HIGH-CAPACITY TEXT STEGANOGRAPHY MODEL BASED ON TWO-LETTER WORD AND DUAL COMPRESSION TECHNIQUE

SALWA SHAKIR BAAWI ALBU RGHAF

UNIVERSITI KEBANGSAAN MALAYSIA

HIGH-CAPACITY TEXT STEGANOGRAPHY MODEL BASED ON TWO-LETTER WORD AND DUAL COMPRESSION TECHNIQUE

SALWA SHAKIR BAAWI ALBU RGHAF

THESIS SUBMITTED IN FULFILMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF INFORMATION SCIENCE AND TECHNOLOGY
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI

2018

MODEL STEGANOGRAFI TEKS BERKAPASITI TINGGI BERASASKAN PERKATAAN DUA HURUF DAN TEKNIK PEMBANGUNAN DUAL

SALWA SHAKIR BAAWI ALBU RGHAF

TESIS YANG DIKEMUKAKAN UNTUK MEMPEROLEHI
IJAZAH DOKTOR FALSAFAH

FAKULTI TEKNOLOGI DAN SAINS MAKLUMAT
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI

2018

DECLARATION

I hereby declare that the work in this thesis is my own except for quotations and summaries which have been duly acknowledged.

20 March 2019

SALWA SHAKIR BAAWI
ALBU RGHAF
P78424

ACKNOWLEDGMENT

In the name of Allah, most gracious and most merciful.

I praise the Almighty Allah for all his blessings and for giving me patience and good health throughout my Ph.D. research.

This research project would not have been completed without the support of many people. I wish to express my sincerest appreciation and gratitude to my supervisors Dr. Mohd Rosmadi Mokhtar and Dr. Rossilawati Sulaiman for their generous and continuous support, their excellent advice and guidance throughout this research, and the time, cooperation, and effort they have dedicated to the supervision of this work. I thank them for their valuable suggestions, corrections, and unlimited encouragement throughout the different stages of this research.

I also thank the staff of the Faculty of Information Science and Technology of the National University Malaysia (UKM) for their cooperation throughout the years.

I wish to thank all my friends who have supported and inspired me to reach my goals. I also thank all postgraduate students in the Faculty of Information Science and Technology for their help and friendship as well as for creating a pleasant working environment during my years at UKM.

I especially thank my beloved family, who has been my strongest source of motivation, inspiration, undying love, support, and encouragement throughout my years of study. I am very grateful to my brothers and sisters for all the sacrifices that they have made on my behalf. Each one of you has given me inspiration and motivation. My God, SWT, blesses us all.

I am also very grateful to my Ph.D. studentship sponsored by the Iraqi government as represented by the Ministry of Higher Education and Scientific Research, which made this study possible and helped create a positive educational environment where I can work. I also thank the employees of the Iraqi Cultural Attaché in Malaysia for all their help throughout my Ph.D. studies.

ABSTRACT

Steganography is a method for hiding information in other media to obtain a secure communication medium and protect data during a transmission process. Specifically, text steganography concentrates on using texts to conceal secret messages and utilizes data stored within text documents. However, text steganography is a challenging type of information hiding given the lack of data redundancy in a text file relative to other carrier files. Simultaneously, a trade-off occurs between steganographic capacity and stego text quality. Thus, increasing steganographic capacity and enhancing stego text quality are challenging tasks. This study aims to improve the data-hiding rate in accordance with the related requirements. To achieve this goal, four objectives are addressed. The first objective is to identify suitable properties in English words which provide considerable redundant data in a text file and achieve high transparency. The second objective is to enhance data-hiding capacity technique whilst maintaining the quality of the stego text file. The third objective is to propose a new high-capacity text steganography model (HICATS) that simultaneously adjusts the requirements on the basis of two proposed techniques based on the two-letter word (TLW) and dual compression techniques. The final objective is to evaluate the proposed model using performance evaluation measurements of transparency and hiding capacity; these measurements focus on the similarity of a cover and a stego file, the cover file size before and after an embedding process, a hiding capacity ratio and a space-saving ratio. This study adopts a design science research methodology which comprises six phases, namely, identifying the problem and research motivation, defining the objectives of the solution, designing, and development, demonstration, evaluation and communication. Empirical findings show that the proposed two-letter word approach satisfies perceptual transparency in the cover file, in which an average similarity string measurement value is approximately 95% between the cover and the stego text files. Results also reveal that dual compression improves the similarity string measurement of the enhanced TLW (En-TLW) from 97% to 99% without affecting the stego text quality. The En-TLW technique doubles the hiding capacity whilst maintaining the similarity metric considering its capability to hide four bits of a secret message in each TLW. The dual-compression technique of Lempel–Ziv–Welch, and Huffman coding successfully reduces the secret message by 64% before the embedding process. Experimental outcomes from HICATS model on the cover file size before and after the embedding process show a significant reduction from 51.27% using the TLW technique to only 12.08% using the model. The hiding capacity ratio increases from 2.64% using the TLW technique to 3.71% using the HICATS model. In addition, the model improves the space-saving ratio from 21.90% to 28.28%. These results indicate that the proposed HICATS model and the two techniques notably enhance the hiding capacity of text steganography with reasonable transparency and similarity values. In the future, HICATS can be used in another language with different cases of cover medium files, such as a word, Microsoft PowerPoint Open XML Presentation File and Portable Document Format file.

ABSTRAK

Steganografi adalah kaedah untuk menyembunyi maklumat dalam media lain bagi mendapatkan medium komunikasi serta data yang selamat semasa proses penghantaran maklumat. Steganografi teks menumpu kepada penggunaan teks untuk menyembunyi mesej rahsia dan mengguna data yang disimpan dalam dokumen teks tersebut. Walau bagaimanapun, steganografi teks merupakan salah satu teknik penyembunyian maklumat yang paling mencabar kerana kurangannya pertindihan data dalam fail pembawa berjenis teks, berbanding dengan jenis fail pembawa yang lain. Pada masa yang sama, keseimbangan juga perlu diberi perhatian di antara kapasiti steganografi dan kualiti teks stego. Hal ini juga merupakan suatu cabaran, iaitu untuk meningkat kapasiti steganografi dan pada masa yang sama meningkat kualiti teks stego. Oleh itu, kajian ini bertujuan untuk meningkat kadar persembunyian data mengikut keperluan yang berkaitan. Empat objektif dikenalpasti bagi mencapai matlamat ini. Objektif pertama adalah untuk mengenal pasti ciri-ciri yang sesuai dalam perkataan Bahasa Inggeris yang menyediakan pertindihan data yang mencukupi dalam fail teks dan mencapai ketelusan yang tinggi. Objektif kedua ialah untuk meningkat teknik kapasiti penyembunyian data dan pada masa yang sama mengekal kualiti fail teks stego. Objektif ketiga ialah mencadang model baru bagi steganografi teks berkapasiti tinggi (HICATS) yang dapat disesuaikan dengan keperluan, berdasarkan dua teknik yang dicadangkan iaitu pendekatan perkataan dua huruf (TLW) dan teknik pemampatan teks. Objektif akhir adalah untuk menilai model yang dicadang menggunakan ukuran penilaian prestasi ketelusan dan ruang simpanan. Penilaian ini menumpu kepada persamaan fail penutup dan fail stego, saiz fail penutup sebelum dan selepas proses pembenaman mesej rahsia, nisbah kapasiti penyembunyian data, serta nisbah ruang simpanan. Penyelidikan ini mengguna metodologi penyelidikan sains reka bentuk (DSRM), yang terdiri daripada enam fasa iaitu mengenal pasti masalah dan motivasi penyelidikan, menentu objektif penyelesaian, perancangan dan pembangunan, demonstrasi, penilaian, dan komunikasi. Hasil penemuan secara empirikal menunjukkan bahawa pendekatan perkataan dua huruf yang dicadang memenuhi ketelusan perseptual dalam fail penutup, di mana nilai ukuran persamaan adalah kira-kira 95% di antara fail teks penutup dan fail teks stego. Hasil keputusan juga menunjukkan bahawa dwi pemampatan meningkat ukuran kesamaan bagi TLW dipertingkat (En-TLW) daripada 97% kepada 99% tanpa menjejaskan kualiti teks stego. Teknik En-TLW menggandakan kapasiti penyembunyian data dan pada masa yang sama mengekal metrik kesamaan kerana kemampuannya menyembunyi empat bit mesej rahsia dalam setiap TLW. Teknik dwi mampatan Lempel-Ziv-Welch dan Pengekodan Huffman berjaya mengurang saiz mesej rahsia sebanyak 64% sebelum proses pembenaman. Hasil eksperimen dari model HICATS pada saiz fail penutup sebelum dan selepas proses pembenaman menunjukkan pengurangan ketara, iaitu dari 51.27% dengan teknik TLW kepada 12.08%, dengan menggunakan model. Sementara itu, dengan mengguna teknik TLW, nisbah kapasiti penyembunyian data meningkat daripada 2.64% kepada 3.71%, dengan menggunakan model HICATS. Model ini juga meningkatkan nisbah simpanan ruang dari 21.90% kepada 28.28%. Keputusan ini menunjukkan bahawa model HICATS yang dicadang dengan kedua teknik ini dapat meningkatkan kemampuan penyembunyian data dalam steganografi teks, dengan nilai ketelusan dan kesamaan yang munasabah. Pada masa

akan datang, HICATS boleh diguna dalam bahasa lain dengan fail media yang beza seperti, Fail Pembentangan XML Terbuka, Microsoft PowerPoint dan fail Format Dokumen Portable.

