | | | | | | | 1 |

## Appendix A.

Original Bengali text used to hide secret data through transliteration.



প্রতিবেদন লিখন প্রিয় শিক্ষার্থী, বাংলা ২য় পত্রের একটি নমুনা প্রতিবেদন দেওয়া হলো। প্রশ্ন: নিরক্ষরতা দূরীকরণের লক্ষ্যে কোনো বিশেষ এলাকার তথ্যানুসন্ধানসংবলিত একটি প্রতিবেদন রচনা করো। বরাবর পরিচালক, গণশিক্ষা কার্যক্রম, প্রগতি কেন্দ্র, ধামরাই, ঢাকা। বিষয়: নিরক্ষরতা দূরীকরণের লক্ষ্যে একটি বিশেষ এলাকার প্রতিবেদন। আদেশ নম্বর: গ, শ, ক/০২/০৪/২০১৪) জনাব, আপনার আদেশপ্রাপ্ত হয়ে (আদেশ নম্বর: গ, শ, ক/০২/০৪/২০১৪) ঢাকা জেলার ধামরাই উপজেলার শরীফবাগ গ্রামের নিরক্ষরতা দূরীকরণ-সংক্রান্ত একটি প্রতিবেদন পেশ করছি। শিমুলতলি গ্রামের নিরক্ষরতা দূরীকরণ শিমুলতলি গ্রামটি দুই বর্গকিলোমিটার এলাকার একটি ছোট গ্রাম। তবে গ্রামটি ঘনবসতিপূর্ণ এবং এখানে সাক্ষরতার হার খুব কম। শহর থেকে দূরে অবস্থিত গ্রামটি উন্নয়নের সুযোগ থেকেও বঞ্চিত। গ্রাম থেকে নিরক্ষরতা দূর করা সম্ভব না হলে এখানকার মানুষের ভাগ্যের কোনো পরিবর্তন ঘটবে না। সে জন্য নিরক্ষরতা দূরীকরণের লক্ষ্যে একটি পরিকল্পনা গ্রহণ করা যায়। সাক্ষরতা সম্প্রসারণের জন্য সবচেয়ে গুরুত্বপূর্ণ কাজ হলো শিক্ষা সম্পর্কে সচেতনতা। শিক্ষার প্রয়োজনীয়তা সম্পর্কে গ্রামের লোকদের অবহিত করতে হবে। এ ব্যাপারে গ্রামের শিক্ষিত লোকেরা দায়িত্ব নিতে পারেন। গণশিক্ষার জন্য গণশিক্ষা কেন্দ্র স্থাপন করতে হবে। এর জন্য গ্রামের বিত্তবান ব্যক্তিদের বাড়ির বৈঠকখানা সাময়িকভাবে ব্যবহার করা যেতে পারে। নিরক্ষর নারীরা বিকালবেলায় সেখানে সমবেত হয়ে সাক্ষরতার পাঠ গ্রহণ করবেন। নিরক্ষর পুরুষরা সন্ধ্যার পর পাঠ নেবেন। গণশিক্ষা কেন্দ্রে শিক্ষকের দায়িত্ব পালন করবে স্কুল-কলেজে পড়ছে এমন ছাত্রছাত্রী। তারা অবসর সময়ে বা ছুটির দিনে গণশিক্ষা কেন্দ্রে শিক্ষকতার কাজ করবে। গ্রামের যে কজন শিক্ষিত লোক আছেন, তাঁরাও এ দায়িত্ব পালন করতে পারেন। তবে শিক্ষকতার কাজটি হবে স্বেচ্ছামূলক এবং এতে কোনো সম্মানী থাকবে না। গণশিক্ষা কেন্দ্রের মাধ্যমে স্বল্পসময়ে সাক্ষরতার প্রসার ঘটাতে হবে। বর্তমানে বিভিন্ন সংস্থা গণশিক্ষার ওপর কাজ করছে। তাদের উদ্ভাবিত পাঠ্যপুস্তক ও শিক্ষা উপকরণ এ ক্ষেত্রে ব্যবহার করা যেতে পারে। সাক্ষরতার প্রসারের সঙ্গে সঙ্গে নব্য সাক্ষরদের চর্চা অব্যাহত রাখার ব্যবস্থাও করতে হবে। শিক্ষার পাশাপাশি জনগণকে বৃত্তিমূলক শিক্ষার সুযোগ করে দিতে হবে। তাহলে সাক্ষরতার সার্থকতা সম্পর্কে জনগণ অবহিত হতে পারবে। গ্রামীণ জীবনের উন্নয়নের জন্য সরকারের প্রদত্ত সুযোগ-সুবিধা এ গ্রামেও সম্প্রসারণ করা দরকার। কৃষি, স্বাস্থ্য ও পুষ্টি, জনসংখ্যা সমস্যা ইত্যাদি সম্পর্কে জনগণকে সচেতন করতে হবে। আর্থিক কর্মকাণ্ডের সঙ্গেও তাদের সম্পৃক্ত করা দরকার। শিক্ষার মাধ্যমে জাতিকে এগিয়ে নিয়ে যাওয়া দরকার সকলের। আমাদের জীবনের অনগ্রসরতা দূর করার জন্যও দরকার শিক্ষার। গ্রামের লোকদের শিক্ষিত করে তোলার যে প্রচেষ্টা আজ সারা দেশে পরিচালিত হচ্ছে, তার সুফল এ প্রত্যন্ত গ্রামেও ছড়িয়ে দিতে হবে। ওপরের সুপারিশ করা পদ্ধতি অবলম্বন করলে স্বল্পসময়ের মধ্যে গ্রাম থেকে নিরক্ষরতা দূর করা সম্ভব হবে। প্রতিবেদকের নাম ও ঠিকানা: ফারদিন হোসেন, মাঠকর্মী, গণশিক্ষা প্রগতি কেন্দ্র, ধামরাই, ঢাকা প্রতিবেদনের শিরোনাম: নিরক্ষরতার দূরীকরণের লক্ষ্যে একটি বিশেষ এলাকার প্রতিবেদন। প্রতিবেদন তৈরির সময়: সকাল ১০টা তারিখ: ১০/০৪/২০১৪ প্রেরক ফারদিন হোসেন মাঠকর্মী, গণশিক্ষা কেন্দ্র, প্রগতি কেন্দ্র, ধামরাই, ঢাকা ডাকটিকিট প্রাপক পরিচালক গণশিক্ষা কার্যক্রম, প্রগতি কেন্দ্র ধামরাই, ঢাকা। # বাকি অংশ ছাপা হবে আগামীকাল শিক্ষক বীরশ্রেষ্ঠ নূর মোহাম্মদ পাবলিক কলেজ, ঢাকা