TABLE OF CONTENTS

CHAPTER VI CONCLUSIONS AND FUTURE WORK

APPENDICES

LIST OF TABLES

LIST OF ILLUSTRATIONS

Figure No. **Page**

LIST OF ABBREVIATIONS

DSRM	Design Science Research Methodology
OCR	Optical Character Recognition
.docx	A document created by Microsoft Word
.pdf	Portable Document Format.
.pptx	Microsoft PowerPoint Open XML Presentation File
.txt	Text File Format
HVS	Human Visual System
E	Embedding Process
D	Extracting Process
M	Secret Message
C	Cover File
S	Stego File
K	Stego Key
SPAM	Spamming Undesired Electronic Messages
.wav	Waveform Audio File Format
MIDI	Musical Instrument Digital Interface
AVI	Audio Video Interleave
DCT	Discrete Cosine Transform
MP4	Digital Container format
MPEG	Moving Picture Exports Group
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
TCP	Transmission Control Protocol
IP	Internet Protocol
OSI	Open Systems Interconnection Model
EOF	End Of File

HTML	Hypertext Markup Language
HR	Hiding Capacity Ratio
SSR	Saving Space Ratio
SIR	Size Increasing Ratio
UniSpaCh	Unicode Space Character
DASH	Space Character (White Space)
ASCII	American Standard Code for Information Interchange
LZW	Lempel-Ziv-Welch
RNA	Ribonucleic Acid
MRNA	Messenger Ribonucleic Acid
MRLE	Modified Run Length Encoding
RLE	Run Length Encoding
UTF-16	16-bit Unicode Transformation Format
UTF-32	32-bit Unicode Transformation Format
DBCS	Double-Byte Character Sets
MBCS	Multi-Byte Character Sets
ACW	Adaptive Character Word
UDC	User-Defined Code
HICATS	High Capacity Text Steganography
TLW	Two-Letter Word
ZWJ	Zero Width Joiner Unicode character
ZWNJ	Zero Width non-Joiner Unicode character
En-TLW	Enhanced of Two-Letter Word

CHAPTER I

INTRODUCTION

1.1 RESEARCH OVERVIEW

With the excessive usage of the Internet and the emergence of various computing technologies, accessing and exchanging multimedia files, including audio, video, text, and image, among individuals and groups have become very convenient. Nevertheless, the distribution of such a vast amount of data over the Internet makes them vulnerable to attacks. Therefore, protecting sensitive data has become an important issue that requires an immediate solution. Digital information security has also become an important field of research (Malik et al. 2017), and various methods for ensuring data security have been proposed. Many studies are also underway to enhance Internet safety.

Information security covers a wide range of topics that fall under two significant disciplines, namely, cryptography and data hiding, as depicted in Figure 1.1. Both of these disciplines aim to ensure the security of transferring data over public networks. Cryptography hides the contents of messages by rearranging them. Enciphering the traffic cannot sufficiently protect data because hackers are generally able to bypass encrypted communication. Although cryptography utilizes unintelligible text, third parties are always aware of the connection taking place, thereby posing a significant disadvantage for the utilization of cryptography in ensuring data security (Agarwal 2013).

Data hiding is another popular method for ensuring data security that can be divided into two branches, namely, steganography and watermarking (Odeh et al. 2014). Steganography overcomes the limitation of cryptography by hiding one message into another, thereby preventing the detection of this message by parties other than its intended recipient. Steganography, which is considered the most common sub-discipline of data hiding (Rafat & Sher 2013; Banerjee et al. 2012), can be defined as the science or art of invisible communications. Therefore, a steganographic system hides secret data from the public to prevent arousing the suspicion of eavesdroppers. Two parameters, namely, capacity and transparency, represent the two essential aspects of the system. However, these two aspects are at odds with each other.

As its name suggests, watermarking assert intellectual property rights by concealing information (Mir & Hussain 2011). This technique can be applied in various applications, such as source tracking, copyright protection, content augmentation, and fingerprint applications.

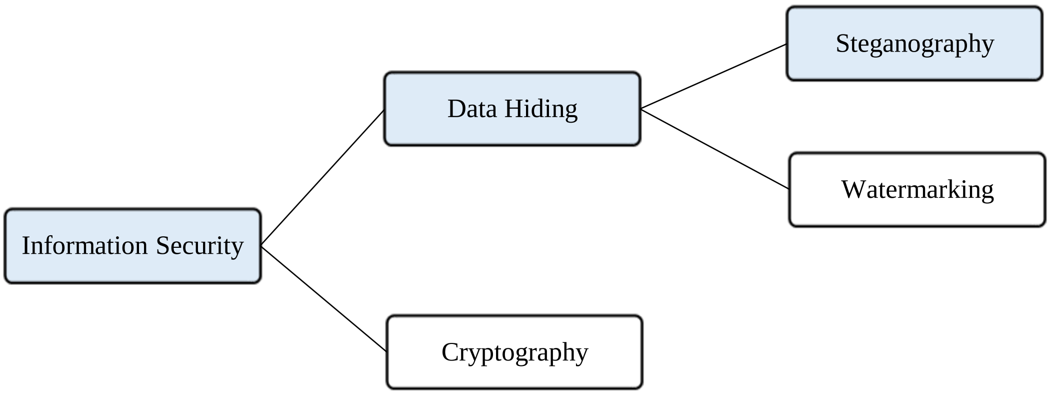


Figure 1.1 Disciplines of information security

As shown in Figure 1.1, this study focuses on steganography, which is classified under data hiding in the field of information security. This chapter introduces the research, including the study background in Section 1.2, the problem statement in Section 1.3, the research questions in Section 1.4, the research objectives in Section 1.5, the research motivation and scope in Section 1.6, and the thesis outlined in Section 1.7.

1.2 RESEARCH BACKGROUND

The wide availability of computers and the Internet has also increased the availability of information. However, such wide availability has jeopardized the security of information that must be kept proprietary or secret (Djebbar et al. 2012). Accordingly, several approaches for protecting and hiding confidential data have been proposed, with steganography being one of the most popular subdisciplines of information hiding (Hiary et al. 2016; Banerjee et al. 2012). As briefly mentioned in the previous section, this technique seeks to hide one message within another to prevent the former from being detected by people other than its intended recipient (Malik et al. 2017).

Steganography has become a popular research focus in the past years. The application of this technique can be traced back to the Greek civilization when messengers tattoo secret messages on their shaven heads and then wait for their hair to grow and cover their tattoos (Vidhya & Paul 2015). Other ancient steganography methods include using wax tablets to hide messages (Alshayeji et al. 2017).

Some people have also used invisible ink to write hidden messages during World War II. The contents of this message are not revealed to anyone until it is exposed to heat. The Germans have also developed the microdot technique, which uses photographs the size of printed dots that can only be deciphered by Germans. These microdots were included in letters and remained undetected during their delivery due to the tiny size of the pictures. This technique has also been applied to hide messages in human body parts, such as nostrils, ears, or fingernails (Chang & Clark 2014).

Modern steganography formally began in 1985 with the advent of personal computers. Therefore, this term almost exclusively refers to electronic than physical media, such as those used in ancient times. Das et al. (2011) examined modern steganography that uses the features of the transferred media such as text (Li et al. 2015; Mohamed 2014; Souvik Roy & Venkateswaran 2013; Roslan et al. 2011), image (Jain et al. 2017; Atawneh et al. 2013), audio (Ali et al. 2017; Patil & Pawar 2016), and video (Mazurczyk et al. 2016; Sadek et al. 2015) to conceal the message.

Steganography that uses text as its media remains one of the most sensitive data encoding techniques due to its ability to apply obvious changes in the syntax and semantics of the cover media. Therefore, researchers have specifically examined how messages are hidden in cover texts to preserve both their meaning and syntax (Kumar et al. 2015). One notable feature of text media is their structure, which is more visually apparent in terms of syntax and grammar compared with other media, such as images.

The text structure of messages can also be altered, thereby allowing these messages to be embedded without considerably changing their output. While alterations in audio or image media may be undetectable, the inclusion of a certain letter or punctuation mark in text files may be easily caught by the reader. Despite these disadvantages, text steganography is still mainly used because of its very small file size and easy application (Agarwal 2013; Souvik Roy & Venkateswaran 2013).

Capacity, transparency, and robustness are the essential requirements of steganography (Aman et al. 2017; Ardakani et al. 2015; Odeh et al. 2013a). Capacity determines the quantity of data that can be embedded in a cover file. Transparency measures the innocuous look of a stego medium to an eavesdropper and eliminates the impact of suspicious behavior. This property can also avert the attention of intruders by presenting an image as an ordinary text, thereby keeping its secret information secure (Aman et al. 2017). Robustness refers to the ability to resist the alteration of a secret message. Therefore, steganographic techniques primarily aim to achieve the high capacity and low distortion of the cover file (Lee & Chen 2013).

According to (Ardakani et al. 2015), the three requirements above are mostly related to one another, that is, increasing any of these requirements will influence the other two. Therefore, achieving all these requirements at the same time for a single scheme is a challenging task. Transparency is measured based on the manipulation or alteration of the cover media during the embedding of the secret message. Increasing the hiding capacity is a primary requirement in steganography. However, such a requirement cannot be fulfilled due to the cost of transparency. Robustness is especially important for watermarking and copyrighting and is most often used in digital watermarking. Given that digital copies of data are often kept the same as the original, digital watermarking may be classified as a passive protection technique that marks yet does not degrade or control the access to data (Alginahi et al. 2014).

Text steganography techniques can be classified into three categories according to their application domain. Many works (Tutuncu & Hassan 2015; Roslan et al. 2014; Odeh et al. 2013; Amirtharaj & Rayappan 2013) have classified these techniques into format-based, linguistic-based, and random and statistical generation methods, while Mansor et al. (2017) and Wai and Khine (2011) classified these techniques into format-based and linguistic methods. Each of these categories has unique advantages and disadvantages, which are presented in detail in Section 2.7. The linguistic methods adopted in text steganography have limited hiding capacity, while the format-based method, despite, its ability to address the main drawback (i.e., capacity) of linguistic methods, has low robustness. Recent studies have adopted data compression techniques to increase hiding capacity, which is one of the primary requirements of steganography techniques along with preserving the transparency of the cover file such as (Aman et al. 2017; Kumar, Malik, et al. 2016; Kumar et al. 2014; Satir & Isik 2014).