Transliteration of the above Bengali text hiding no secret data and hence the normal text.

p„rtibedn likhn p„riy shik„Shar„thI, bangla 2y pt„rer ekTi nmuna p„rtibedn deOya hlO. p„rsh„n: nirk„Shrta dUrIkrNer lk„Sh„ze kOnO bisheSh elakar tth„zanusn„dhansngblit ekTi p„rtibedn rcna krO. brabr pricalk, gNshik„Sha kar„zk„rm, p„rgti ken„d„r, dhamrai, Dhaka. biShy: nirk„Shrta dUrIkrNer lk„Sh„ze ekTi bisheSh elakar p„rtibedn. adesh nm„br: g, sh, k/02/04/2014) jnab, apnar adeshp„rap„t hye (adesh nm„br: g, sh, k/02/04/2014) Dhaka jelar dhamrai upjelar shrIphbag g„ramer nirk„Shrta dUrIkrN-sngk„ran„t ekTi p„rtibedn pesh krchi. shimultli g„ramer nirk„Shrta dUrIkrN shimultli g„ramTi dui br„gkilOmiTar elakar ekTi chOT g„ram. tbe g„ramTi ghnbstipUr„N ebng ekhane sak„Shrtar har khub km. shhr theke dUre obs„thit g„ramTi un„nyner suzOg thekeO bNG„cit. g„ram theke nirk„Shrta dUr kra sm„bhb na hle ekhankar manuSher bhag„zer kOnO pribr„tn ghTbe na. se jn„z nirk„Shrta dUrIkrNer lk„Sh„ze ekTi prikl„pna g„rhN kra zay. sak„Shrta sm„p„rsarNer jn„z sbceye gurut„bpUr„N kaj hlO shik„Sha sm„pr„ke scetnta. shik„Shar p„ryOjnIyta sm„pr„ke g„ramer lOkder obhit krte hbe. e b„zapare g„ramer shik„Shit lOkera dayit„b nite paren. gNshik„Shar jn„z gNshik„Sha ken„d„r s„thapn krte hbe. er jn„z g„ramer bit„tban b„zk„tider baRir bOIThkkhana samyikbhabe b„zbhar kra zete pare. nirk„Shr narIra bikalbelay sekhane smbet hye sak„Shrtar paTh g„rhN krben. nirk„Shr puruShra sn„dh„zar pr paTh neben. gNshik„Sha ken„d„re shik„Shker dayit„b paln krbe s„kul-kleje pRche emn chat„rchat„rI. tara obsr smye ba chuTir dine gNshik„Sha ken„d„re shik„Shktar kaj krbe. g„ramer ze kjn shik„Shit lOk achen, taîraO e dayit„b paln krte paren. tbe shik„Shktar kajTi hbe s„bec„chamUlk ebng ete kOnO sm„manI thakbe na. gNshik„Sha ken„d„rer madh„zme s„bl„psmye sak„Shrtar p„rsar ghTate hbe. br„tmane bibhin„n sngs„tha gNshik„Shar Opr kaj krche. tader ud„bhabit paTh„zpus„tk O shik„Sha upkrN e k„Shet„re b„zbhar kra zete pare. sak„Shrtar p„rsarer sNg„ge sNg„ge nb„z sak„Shrder cr„ca ob„zaht rakhar b„zbs„thaO krte hbe. shik„Shar pashapashi jngNke brrit„timUlk shik„Shar suzOg kre dite hbe. tahle sak„Shrtar sar„thkta sm„pr„ke jngN obhit hte parbe. g„ramIN jIbner un„nyner jn„z srkarer p„rdt„t suzOg-subidha e g„rameO sm„p„rsarN kra drkar. krriShi, s„bas„th„z O puSh„Ti, jnsngkh„za sms„za it„zadi sm„pr„ke jngNke scetn krte hbe. ar„thik kr„mkaN„Der sNg„geO tader sm„prrik„t kra drkar. shik„Shar madh„zme jatike egiye niye zaOya drkar skler. amader jIbner ong„rsrta dUr krar jn„zO drkar shik„Shar. g„ramer lOkder shik„Shit kre tOlar ze p„rceSh„Ta aj sara deshe pricalit hc„che, tar suphl e p„rt„zn„t g„rameO chRiye dite hbe. Oprer suparish kra pd„dhti oblm„bn krle s„bl„psmyer mdh„ze g„ram theke nirk„Shrta dUr kra sm„bhb hbe. p„rtibedker nam O Thikana: phardin hOsen, maThkr„mI, gNshik„Sha p„rgti ken„d„r, dhamrai, Dhaka p„rtibedner shirOnam: nirk„Shrtar dUrIkrNer lk„Sh„ze ekTi bisheSh elakar p„rtibedn. p„rtibedn tOIrir smy: skal 10Ta tarikh: 10/04/2014 p„rerk phardin hOsen maThkr„mI, gNshik„Sha ken„d„r, p„rgti ken„d„r, dhamrai, Dhaka DakTikiT p„rapk pricalk gNshik„Sha kar„zk„rm, p„rgti ken„d„r dhamrai, Dhaka. # baki ongsh chapa hbe agamIkal shik„Shk bIrsh„reSh„Th nUr mOham„md pablik klej, Dhaka

Corresponding stego text containing the secret data:
“bkifsyodlhmbgbtlvadhcxuqqbhowjbhjzerksft”

p„rtibedn liKhn P„riy SiK„SHar„tHI, bangLa 2y Pt„rer EkTi nMuna p„rtibedn deOYa hLO. p„rsh„n: nirQ„Shrta dUrIkrNer LQ„Sh„ze qOnO BiSeSh ElaQar ttH„zanusn„dHansngblit eqTi P„rtibedn rcna KrO. braBr PricaLK, gNshiK„Sha Kar„zK„rM, p„rgti ken„d„r, dHamrai, Dhaqa. BiShy: nirQ„Shrta dUrIkrNer lq„Sh„ze EKTi BisheSh ElaQar P„rtiBedn. adeS nM„Br: G, S, K/02/04/2014) jnaB, APnar AdeSp„rap„t hYe (adesh nm„br: G, sh, q/02/04/2014) DHaKa jelar dhamrai oopjeLar SrIPhBag G„raMer nirq„Shrta dUrIKrNsngQ„ran„t EkTi P„rtibedn peS qrchi. SiMuLtli g„ramer nirq„Shrta dUrIqrN Simultli G„raMTi dui Br„GQiLOMiTar ELaKar EQTi CHOT G„ram. tbe g„raMTi gHnBstiPUr„N EBng EqHane sak„SHrtar Har qHuB qM. SHr tHeKe dUre oBs„tHit G„ramTi un„nyner suZOG tHeQeO BNG„Cit. G„ram tHeqe nirq„Shrta dUr Qra sM„BhB na HLe EqhanQar manuSher bHag„zer kOnO pribr„tn ghTbe na. se jn„z nirk„Shrta dUrIkrNer lk„Sh„ze ekTi prikl„pna g„rhN kra zay. sak„Shrta sm„p„rsarNer jn„z sbceye gurut„bpUr„N kaj hlO shik„Sha sm„pr„ke scetnta. shik„Shar p„ryOjnIyta sm„pr„ke g„ramer lOkder obhit krte hbe. e b„zapare g„ramer shik„Shit lOkera dayit„b nite paren. gNshik„Shar jn„z gNshik„Sha ken„d„r s„thapn krte hbe. er jn„z g„ramer bit„tban b„zk„tider baRir bOIThkkhana samyikbhabe b„zbhar kra zete pare. nirk„Shr narIra bikalbelay sekhane smbet hye sak„Shrtar paTh g„rhN krben. nirk„Shr puruShra sn„dh„zar pr paTh neben. gNshik„Sha ken„d„re shik„Shker dayit„b paln krbe s„kul-kleje pRche emn chat„rchat„rI. tara obsr smye ba chuTir dine gNshik„Sha ken„d„re shik„Shktar kaj krbe. g„ramer ze kjn shik„Shit lOk achen, taîraO e dayit„b paln krte paren. tbe shik„Shktar kajTi hbe s„bec„chamUlk ebng ete kOnO sm„manI thakbe na. gNshik„Sha ken„d„rer madh„zme s„bl„psmye sak„Shrtar p„rsar ghTate hbe. br„tmane bibhin„n sngs„tha gNshik„Shar Opr kaj krche. tader ud„bhabit paTh„zpus„tk O shik„Sha upkrN e k„Shet„re b„zbhar kra zete pare. sak„Shrtar p„rsarer sNg„ge sNg„ge nb„z sak„Shrder cr„ca ob„zaht rakhar b„zbs„thaO krte hbe. shik„Shar pashapashi jngNke brrit„timUlk shik„Shar suzOg kre dite hbe. tahle sak„Shrtar sar„thkta sm„pr„ke jngN obhit hte parbe. g„ramIN jIbner un„nyner jn„z srkarer p„rdt„t suzOg-subidha e g„rameO sm„p„rsarN kra drkar. krriShi, s„bas„th„z O puSh„Ti, jnsngkh„za sms„za it„zadi sm„pr„ke jngNke scetn krte hbe. ar„thik kr„mkaN„Der sNg„geO tader sm„prrik„t kra drkar. shik„Shar madh„zme jatike egiye niye zaOya drkar skler. amader jIbner ong„rsrta dUr krar jn„zO drkar shik„Shar. g„ramer lOkder shik„Shit kre tOlar ze p„rceSh„Ta aj sara deshe pricalit hc„che, tar suphl e p„rt„zn„t g„rameO chRiye dite hbe. Oprer suparish kra pd„dhti oblm„bn krle s„bl„psmyer mdh„ze g„ram theke nirk„Shrta dUr kra sm„bhb hbe. p„rtibedker nam O Thikana: phardin hOsen, maThkr„mI, gNshik„Sha p„rgti ken„d„r, dhamrai, Dhaka p„rtibedner shirOnam: nirk„Shrtar dUrIkrNer lk„Sh„ze ekTi bisheSh elakar p„rtibedn. p„rtibedn tOIrir smy: skal 10Ta tarikh: 10/04/2014 p„rerk phardin hOsen maThkr„mI, gNshik„Sha ken„d„r, p„rgti ken„d„r, dhamrai, Dhaka DakTikiT p„rapk pricalk gNshik„Sha kar„zk„rm, p„rgti ken„d„r dhamrai, Dhaka. # baki ongsh chapa hbe agamIkal shik„Shk bIrsh„reSh„Th nUr mOham„md pablik klej, Dhaka