1.3 PROBLEM STATEMENT

Text-based steganography uses written texts to conceal secret messages (Din et al. 2017), through many formats of text files that are applied in storing data, such as .txt and .docx. Nonetheless, text steganography is regarded as a challenging type for embedding data, because text files are limited by the lack of redundant data compared with other cover media (Aman et al. 2017; Xiang et al. 2017; Ekodeck & Ndoundam 2016; Vidhya & Paul 2015; Majumder & Changder 2013; Odeh & Elleithy 2012; Garg 2011). Researchers have attempted to improve performance metrics; thus, considerable data can be embedded using minimal time and storage overhead. Such an improvement can be achieved by the utilization of text files as cover media, given that it uses minimal storage space and can be easily applied (Malik et al. 2017; Kumara et al. 2016; Tutuncu & Hassan 2015).

Researchers aim to explore the performance of the steganography system in terms of hiding capacity and transparency (Aman et al. 2017; Xiang et al. 2017;

Ekodeck & Ndoundam 2016; Vidhya & Paul 2015). Many researchers have attempted to address security issues in which the transparency factor is considered a security factor (Aman et al. 2017; Obeidat 2017; Abbasi et al. 2015; Ardakani et al. 2015; Mohamed 2014), and compromises the amount of data that a user wants to hide inside a digital medium (Alotaibi & Elrefaei 2018; Malik et al. 2017; Obeidat 2017; Budiman & Novamizanti 2015; Chang & Clark 2014; Satir & Isik 2014; Ali & Saad 2013; Bhaya et al. 2013; Por et al. 2012; Satir & Isik 2012; Mir & Hussain 2011; Shu et al. 2011). However, most of these attempts suffer from limitations such as in payload capacity, security or transparency (discussed in Chapter 2, Section 2.7.5).

Several text steganography schemes adopt three categories in embedding and extraction processes: format-, linguistic- and random and statistical generation-based categories. The motivation behind adopting these categories is to enhance the performance of text steganography schemes by increasing transparency and hiding capacity. Although these categories suffer from limitations, as discussed in Chapter 2, Sections 2.7.4 and 2.7.5, they are still used by researchers in text steganography schemes.

Many techniques have been applied to text steganography in various languages, such as Arabic, English, and Persian. Researchers have adopted specific properties according to each language to introduce redundancy, which is used in the embedding process. Arabic texts have many properties (characteristics), such as the connected letters (Obeidat 2017). This is a high-capacity method that does not change the size of the text. Furthermore, the isolated letters used by the researcher Mohamed (2014), provide a low in capacity, but the technique is a relatively secure algorithm and resist traditional attacking methods, such as known cover, known message, chosen stego, chosen message and known stego (Böhme 2010). Roslan et al. (2014), proposed a new algorithm for solving the hiding capacity issue in their previous work (Roslan et al. 2011). This new algorithm utilizes the main primitive structure of Arabic letters, such as dots, sharp edges and typographical proportions (the calligraphy proportion used for writing Arabic calligraphy), for embedding secret bits. It uses randomization in placing secret bits to make the algorithm secure. A technique presented by Odeh et al. (2012), divides Arabic letters into four groups based on the number of points in each letter (such as letters without points, with one-point and with two-points and multipoint letters). The results of these studies enhanced each method’s capacity and robustness.

Meanwhile, some properties are also used in English text to embed data. For example, some techniques use only one letter, such as the technique proposed in (Ali & Saad 2013), This technique uses capital and small letters simultaneously. This technique achieves substantial data while keeping the exact meaning of the text. Bhaya et al. (2013), introduced a method that utilizes capital letters as carriers in English text. The benefits of this method are that it has a large capacity and good perceptual transparency. However, the stego document in the study increased by approximately 0.766% of its original size. Dulera et al. (2011), proposed three approaches for embedding data in English text. These approaches utilize characters with round shapes or straight vertical lines as distinctive elements and adopts quadruple categorization of English letters as the basis. These approaches achieve considerable transparency, but the quality of the cover file is noticeably low to the reader. Other techniques, such as the technique proposed in (Shivani et al. 2015). use one word for embedding data. This method focuses on combining the abbreviation method with a zero-distortion technique to overcome the limitation of the abbreviation method. However, only a small amount of data can be hidden in a text file at a time. Chang and Clark (2014), proposed a technique that depends on exploiting synonyms by replacing selected words with their synonyms. This technique provides a reasonable level of transparency but has low data hiding capacity.

So far, the discussion has covered Arabic and English letters. Arabic letters have four main positions, namely, the letter Ain ‘ع’, at the initial position ‘عـ’, in the middle position ‘ـعـ’, at the last position ‘ـع’, and as an isolated letter ‘ع’. The position property of the Arabic letter can provide increased redundancy or capacity to hide secret data (Alotaibi & Elrefaei 2015; Odeh & Elleithy 2012b, 2012c). However, this case does not follow in English letters. The position of one letter in an English word does not affect its form. The English language has many different words comprising letters. The shortest words in English, which comprise only one letter, are ‘I’ and ‘A’. This study attempts to use two-letter words in the English language, such as “on”, “of”, “in” and “or” including any word with two letters and above, such as “**an**d”, “**we**re”, “**or**der”.

Two-letter words are adopted because doing so may provide considerable data redundancy compared with techniques from previous studies in English text steganography.

Several compression techniques have been proposed in the literature, and these techniques can be used with cover text media. For example, (Satir & Isik2014; Lee & Chen 2013; Majumder & Changder 2013; Satir & Isik 2012) proposed data-hiding strategies that offer excellent hiding capacity without changing the stego file despite using a low-quality stego file. Lossless techniques, such as Lempel–Ziv–Welch (LZW), Huffman coding, Burrows-Wheeler, move-to-front, arithmetic coding and run-length encoding (RLE), are examples of such compression techniques. Each of them provides different compression ratios and reconstructed file qualities. Thus, a technique that can enhance data-hiding capacity while maintaining the quality of the stego text files must be identified. In particular, the size of hidden data and the level of suspicion are crucial problems in steganography; these problems only increase the size of stego files (Ekodeck & Ndoundam 2016; Kumar et al. 2016; Mahato et al. 2014; Stojanov et al. 2014; Bhaya et al. 2013; Souvik Roy & Venkateswaran 2013; Por et al. 2012).

Another issue in text steganography is related to the simultaneous adjustment of all the steganography requirements; these issues are data-hiding capacity, transparency, and quality of the stego text file. The focus is given to preserving the trade-off between the requirements by increasing the hiding capacity with enhancing transparency or maintaining the quality of the stego file (reducing the size of hidden data). In this case, a model that achieves useful text steganography must be developed. This model should provide increased data-hiding capacity, simultaneously reduce the size of hidden data and the level of suspicion and handle redundant data. Therefore, this model is expected to solve redundant data issues related to text files.

1.4 RESEARCH QUESTIONS

The study attempts to answer the following research questions:

RQ1: What are the properties in the English texts which provide considerable redundant data in a text file and can achieve high transparency?

RQ2: Which technique can enhance the data-hiding capacity and maintain the quality of stego text files?

RQ3: What are the chances of simultaneously obtaining a high hiding capacity, ensuring the similarity between the cover and stego files and maintaining the cover file size in a particular steganography model?

RQ4: How can the proposed model be evaluated by using evaluation measures?

1.5 RESEARCH OBJECTIVES

The ultimate goal of this research is to propose a new, efficient text steganography technique for data safety and protection applications. The primary objectives of this thesis are summarized below:

RO1: To identify the suitable properties in English words which provide considerable redundant data in a text file and can achieve high transparency.

RO2: To enhance the data-hiding capacity technique whilst maintaining the quality of stego text files using compression techniques.

RO3: To propose a new high-capacity text steganography model that simultaneously adjusts the requirements on the basis of tow-letter word and dual compression techniques.

RO4: To evaluate the proposed model by using performance evaluation measurements of transparency and hiding capacity.

1.6 RESEARCH MOTIVATION AND SCOPE

The extensive usage of modern communication highlights the need to protect sensitive information. Network security has become increasingly relevant these days due to a large amount of data being exchanged over networks. Specifically, confidentiality and data security must be maintained to safeguard data from unwanted access. Steganography refers to the science and art of concealing information in ways that ensure minimal detection by potential attackers. This method may be employed in conjunction with existing communication methods for exchanging confidential information.

The research interest in steganography has grown in recent years for two main reasons (Osman et al. 2016). First, various governments have restricted the availability of encryption services, thereby motivating people to study methods by which private messages can be embedded in seemingly innocuous cover messages. Second, publishing and broadcasting industries have become interested in information hiding techniques to hide encrypted copyright marks and serial numbers in audio recordings, digital films, multimedia products, and books. Sumathi et al. (2013) argued that steganography must be examined further to develop the information hiding domain. Moreover, related studies have mostly focused on hiding data in image, audio, and video files, while relatively few studies have examined hiding information in text files.

The scope of this research is to ensure better transparency and high capacity when transmitting messages. A wide selection of steganographic algorithms that depend on the chosen media may be used to fulfill such a requirement. This study applies a text steganographic algorithm because text consumes limited memory, can be communicated merely, and is widely available all over the Internet in a digital form, unlike other mediums.

Data compression techniques can be classified into lossless compression and lossy compression techniques. The proposed model in this research is based on the lossless compression algorithm and uses the optional stego key. This algorithm hides secret data in a text document by applying an insertion technique that uses invisible spacing symbols. In order to establish the portability and operability of this model, C# programming is utilized for the system code. Therefore, those platforms that are incompatible with C# programming are unable to execute the proposed algorithm.

The limitations of this study include only embedding a secret text message into a cover file with the .txt format. The hiding technique is tested only on two-letter words in English scripts, only applies a pure or a secret stego key and transmits the stego key through different methods depending on the sender's and receiver's sites. The information of the Huffman tree is transferred from the sender's site to the receiver's site.