## References

Aabed, M.A., Awaideh, S.M., Elshafei, A.R.M., Gutub, A.A., 2007. Arabic diacritics based steganography. IEEE Int. Conf. Signal Process. Commun. 2007, 756–759. https://doi.org/10.1109/ICSPC.2007.4728429.

Acharjee, T., Konwar, A., Ram, R.K., Sharma, R., Goswami, D., 2016. Xorsteg: a new model of text steganography. International Conference on Communication and Electronics Systems (ICCES) 2016, 1–4. https://doi.org/10.1109/CESYS.2016.7889820.

Alattar, A.M., Alattar, O.M., 2004. Watermarking electronic text documents containing justified paragraphs and irregular line spacing. In: Electronic Imaging 2004, International Society for Optics and Photonics. pp. 685–695.

Banerjee, I., Bhattacharyya, S., Sanyal, G., 2012. Text Steganography through Quantum Approach. Springer, Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-31686-974, pp. 632–643.

Bassil, Y., 2012. A generation-based text steganography method using sql queries. Int. J. Comput. Appl. 57 (12), 27–31.

Bennett, K., 2004. Linguistic steganography: Survey, analysis, and robustness concerns for hiding information in text, Tech. rep., Purdue University, cERIAS TR 2004-13.

Bhattacharyya, S., Banerjee, I., Sanyal, G., 2011. Bengali steganography using calp with a novel bengali word processor. INFOCOMP J. Comput. Sci. 10 (4), 40–56.

Bin, Y., Xingming, S., Lingyun, X., Zhiqiang, R., Ruizhen, W., 2011. Steganography in ms excel document using text-rotation technique. Inf. Technol. J. 10 (4), 889–893.

Chang, C.-C., Lin, C.-J., 2011. Libsvm: a library for support vector machines. ACM Trans. Intell. Syst. Technol. 2 (3). https://doi.org/10.1145/1961189.1961199. 27 (1–27), pp. 27.

Changder, S., Debnath, N.C., 2009. A new approach for steganography in bengali text. J. Comp. Methods Sci. Eng. 9 (1,2S1), 111–122. URLhttp://dl.acm.org/citation.cfm?id=1608790.1608800.

Changder, S., Ghosh, D., Debnath, N.C., 2010. Linguistic approach for text steganography through indian text. In: 2010 2nd International Conference on Computer Technology and Development. pp. 318–322.https://doi.org/10.1109/ICCTD.2010.5645862.

Changder, S., Ghosh, D., Debnath, N.C., 2010. Lcs based text steganography through indian languages. In: 2010 3rd International Conference on Computer Science and Information Technology, Vol. 8. pp. 53–57.https://doi.org/10.1109/ICCSIT.2010.5563974.