1.7 THESIS OUTLINE

The organization of this thesis comprises six chapters. Chapter 1 presents general information about this study, including its background, problem, questions, objectives, motivation and scope, methodology, and outline. Chapter 2 reviews the literature related to the terminology, classification, categories, and requirements of steganography. The attack types of steganography and methods of steganalysis present in Chapter 2. This chapter also discusses text steganography and its different classes. The drawbacks of different classes of text steganography techniques are also analyzed. Also, this chapter proposes a new classification for text steganography techniques. The findings can facilitate the formulation of the questions and objectives of this work and provide a foundation for the proposed solution.

Chapter 3 presents the adopted methodology for achieving the proposed research objectives. Each stage of the problem is identified up to the communication stage. The central evaluation criteria which are transparency, similarity sting measure, hiding capacity, increasing file size, and saving space measures are explained. The datasets used in the selection of text files for the tests are also described. Chapter 4 fully presenting the conceptual framework and the adopted methods. Also, it describes the proposed model for text steganography and its four phases, namely, encoding, hiding, extracting, and decoding. The sub-processes within each phase are also discussed. The proposed model for this work is also developed and demonstrated.

Chapter 5 presents the experiments and model evaluation. Four experiments on transparency, similarity sting measure, hiding capacity, increasing file size, and saving space measures are conducted. The results that are obtained from comparing the hiding capacity, the similarity of the cover file, the saving space ratio, and the size of the cover file before and after the embedding process are also presented in this chapter.

Chapter 6 concludes the thesis and presents the findings in relation to the research objectives. The contributions of this work are also highlighted. Some suggestions for further research are also given.

CHAPTER II

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter reviews the literature on steganography with the purpose of finding gaps from which a new text steganography method can be developed. Sections 2.2 and 2.3 present an overview of steganography starting from its definitions and related terminologies. Section 2.4 discusses various steganography classifications, Section 2.5 describes the requirements of the steganography system, Section 2.6 investigates steganalysis, and Section 2.7 structures text steganography into three clusters, namely, format, random and statistical generation, and linguistic. It also examines various steganography techniques with the aim of justifying the text steganography method proposed in this study. Then proposes a new classification on text steganography techniques. Section 2.7.6 examines various related concepts with the aim of justifying the conceptual framework proposed in this study. Section 2.9 summarizes the chapter.

2.2 DEFINITIONS OF STEGANOGRAPHY

The etymology of the word “steganography” can be traced to the Greek word for “hidden” and “writing.” Holistically, this word refers to a wide variety of methods that allow secret communications by hiding the contents of a message to all but its intended recipient (Sumathi et al. 2013). Some researchers have also defined this term as the “art and science” of communicating in such a way to conceal the existence of communication (Sangita Roy & Manasmita 2011).

This form of hidden communication may take place via audio, video, image, text, or protocol. Modern steganographic systems utilize various forms of multimedia, including image, audio, video, and text. These media are referred to as “cover media.” Those individuals who often transfer digital pictures via email or share them freely over the Internet aim for confidentiality to safeguard the real content of their messages (Cole 2003).

In its simplest definition, steganography refers to the act of hiding information within other pieces of information. In contrast to other data security techniques such as cryptography (protecting information via entirely altered media), steganography does not aim to change the structure of the secret message but rather conceal the intended message within a cover media (may also be called stego-media) as previously discussed. By nature, steganographic methods suffer from the difficulty in recovering data without applying a procedure known only to the intended recipients of the message. This study seeks to investigate digital steganography and its techniques, particularly text steganography.

2.3 TERMINOLOGY OF STEGANOGRAPHY

To further understand the concept of steganography, Simmons (1984), introduced a story where two prisoners, Alice and Bob attempt to exchange secret messages without being noticed by their warden, Wendy. As soon as she notices any interaction between her two prisoners, Wendy immediately terminates the communication between Alice and Bob. This two-prisoner situation is commonly used to illustrate the usage of cryptography in the context of arbitrary countries. Specifically, two countries may seek to exchange information without being intercepted by another country (i.e., the “warden”) and thereby resort to applying steganographic methods in their everyday conversations to dispel any suspicion.

Figure 2.1 presents an information-theoretic framework for steganography that has been previously applied in (Cox et al. 2008; Cachin 2004; Katzenbeisser & Petitolas 2000). This framework comprises two processes, namely, embedding (E) and extraction (D).

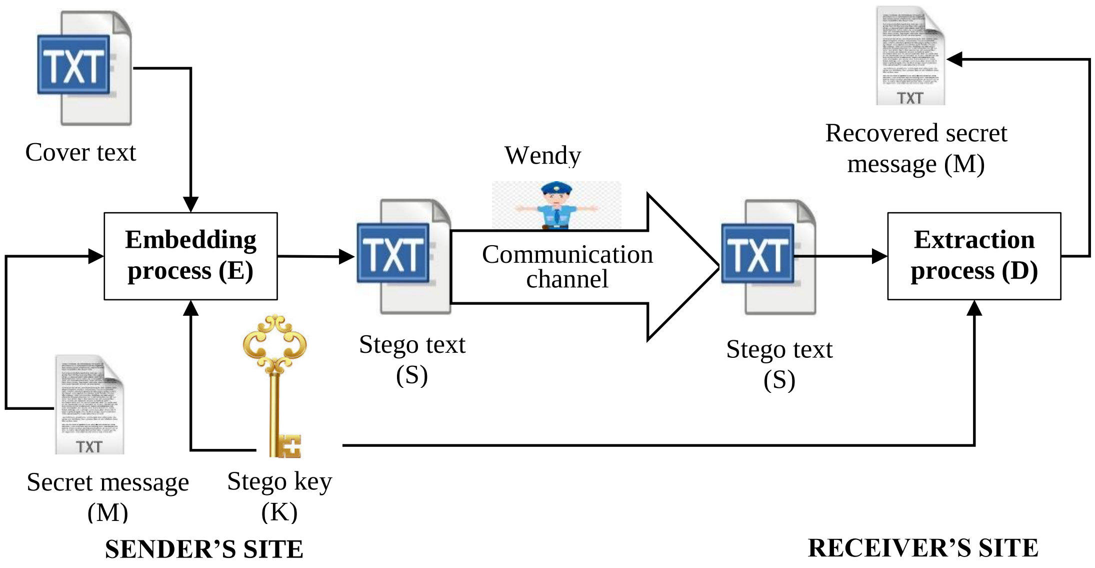


Figure 2.1 Information-theoretic framework for steganography

The left side of Figure 2.1 shows the sender, who embeds the secret message into the cover text file (C) and transmits the modified "stego text" (S) to the receiver (Bob) on the right. Wendy, who aims to intercept the messages between the sender and receiver, is placed at the bottom of the figure. She continuously attempts to catch the message or detect an information exchange between these parties. This model is structured in a way that only the intended recipient can extract the message due to the shared secret between the transmitter and receiver. This shared secret can be an algorithm for extraction or particular parameters of the algorithm and can be illustrated as a "key."

A steganographic system can be mathematically defined as a quintuple $\wp = (C, M, K, D_K, E_K)$, where C represents the set of cover media utilized during the communication, M represents the set of all hidable messages that must be transmitted by using covers, and K represents a stego key drawn from a set key. A steganographic system is formed by two functions, namely, the embedding function $E_K: C \times M \times K \rightarrow S$ and the extraction function $D_K(E_K(C, M, K), K) = M$ (Dumitrescu et al. 2017).

The secret message(M), which contains sensitive data, requires a certain type of camouflage. The cover media (C) refers to the media that hide the message inside their bodies. The block (E) represents the process of generating a stego file (S) by embedding the secret information within the cover that is encrypted by using a stego key(K). The stego file (S) is a combination of the cover medium into which the secret message is embedded. The block (D) represents the process of extracting the embedded information from the stego file. The stego key (K) represents the element that coordinates the procedures of inserting the message inside the cover and extracting this same message from the stego file.

Holistically, steganography refers to the activity of hiding confidential information within other media (C) to produce a stego file (S) by using a stego key (K) at end of the sender. Steganography may also be utilized by the recipient to extract the hidden message(M). Figure 2.1 illustrates the basic concepts of the steganographic system in its entirety.

2.4 CLASSIFICATION OF STEGANOGRAPHY

This section presents the different classifications of steganography. This classification may be based on the cover medium type used, the key used for hiding and extracting hidden messages, or the hiding technique (Abbas & Hamza 2014; Odeh et al. 2012) as shown in Figure 2.2.

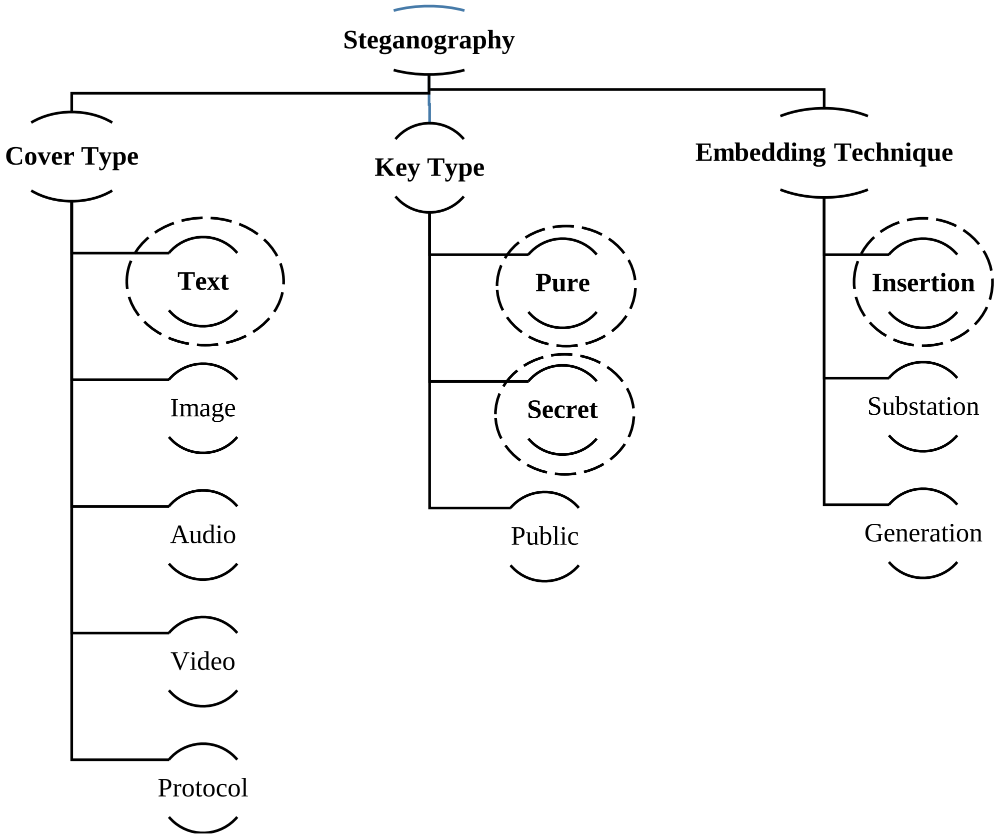


Figure 2.2 Classification of steganography

As shown in Figure 2.2, this study selects the text file as a cover medium and embeds the hidden message by using a secret stego key or without using a stego key (pure). The different classifications of steganography are described in subsections 2.4.1until 2.4.3.

2.4.1 Classification Based On Cover Medium Types

The first point to be clarified in this subsection is the definition of the cover file or the container of the hidden information or secret message. The characteristics of the carrier file or some of its parts can be altered, modified, or manipulated to hide confidential information. However, the manipulations taking place during the embedding process must remain undetected to everyone except for the intended message recipients. Therefore, the format or visual aspect of the carrier files must remain intact after hiding the secret data.

As a result, confidential information can be embedded into various types of cover media. Although the properties of cover files may vary based on the redundancy in the digital representation and unique cover file format, these properties still control the ways in which secret data are embedded into the digital description of cover files. To this end, the cover (carrier) text is a fundamental component of the steganographic system. According to (Amirtharaj & Rayappan 2013), the file types of the cover medium can be classified into text, image, audio, video, or protocol files as shown in Figure 2.2. Various steganographic techniques based on cover medium types are described as follows.

a. Text-based steganography

Text steganography uses text as a cover medium and is widely considered as one of the most problematic steganography techniques (Karadogan & Das 2014; Wai & Khine 2011) given that text files lack redundant data to cover a hidden message (Obeidat 2017; Vidhya & Paul 2015). Another oft-overlooked flaw of this technique is that an unwanted party may intercept text steganography by merely replacing the text itself or by reformatting the text to some other form (such as .txt to .pdf). Various text-based steganography techniques have been developed over the past years, including line shift, word shift, open spaces, and semantic.

Despite these limitations, text steganography is still being used due to various motivations. The first motivation is that a secret message may easily bypass the attention of attackers by encoding this message within an Internet article or an email message. The text may also be classified as a spam message to avoid notice. Propagation steganography may be used to produce artificial messages that resemble spam messages, which rarely receive any attention due to their frequent occurrence. The “mimic algorithm” or “mimicry,” which can be used to generate spam messages artificially, has been classified as a propagation steganographic technique. Mimic texts may not be linguistically correct and may be able to fool spam filters statistically. When investigated, mimic texts can be easily detected by humans while retaining the characteristics of spam messages.

b. Image-based steganography

As its name suggests, image-based steganography utilizes images as cover media. Image-based steganographic techniques are most widely used as they benefit from the limited capacity of the human visual system (HVS) (Mohamed & Mohamed 2016; Sheelu 2013). Messages contain a significant amount of redundant information that facilitates the concealment of confidential information (Subhedar & Mankar 2014; Banerjee et al. 2011). The pixels included within an image offers many possibilities for encoding information and are indecipherable by the stego key, which may take the form of an algorithm. Given the complexity of these pixels, the data may smoothly go undetected by the human eye.

c. Audio-based steganography

Audio-based steganography utilizes audio as a cover medium (Sheelu 2013). Those messages that are concealed by audio-based steganographic techniques are the most difficult to be intercepted because they are concealed in a way that they become imperceptible by the human ear (Kaur & Mahajan 2016; Thorat & Kharat 2015; Bheda et al. 2013). Some people may face difficulties in listening to a tone that immediately follows a louder tone. Therefore, the data may be hidden by using barely audible noises to overcome audio compression and to prevent human ears from picking up the original file. Another possibility is using a very faint echo to conceal the data and to delay the interception of these data. The file formats for audio steganography may include .wav, .midi, and .avi (Das & Bandyopadhyay 2015). Several other techniques for audio-based steganography have also been proposed in the literature, including LSB, parity, phase coding, spread spectrum, and echo hiding (Patil & Pawar 2016).

d. Video-based steganography

Video-based steganography uses videos as cover media. Given that videos are essentially several images in a sequence within a specific frame rate, they can hide information in ways similar to images. Discrete cosine transform changes the values (such as 8.667 to 9) that are applied when inserting data within images in a video. This approach typically ensures that the information is imperceptible by the human eye. Typical file formats for video-based steganography include .mp4, H.264, .avi, and .mpeg (Das & Bandyopadhyay 2015; Hussain & Hussain 2013).

e. Protocol-based steganography

Protocol steganography employs network protocols, typically UDP, ICMP, TCP, and IP, as cover media. Some hidden channels within OSI network layer models apply steganography in unused header bits within TCP and IP fields (Kaur & Rani 2016; Das & Bandyopadhyay 2015).

2.4.2 Type Keys-Based Classification

Three types of steganographic protocols were identified in (Rafat & Sher 2013; Dunbar 2002). Namely, pure steganography, secret key steganography, and public key steganography. Each of these protocols is described as follows.

a. Pure steganography

Pure steganography is a steganographic system that does not require a stego key as a cipher. Therefore, this technique offers minimal security because the transmitter and receiver only rely on random guessing and presumptions about the behavior of each other to encipher the message. The message may be shared among friends in social media or over the Internet.

b. Secret key steganography

In contrast to pure steganographic systems, secret key steganography is a steganographic system that requires a stego key. This technique utilizes a cover and keeps the stego key hidden. Those parties that are private to the key can reverse the process to decipher the message. Given that a stego key is a fundamental feature of secret key steganography, attackers can easily notice and react to the message. However, despite this limitation, only those parties with knowledge of the key can extract the message.

c. Public key steganography

Public key steganography utilizes concepts from public key cryptography. This steganographic technique uses public and private keys to exchange hidden messages. The sender may use the public key to encode the message, which may only be enciphered with the assigned private key. This feature has a direct mathematical relationship with the public key, which can decipher the secret message. Therefore, public key steganography may be a robust approach for implementing steganographic systems given that much-established research has been published on public key cryptography. Numerous levels of security are also utilized to dispel the suspicion of parties that are not involved in the communication. However, as soon as the attackers notice the communication and are alerted to this steganographic technique, they may find a way to crack the algorithm that holds the public key and ultimately intercept the hidden message.

2.4.3 Embedding-Techniques-Based Classification

Three techniques for hiding information within cover media have been proposed in the literature, including insertion-based, substitution-based and generation-based techniques (Baawi et al. 2017; Rafat & Sher 2013; Odeh et al. 2012; Cole 2003).

a. Insertion-based technique

The insertion-based technique designates areas in cover media that will be ignored by processing applications that can read the cover file and by choosing suitable areas within the media for embedding hidden messages. Given that this technique adds secret messages within cover files, one advantage of the insertion-based technique is preserving the original contents of the cover media. However, as one of its major limitations, the large size of the stego file may arouse suspicion. Therefore, the primary aim of the algorithms is to add secret messages without arousing the suspicion of attackers. An embedding characteristic of most files is that they may contain a mark in either EOF or HTML.

b. Substitution-based technique

The substitution-based technique depends on interchanging the component of a carrier file with the secret message in a way that cannot be detected by attackers. This technique works by substituting some bits of information or by deliberately modifying the cover file while producing the least amount of distortion to the cover file. Consequently, the sizes of both the stego file and carrier file become similar. To avoid suspicion, a suitable replacement process must be applied, thereby making it necessary to choose and replace only the insignificant components of the carrier file.

c. Generation-based technique

The generation-based technique disregards the use of a cover file and instead applies a generation engine that uses the secret message as an input to generate a file that appears to be regular and may be presented in text, music, or graphics format.

2.5 REQUIREMENTS OF STEGANOGRAPHIC SYSTEM

Capacity, robustness, and perceptual transparency are the three general requirements for safely hiding information and for measuring the strengths and weaknesses of text steganography techniques. These requirements are collectively known as “the magic triangle” and contradict one another (Abbasi et al. 2015; Ardakani et al. 2015; Alginahi et al. 2014; Zaker & Hamzeh 2012). The relative importance of each of these requirements is mainly based on the steganographic technique applied. Figure 2.3 illustrates the three essential requirements for designing a steganographic system, which will be discussed in the following subsections.