Chen, Z., Huang, L., Yang, W., 2011. Detection of substitution-based linguistic steganography by relative frequency analysis. Digital Invest. 8 (1), 68–77. https://doi.org/10.1016/j.diin.2011.03.001. URLhttp://www.sciencedirect.com/science/article/pii/S1742287611000065.

Chen, Z., Huang, L., Miao, H., Yang, W., Meng, P., 2011. Steganalysis against substitution-based linguistic steganography based on context clusters. Comput. Electr. Eng. 37 (6), 1071–1081. https://doi.org/10.1016/j.compeleceng.2011.07.004. URLhttp://www.sciencedirect.com/science/article/pii/S0045790611001066.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn. 20 (3), 273–297. https://doi.org/10.1007/BF00994018.

Ekodeck, S.G.R., Ndoundam, R., 2016. PDF steganography based on chinese remainder theorem. J. Inf. Secur. Appl. 29, 1–15. https://doi.org/10.1016/j.jisa.2015.11.008. URLhttp://www.sciencedirect.com/science/article/pii/S221421261500068X.

Fawcett, T., 2006. An introduction to roc analysis. Pattern Recogn. Lett. 27 (8), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010.

Gutub, A., Fattani, M., 2007. A novel arabic text steganography method using letter points and extensions. In: Proceedings of World Academy of Science, Engineering and Technology, vol. 21. pp. 28–31.

Gutub, A., Elarian, Y., Awaideh, S., Alvi, A., 2008. Arabic text steganography using multiple diacritics. In: Proceedings of the 5th IEEE International Workshop on Signal Processing and its Applications (WoSPA08), University of Sharjah, Sharjah, UAE.

Hassan, A., Amin, M.R., Azad, A.K.A., Mohammed, N., 2016. Sentiment analysis on Bangla and romanized Bangla text using deep recurrent models. International Workshop on Computational Intelligence (IWCI) 2016, 51–56. https://doi.org/10.1109/IWCI.2016.7860338.

Huang, D., Yan, H., 2001. Interword distance changes represented by sine waves for watermarking text images. IEEE Trans. Circuits Syst. Video Technol. 11 (12), 1237–1245. https://doi.org/10.1109/76.974678.

Huang, H., Tan, J., Sun, X., Liu, L., 2009. Detection of Hidden Information in Webpage Based on Higher-Order Statistics. Springer, Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-04438-025, pp. 293–302.

Johnson, N.F., Duric, Z., Jajodia, S., 2001. Steganalysis. Springer, US, Boston, MA. https://doi.org/10.1007/978-1-4615-4375-63, pp. 47–76.

Kabetta, H., Dwiandiyanta, B.Y., et al., 2011. Information hiding in CSS: a secure scheme text-steganography using public key cryptosystem. Int. J. Cryptography Inf. Secur. 1, 13–22.

Khairullah, M., 2009. A novel text steganography system using font color of the invisible characters in microsoft word documents. In: 2009 Second International Conference on Computer and Electrical Engineering, vol. 1. pp. 482–484.https://doi.org/10.1109/ICCEE.2009.127.

Khairullah, M., 2011. A novel text steganography system in cricket match scorecard. Int. J. Comput. Appl. 21 (9), 43–47.

Khairullah, M., 2014. A novel text steganography system in financial statements. Int. J. Database Theory Appl. 7 (5), 123–132.

Khairullah, M., 2018. Steganography in bengali unicode text. SUST J. Sci. Technol. (In press)

Khan, S., 2014. Convergence in spelling, and spell-checker for romanized bangla in computers and mobile phones. In: 2014 International Conference on Informatics, Electronics Vision (ICIEV). pp. 1–5.https://doi.org/10.1109/ICIEV.2014.6850853.

Kim, Y.-W., Moon, K.-A., Oh, I.-S., 2003. A text watermarking algorithm based on word classification and inter-word space statistics. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition – volume 2, ICDAR '03, IEEE Computer Society, Washington, DC, USA. p. 775. http://dl.acm.org/citation.cfm?id=938980.939559.

Kumar, R., Malik, A., Singh, S., Kumar, B., Chand, S., 2016. A space based reversible high capacity text steganography scheme using font type and style. In: 2016 International Conference on Computing, Communication and Automation (ICCCA). pp. 1090–1094.https://doi.org/10.1109/CCAA.2016.7813878.

Li, L., Huang, L., Zhao, X., Yang, W., Chen, Z., 2008. A statistical attack on a kind of word-shift text-steganography. Int. Conf. Intell. Inf. Hiding Multimedia Signal Processing 2008, 1503–1507. https://doi.org/10.1109/IIH-MSP.2008.42.

Liu, Y., Wang, J., Wang, Z., Qu, Q., Yu, S., 2016. A Technique of High Embedding Rate Text Steganography Based on Whole Poetry of Song Dynasty. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-48671-017. 178–189.

Low, S.H., Maxemchuk, N.F., Brassil, J.T., O'Gorman, L., 1995. Document marking and identification using both line and word shifting. In: INFOCOM '95. Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Bringing Information to People. Proceedings, vol. 2. IEEE. pp. 853–860. https://doi.org/10.1109/INFCOM.1995.515956.

Luo, Y., Huang, Y., 2017. Text steganography with high embedding rate: Using recurrent neural networks to generate chinese classic poetry. In: Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security, IH&#38;MMSec '17, ACM, New York, NY, USA. pp. 99–104.https://doi.org/10.1145/3082031.3083240.

Mahato, S., Khan, D.A., Yadav, D.K., 2017. A modified approach to data hiding in microsoft word documents by change-tracking technique. J. King Saud Univ. – Comput. Inf. Sci. https://doi.org/10.1016/j.jksuci.2017.08.004 (In press),http://www.sciencedirect.com/science/article/pii/S1319157817300939.

Malik, A., Sikka, G., Verma, H.K., 2017. A high capacity text steganography scheme based on huffman compression and color coding. J. Inf. Optim. Sci. 38 (5), 647–664. https://doi.org/10.1080/02522667.2016.1197572.

Malik, A., Sikka, G., Verma, H.K., 2017. A high capacity text steganography scheme based on LZW compression and color coding. Eng. Sci. Technol. Int. J. 20 (1), 72–79. https://doi.org/10.1016/j.jestch.2016.06.005. URLhttp://www.sciencedirect.com/science/article/pii/S2215098616301331.

Mandal, K.K., Jana, A., Agarwal, V., 2014. A new approach of text steganography based on mathematical model of number system. In: 2014 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2014]. pp. 1737–1741.https://doi.org/10.1109/ICCPCT.2014.7054849.

Memon, J.A., Khowaja, K., Kazi, H., 2005. Evaluation of steganography for urdu/arabic text. J. Theor. Appl. Inf. Technol., 232–237

Meng, P., Hang, L., Chen, Z., Hu, Y., Yang, W., 2010. STBS: A Statistical Algorithm for Steganalysis of Translation-Based Steganography. Springer, Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-16435-416, pp. 208–220.

Morkel, T., Eloff, J.H., Olivier, M.S., 2005. An overview of image steganography. In: Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005).

Niimi, M., Minewaki, S., Noda, H., Kawaguchi, E., 2003. A framework of text-based steganography using SD-form semantics model, vol. 44.http://www.know.comp.kyutech.ac.jp/STEG03/STEG03-PAPERS/papers/12-Niimi.pdf.

OmicronLab. Avro keyboard and bangla spell checker!https://www.omicronlab.com/avro-keyboard.html, (accessed: 2017-03-17).

Qi, W., Guo, Z., 2015. Data hiding based on chinese text automatic proofread. In: 2015 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP). pp. 72–75.https://doi.org/10.1109/IIH-MSP.2015.35.

Rabah, K., 2004. Steganography-the art of hiding data. Inf. Technol. J. 3 (3), 245–269.

Rahman, M.S., Khalil, I., Yi, X., Dong, H., 2017. Highly imperceptible and reversible text steganography using invisible character based codeword. In: PACIS 2017 Proceedings., no. 230.http://aisel.aisnet.org/pacis2017/230.