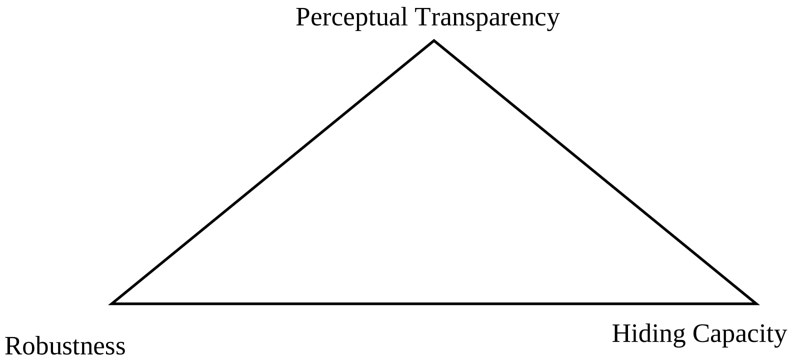


Figure 2.3 Steganographic system requirements

2.5.1 Hiding Capacity

Hiding capacity is a primary decisive parameter for analyzing the performance of a text steganography algorithm. As shown in (Kumar et al. 2014; Satir & Isik 2012), hiding capacity or bit rate refers to the size of the hidden data relative to the size of the stego. Before discussing the evaluation of this requirement, one must distinguish two common terms that are interchangeably used in steganography articles when referring to the capacity, namely, hiding capacity and data payload. The latter refers to the amount of data that can be embedded into the cover, while the former refers to the maximum repetition of the data payload within a stego.

Steganographic systems are mainly used for secret communications and aim to maximize the steganographic capacity and maintain the perceptual transparency of the cover. Therefore, two factors of capacity measurement have been used, namely, hiding ratio (HR) and saving space ratio (SSR) (Din et al. 2017; Osman et al. 2015). More details are given in Chapter 5, subsection 3.3.5c.

2.5.2 Transparency

Transparency or undetectability refers to the nonsuspicious expression of stego texts (Roslan et al. 2011) and represents a data security factor. A high degree of security can be achieved by minimizing the impact of hiding in cover texts (Aman et al. 2017; Obeidat 2017; Abbasi et al. 2015; Ardakani et al. 2015; Mohamed 2014). This task can be completed by reducing the change between cover and stego texts by making the change to parts of the texts that are difficult to detect.

The techniques or methods for evaluating the non-detectability or transparency of steganographic systems may differ from one system to another depending on the type of cover media used for hiding information. For instance, image quality indicates the non-detectability of image-based steganography, whereas file size may reveal the presence of hidden data within a text file and therefore lead to their detection.

Two types of perceptibility, namely, fidelity and quality, can be distinguished and evaluated for transparency. Fidelity refers to the perceptual similarity between covers before and after the embedding process; such similarity can be measured by the human eye (Ardakani et al. 2015). Meanwhile, quality is an absolute measure of the goodness of a cover; this type can be measured by using the Jaro–Winkler quantitative measure of the similarity between the cover and stego (Agarwal 2013). For example, the stego text looks identical to the cover text but also has low quality. However, given that the stego text is indistinguishable from the cover text, the former has high fidelity. More details can be found in Chapter 5, Sections 3.3.5a and 3.3.5b.

2.5.3 Robustness

The importance of robustness differs between steganography and watermarking systems. Robustness is the primary requirement of watermarking systems yet is considered secondary in steganography techniques (Alginahi et al. 2014). This requirement refers to system resistance, that is, the meaning of the confidential information shows significant resistance to intentional and unintentional changes. In the text, steganalysis can perform different processes, including scaling, rotation, changing colors, copy and paste, retyping, and OCR.

As mentioned earlier, the compromise among steganographic capacity, robustness, and transparency must be considered. Even those steganography techniques that hide secret data by using huge file sizes yet apply distortions to stego files are not very useful in maintaining data security. A much more effective solution for such compromise is to increase the steganographic capacity and to control a certain level of stego text quality. Enhancing the stego text quality while retaining steganographic capacity may offer a significant contribution.

2.6 STEGANALYSIS

Steganalysis refers to the art and science of determining whether a media file contains a secret message, extracting that covert message, and finding out its contents. The analogy between steganography and cryptoanalysis is determined by referring to the attempts of breaking cryptographic protocols. Cryptanalysis is considered useful for cryptographic protocols even if the encrypted message is discovered by unwanted parties (Karadogan & Das 2014). An additional requirement is added by steganography, which asserts that the secret message may not be detectable by adverse parties. In other words, adverse parties are also unaware of the existence of the message. Therefore, steganalysis is only useful when the invisibility of the secret message is compromised (Bennett 2004)

2.6.1 Types of Steganographic Attacks

Various types of steganographic attacks include stego-only, known cover, known message, chosen stego, chosen message, and known stego. Steganographic methods may be classified according to the tools that unwanted parties may utilize to analyze the steganographic text. The types of attacks used by steganalysis are described in the following sub-subsections (Böhme 2010) and illustrated in Figure 2.4.

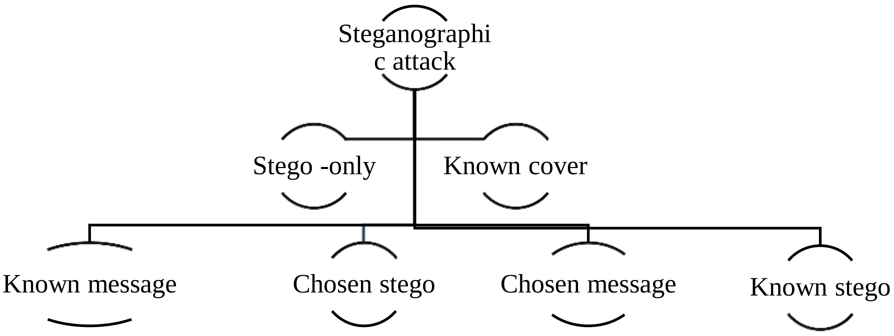


Figure 2.4 Types of steganographic attacks

a. Stego-only attack

In stego-only attacks, the stego file is solely available for analysis, while both the stego file and hidden information are vulnerable to attacks.

b. Known cover attack

In known cover attacks, the original cover file is contrasted with the stego file, which will facilitate the detection of pattern differences. Both the original image and the image that contains the hidden information are available and can be compared.

c. Known message attack

Known message attacks analyze the known patterns that correspond to confidential information and may help prevent future attacks. Even with the message, this type of attack may be complicated to address and may even be considered similar to the stego-only attack.

d. Chosen stego attack

In chosen stego attacks, the steganography algorithm (tool), which may be a software, stego file, or other similar information, is known.

e. Chosen message attack

In chosen message attacks, the steganalyst generates a stego file from some steganography tool or algorithm of a selected message. The goal of this attack is to determine corresponding patterns in the stego file that may point toward the usage of specific steganography tools or algorithms.

f. Known stego attack

In known stego attacks, the steganography tool (algorithm) is known or both the original cover and stego file is available.

The attacker may analyze the steganographic text by performing these attacks (Bennett 2004). Table 2.1 summarizes the potential tools that a steganalyst (attacker) may use in certain situations.

Table 2.1 Summary of steganographic attacks

Type of attacks	Stego File	Original cover file	Hidden message	Stego algorithm or tool
Stego only	√			
Known cover	√	√		
Known message	√		√	
Chosen stego	√			√
Chosen message	√		(see clarification)	
Known stego	√	√		√

2.6.2 Steganalysis Methods

Among the attack types listed in the above table, the known message attack may be surmised from an algorithm that produces a stego file with a hidden message. Therefore, this algorithm may be used for detecting and extracting hidden messages. In the chosen message attack, the attacker inputs various messages within the cover media into numerous algorithms with an aim to check whether the stego file possesses certain properties that resemble the unitary role of the utilized algorithms. All the aforementioned attacks may be classified into visual and aural, structural, or statistical attacks (Bennett 2004), as summarized in Figure 2.5.

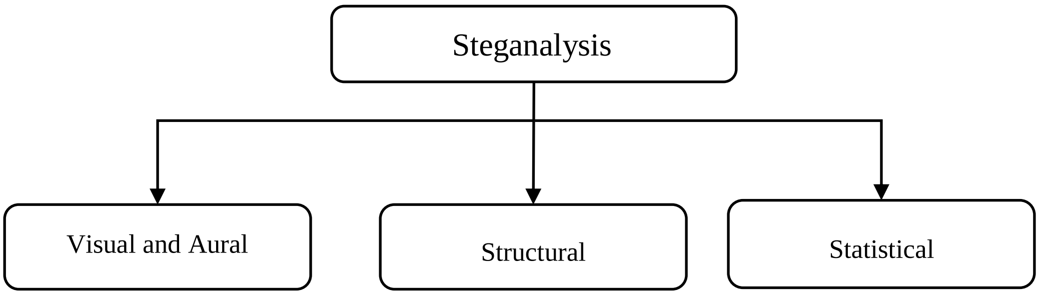


Figure 2.5 Steganalysis attacks

Visual and aural attacks focus on the human factor given that humans can visually or aurally detect whether certain information “looks right.” Showing the LSB of an image offers a standard test of detection. Text steganography attacks can be further extended to grammatical, format-based, lexical, rhetorical, or semantic attacks.

In structural attacks, the format of the digital file is altered while the secret messages are being embedded. The changes to the file format may be very perceptible. In other words, structural attacks affect the structure of the file format itself and detect the patterns in the changes applied to the data format and the file contents.

Statistical attacks examine the anomalies within the statistical profile of the stego file. The main purpose of steganalysis is to collect sufficient statistical evidence to support the presence of images that contain hidden information.

2.7 TEXT STEGANOGRAPHY

Text steganography refers to the process of embedding text into another text to keep its data secure. This method aims to prevent unwanted users from obtaining secret messages for their private use (Singh et al. 2012). This method is also more challenging to use compared with audio and image steganography given the limited amount of redundant data. Text steganography is often applied to cover media and render most data hiding techniques insufficient in terms of capacity and security (Kingslin & Kavitha 2015; Roy & Venkateswaran 2013). Nevertheless, the effectiveness of cover files in hiding secret messages depends on whether the redundant data are detected (i.e., an altered cover file must remain undetected by unwanted users).

Given many limitations of text steganography, previous studies have utilized text steganography methods in other languages, including Arabic (Roslan et al. 2014; Bensaad & Yagoubi 2011; Roslan et al. 2011), English (Agarwal 2013; Bhaya et al. 2013), Indian (Shah & Chouhan 2014; Souvik Roy & Venkateswaran 2013). Farsi (Persian) (Ardakani et al. 2015), Uyghur (Talip et al. 2012), Czech (Khan, Sankineni, et al. 2015), Telugu (Prasad & Alla 2011). These methods may be categorized into format-based, random and statistical generation, and linguistic methods as illustrated in subsections 2.7.1 until 2.7.3. These three categories are described as follows.