Rose, R.H., Jamal, N., 2014. Feasibility of text visualization in text steganalysis. In: 13th International Conference on New Trends in Intelligent Software Methodology Tools, and Techniques, SoMeT 2014. IOS Press.

Salomon, D., 2003. Data Hiding in Text. Springer, New York, New York, NY. https://doi.org/10.1007/978-0-387-21707-911.

Satir, E., Isik, H., 2012. A compression-based text steganography method. J. Syst. Softw. 85 (10), 2385–2394. automated Software Evolution,http://www.sciencedirect.com/science/article/pii/S0164121212001379.

Satir, E., Isik, H., 2014. A huffman compression based text steganography method. Multimedia Tools Appl. 70 (3), 2085–2110. https://doi.org/10.1007/s11042-012-1223-9.

Sattar, M.A., Pathan, A.-M.K., Ali, M.A., 2004. Development of an optimal bangla keyboard layout based on character and fingering frequency. In: National Conference on Computer Processing of Bangla 2004, Independent University, Bangladesh. pp. 38–46.

Shi, S., Qi, Y., Huang, Y., 2016. An approach to text steganography based on search in internet. Int. Comput. Symp. (ICS) 2016, 227–232. https://doi.org/10.1109/ICS.2016.0052.

Shirali-Shahreza, M.H., Shirali-Shahreza, M., 2006. A new approach to persian/arabic text steganography, in: 5th IEEE/ACIS International Conference on Computer and Information Science and 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse (ICIS-COMSAR'06). p. 310–315.https://doi.org/10.1109/ICIS-COMSAR.2006.10.

Shniperov, A.N., Nikitina, K.A., 2016. A text steganography method based on markov chains. Autom. Control Comput. Sci. 50 (8), 802–808. https://doi.org/10.3103/S0146411616080174.

Silverman, B.W., 1986. Density estimation for statistics and data analysis, vol. 26. CRC Press.

Sui, X.G., Luo, H., Zhu, Z.l., 2006. A steganalysis method based on the distribution of first letters of words. In: 2006 International Conference on Intelligent Information Hiding and Multimedia. pp. 369–372.https://doi.org/10.1109/IIH-MSP.2006.265019.

Tian, Y., Shi, Y., Chen, X., Chen, W., 2011. Auc maximizing support vector machines with feature selection. In: Procedia Computer Science 4, 1691–1698, proceedings of the International Conference on Computational Science. ICCS 2011.https://doi.org/10.1016/j.procs.2011.04.183.http://www.sciencedirect.com/science/article/pii/S1877050911002419.

Tiwari, R.K., Sahoo, G., 2011. Microsoft excel file: a steganographic carrier file. Int. J. Digital Crime Forensics (IJDCF) 3 (1), 37–52.

Westfeld, A., Pfitzmann, A., 2000. Attacks on Steganographic Systems. Springer, Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/107197245, pp. 61–76.

Xiang, L., Sun, X., Luo, G., Xia, B., 2014. Linguistic steganalysis using the features derived from synonym frequency. Multimedia Tools Appl. 71 (3), 1893–1911. https://doi.org/10.1007/s11042-012-1313-8.

Xinmei, Meng, P., Ye, Y., Hang, L., 2010. Steganography in chinese text. In: 2010 International Conference on Computer Application and System Modeling (ICCASM 2010), vol. 8, pp. V8-651-V8-654.https://doi.org/10.1109/ICCASM.2010.5620373.

Yu, Z., Huang, L., Chen, Z., Li, L., Zhao, X., Zhu, Y., 2009. Steganalysis of synonym-substitution based natural language watermarking. Int. J. Multimedia Ubiquitous Eng. 4 (2), 21–34.

Zhao, X., Huang, L., Li, L., Yang, W., Chen, Z., Yu, Z., 2009. Steganalysis on character substitution using support vector machine. Second International Workshop on Knowledge Discovery and Data Mining 2009, 84–88. https://doi.org/10.1109/WKDD.2009.105.