2.7.1 Format-Based Methods

Format-based methods are named such because they may need to physically alter the format of the cover text. However, doing so may produce unwanted consequences. For example, these changes can be easily detected when the suspected steganographic text is compared with the original text. Small differences, such as extra white spaces, different font sizes, and misspellings, can also be detected by using a basic word processor (Khairullah 2014). The various methods for embedding information into text files are described as follows.

a. Line shifting coding

Line shifting coding is conducted by vertically shifting lines by a few degrees. For example, each line can be moved up or down by 1/300 inch, thereby making this method especially suitable for printed text. This shifting may represent binary values that transmit hidden information, and marked lines reveal the direction of the shift (Sangita Roy & Manasmita 2011) as shown in Figure 2.6. Nevertheless, the secret message becomes prone to detection when specialized software, such as optical character recognition (OCR), is utilized or when new text is added to the document.

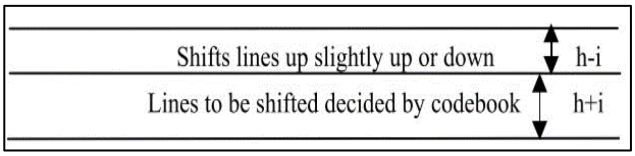

Shifts lines up slightly up or down
h-i
Lines to be shifted decided by codebook
h+i

Figure 2.6 Example of line shifting coding
Source: (Sangita Roy & Manasmita 2011)

b. Word shifting coding

In contrast to line shifting coding, word shifting coding moves words horizontally than vertically. Therefore, this method is especially suitable for texts with different distances between words. Horizontal shifts tend to be less perceptible than the changes in vertical line spacing because different distances between words in a line is common (Sangita Roy & Manasmita 2011) as shown in Figure 2.7. However, word shifting coding may be detected by unwanted parties who may be familiar with the algorithm because this algorithm uses existing texts to check for key variations within the text. Word shifting coding has two other limitations. First, this method examines the text to check for any horizontal shifts among the words. Second, attackers may simply retype the shifted words by using OCR software

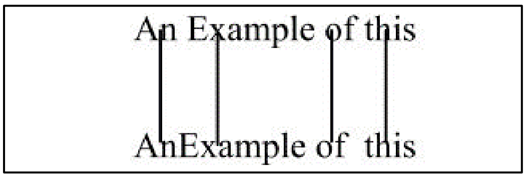
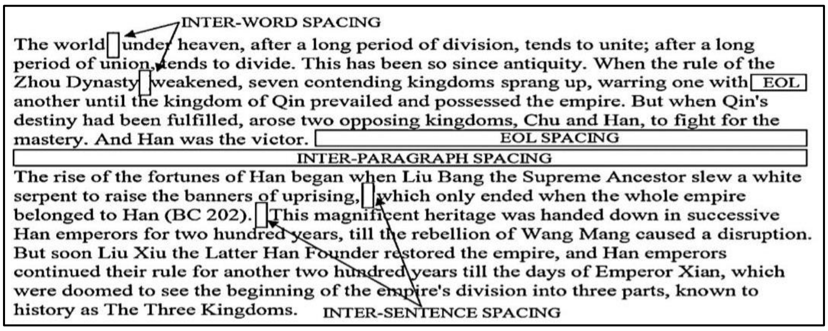


Figure 2.7   Example of word shifting coding
Source: (Sangita Roy & Manasmita 2011)

c.   Open space coding

As the most popular text steganography method, open space coding places white spaces on the text to conceal coded data (Por et al. 2012), as shown in Figure 2.8. These white areas prevent potential identification by unwanted parties by retaining the definitions of the text document. These white areas can also hide secret messages by accounting for the spaces between sentences, end-of-line or end-of-file spaces, and areas in between words (see (Por et al. 2012)).


INTER-WORD SPACING
EOL
EOL SPACING
INTER-PARAGRAPH SPACING
INTER-SENTENCE SPACING

The world under heaven, after a long period of division, tends to unite; after a long period of union, tends to divide. This has been so since antiquity. When the rule of the Zhou Dynasty weakened, seven contending kingdoms sprang up, warring one with another until the kingdom of Qin prevailed and possessed the empire. But when Qin's destiny had been fulfilled, arose two opposing kingdoms, Chu and Han, to fight for the mastery. And Han was the victor.

The rise of the fortunes of Han began when Liu Bang the Supreme Ancestor slew a white serpent to raise the banners of uprising, which only ended when the whole empire belonged to Han (BC 202). This magnificent heritage was handed down in successive Han emperors for two hundred years, till the rebellion of Wang Mang caused a disruption. But soon Liu Xiu the Latter Han Founder restored the empire, and Han emperors continued their rule for another two hundred years till the days of Emperor Xian, which were doomed to see the beginning of the empire's division into three parts, known to history as The Three Kingdoms.

Figure 2.8   Example of open space coding
Source: (Por et al. 2012)

d. Feature coding

Feature coding modifies the features of texts in various ways. First, the specific characteristics of the text may be altered, including its font type (Bhaya et al. 2013; Bhuvaneshwari et al. 2013; Bhaya 2011), font size and style (Mahato et al. 2014; Samanta et al. 2014). Letter cases may also be alternated between lower and upper cases (Ali & Saad 2013). Other features that can be altered include the height of characters and the letter points (such as in letters “i" and “j”) (Khan, Abhijitha, et al. 2015), Other possible approaches include replacing the multi-points in letters (Odeh et al. 2012), and using separate letters at the beginning and end of a word to hide data (Mohamed 2014).

2.7.2 Random and Statistical Generation Methods

Random and statistical generation methods randomize the sequence of characters in a text. Statistical properties, including the word length or the occurrence frequency of a letter, are utilized in text documents in order for specific words to share the statistical properties of words in a given language (Zhang & Zhong 2014).

2.7.3 Linguistic Methods

Many studies (Vidhya & Paul 2015; Prasad & Alla 2011; Shu et al. 2011) have applied linguistic methods, which depend on the properties of the language used for the generated and altered text. The coded message is embedded in the syntax of the language. Linguistic methods can be classified into syntactic and semantic methods. In syntactic methods, punctuations are inserted into certain places within the document, while in semantic methods, some words in the document are replaced with synonyms (Huanhuan et al. 2017; Mahato et al. 2017; Chang & Clark 2014). To efficiently utilize linguistic methods, one must consider two factors, namely, the hiding and space–saving ratio and the capacity of the hidden text, when comparing the original text with the cover text.

Figure 2.9 classifies the main text steganography methods into three categories as described in (Ekodeck & Ndoundam 2016; Tutuncu & Hassan 2015; Odeh et al. 2014; Agarwal 2013). This study applies open space coding, which is classified as a format-based method.

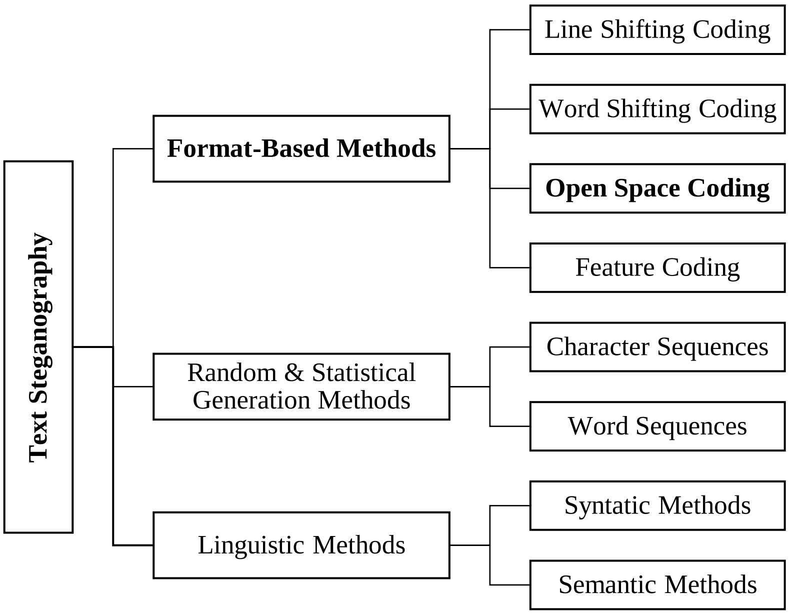


Figure 2.9 Categories of main text steganography methods

2.7.4 Related Work on Text Steganography

Many studies have applied different text steganography methods to embed data within texts and in various languages depending on the properties and features of each language. A steganography system must satisfy several requirements, including capacity, transparency, and robustness. Accordingly, many studies have devised methods that can fulfill all these requirements. These methods can be classified into format-based, random and statistical generation and linguistic methods (Kingslin & Kavitha 2015; Agarwal 2013; Amirtharaj & Rayappan 2013) as summarized in Figure 2.9. The advantages and limitations of these approaches are reviewed in detail as follows.

a. Format-based methods

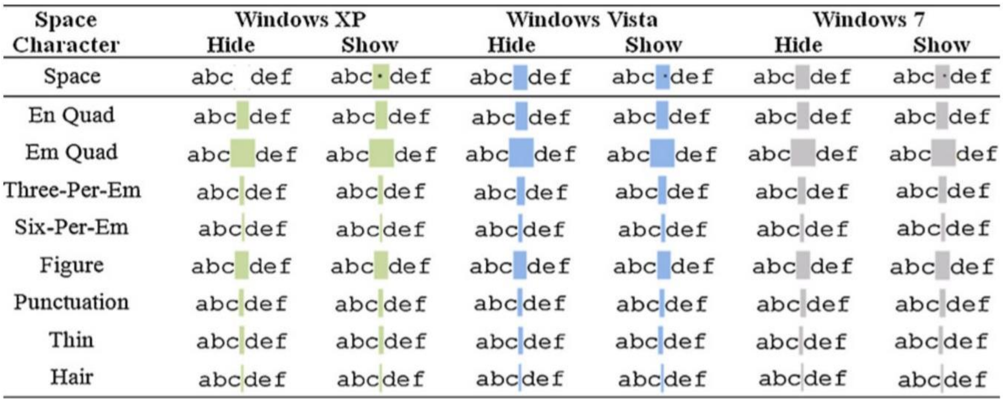
Format-based methods physically format the text to embed information. They do not alter any word or sentence, thereby retaining the value of the cover text. Several studies have employed format-based methods to improve the capacity of text steganography.

Sangita Roy and Manasmita (2011) proposed a text steganography algorithm that employs special characters, line shifting, and word shifting to code the text. This algorithm aims to develop a copy protection technique with a high cover object capacity that converts the secret data from their original characters into hidden binary data, thereby allowing the embedding of multiple bits into a single line of the cover text and making the changes to the original document impossible to detect. However, given the high computational complexity and a large volume of text required for encoding a small number of bits, this algorithm is considered inefficient. This algorithm also shows low robustness because the encoded private data are lost when the spaces are removed by using a word processing software (Sangita Roy & Manasmita 2011).

Por et al. (2012) employed another format-based method that utilizes open space coding. This approach is based on UniSpaCh, a space character manipulation technique that is particularly suitable for Microsoft Word documents that use Unicode space characters. Specifically, UniSpaCh uses the white spaces in any document by embedding the payload into inter-sentence, inter-word, end-of-line, and inter-paragraph spacing using Unicode space characters. Manipulating white spaces does not significantly alter the overall appearance of a document. Compared with conventional methods, UniSpaCh is undetectable, has a higher hiding efficiency, and remains robust when subjected to DASH attacks (Por et al. 2012). The contents of the cover document also have minimal influence on the hiding efficiency of UniSpaCh according to (Obeidat 2017).

DASH attacks are structural attacks that detect any space character modifications applied to the text based on the type of spacing. Actually, the Unicode standard contains 18 space characters according to (Por et al. 2012) can be seen in Table 2.2.

Table 2.2 Space characters in the Unicode standard
Source: (Por et al. 2012)



Space Character	Windows XP Hide	Windows XP Show	Windows Vista Hide	Windows Vista Show	Windows 7 Hide	Windows 7 Show
Space	abc def	abc·def	abc def	abc·def	abc def	abc·def
En Quad	abc def	abc def	abc def	abc def	abc def	abc def
Em Quad	abc def	abc def	abc def	abc def	abc def	abc def
Three-Per-Em	abc def	abc def	abc def	abc def	abc def	abc def
Six-Per-Em	abc def	abc def	abc def	abc def	abc def	abc def
Figure	abc def	abc def	abc def	abc def	abc def	abc def
Punctuation	abc def	abc def	abc def	abc def	abc def	abc def
Thin	abc def	abc def	abc def	abc def	abc def	abc def
Hair	abc def	abc def	abc def	abc def	abc def	abc def

Vertical point shifting is another format-based steganographic method. According to (Bensaad & Yagoubi 2011), this method is fragile when subjected to attacks, such as when using OCR and retyping the text by using different fonts. In response to this drawback, Odeh et al. (2012) proposed a novel vertical point shifting method that examines the shifting and point-to-point distance and then uses the results to pass two bits in each multipoint letter. The stego file is then converted into an image to improve its hiding capacity and robustness, thereby addressing retyping issues. This algorithm can also be applied to other languages, such as Pashto and Urdu (Odeh et al. 2012). Table 2.3 lists several cases that use multipoint letters. This method falls under the category of feature coding using format-based methods (see Table 2.3).

Table 2.3 Cases that use multipoint letters

Secret Bits	Shifting Points	Distance Between Points
00	No shift	Normal
01	No shift	Points are separated
10	Shift up	Normal
11	Shift up	Points are separated

Few studies on feature coding (character coding) have applied format-based methods to deal with font attributes, including type, case, size, color, and style (italic, bold, and underline). For instance, Bhaya (2011) developed a text steganography method that utilizes SMS texts as carriers. The character fonts in a message are changed in a way that the proportional font holds the bit value “1” or “0.” The results generated show perceptual transparency and complexity. However, this method shows low capacity given the fixed sizes of SMS messages in cellphones (Bhaya 2011).

Consequently, Bhaya et al. (2013) proposed SEFT, a new text steganography method that uses Microsoft Word to replace fonts with other fonts that possess similar features. The hidden messages are concealed within the capital letters of the carrier, thereby highlighting the very high capacity and perceptual transparency of this method. However, in SEFT, the file size of the stego document increases by roughly 0.766% of its original size, thereby making the hidden messages detectable (Bhaya et al. 2013).

Novel text steganography techniques for feature coding has generally been classified as format-based methods. Stojanov et al. (2014) proposed a data hiding method called “property coding,” which uses Microsoft Word as the carrier and adopts the properties of document objects, including character scaling, underline style, or paragraph borders, to hide the data. However, property coding cannot be used in copyright protection applications given that this method lacks robustness and is bound to certain file types (Xiao et al. 2016). Although this method shows relatively high accuracy, a slight increase in file size (roughly 1%) may make the message perceptible (Stojanov et al. 2014).

To achieve a visual stego text quality, some format-based methods use the invisible spaces between words to hide data. For instance, Mahato et al. (2014) hid secret messages in unseen spaces by changing the font size in Microsoft Word files. They assumed that a slight difference between invisible spaces and other letters in terms of font size will not be reflected in both the file and the disk space. Therefore, steganography can be intelligently achieved. These format-based methods also have a very high hiding capacity. They can be applied to all languages and can resist several types of attacks, including copying, cutting, changing text colors and font styles, and inserting a word or phrase in the text. However, these methods show some limitations, such as increasing the size of the stego file and the tendency for retyping to destroy the robustness of the algorithm (Mahato et al. 2014).

Other researchers have proposed text steganography methods that are based on glyphs of the shapes of characters. Roslan et al. (2014) attempted to increase the capacity of text steganography by using the primitive structure of Arabic characters. Their method concealed secret bits within primitive structures, such as the dots, sharp edges, and typographical proportion of Arabic letters. Their findings underscore the high capacity and perceptual transparency of their proposed method and reveal that this method offers a low level of security (Roslan et al. 2014).

Mohamed (2014) provided high-capacity and high-security algorithms that cover media by using Arabic text. These algorithms search for letters within the words displayed in the carrier text and then conceal the data in the same carrier text by using letters without creating any noticeable changes in the target word. The isolated characters at the start and end of a word are considered to simplify these algorithms. The findings reveal the high carrier media capacity–hiding capacity rate ratio of these algorithms and their resistance to traditional attacks by applying small changes to the carrier text (Mohamed 2014).

Al-Asadi and Bhaya (2016) proposed another text steganography technique classified as a format-based method. This technique aims to hide data in Excel sheets by changing both the font color and type of each character in the text within a cell. This technique uses two similar colors with different values for two bits (1,0) and two similar font types with different font types for two bits (1,0). Each pair of characters is hidden within seven characters in the cover text. This technique shows high transparency and can make the recovery of the secret message especially difficult. However, given that the secret message will be destroyed if the attacker changes the font color or font type, this technique shows low robustness (Al-Asadi & Bhaya 2016).

Besides, Kumar et al. (2016) employed several text steganography methods to hide data within Microsoft Word files. Specifically, the secret data bits are hidden within white spaces by altering the font types and styles. Given that these white areas are actually invisible characters, any changes in their style largely go unnoticed and do not arouse any suspicion. Therefore, any alterations made within the cover text are not visible and are visually imperceptible. Consequently, this method achieves high holding capacity, high hiding efficiency, and good visual transparency. The main advantage of this method is that no additional space or characters are added to hide the secret data, thereby retaining the file size of the document. The authors also measured the number of concealed bits that were engrafted within the cover files and compared their findings with those generated by the UniSpaCh method (Kumar et al. 2016).

b. Random and statistical generation methods

The second category to text steganography is random and statistical generation methods. For instance, Souvik Roy and Venkateswaran (2013) proposed a zero-distortion technique may be utilized in a wide variety of cases. If a match is found among bit values, the positions are saved within matrices of spots. The model comprises an array of information which can be used to generate secret texts. In order to ensure the security of array positions, the positions are encrypted using an indexed-based chaotic sequence. The results of the aforementioned method may provide a layer for authentication and another for security, applicable for physical security. The applications of the latter may include online shopping or online banking. However, flaws of this method are that the size of the file is increased significantly and larger messages exceeding the number of words are required. The ratio of the secret message to the original message is 1: 2, thus determining that this method needed to double the required number of words to hide the secret message.

Another novel text steganography method has been developed based on the circular nature of English language characters. According to (Dulera et al. 2011), this method specifies a quadruple categorization by considering the curves and the vertical and horizontal lines present in English language characters. This feature can store two bits behind one character at one time, thereby making this method applicable to diagrams. The applicability of these approaches on a specific data set should be protected secret to secure them.

Majumder and Changder (2013) proposed a novel text steganography method that generates a summary of an English text file. This approach receives and inserts concealed information into a publicly available text. The secret message is then embedded into the summary depending on whether the reflection symmetry features of the English alphabet letters lie along the axis of reflection. This approach tries to create a cover text by summarizing the selected input text that is sent to the receiver. After receiving the cover text, secret bits are extracted to recover the original message based on the similarities in the features of the English alphabet. However, the experimental results indicate that this method can embed huge amounts of data at minimal security and low transparency (Iyer & Lakhtaria 2016).

Satir and Isik (2012) utilized a steganography method that aims to address capacity and security issues. This approach uses the Lempel–Ziv–Welch (LZW) data compression algorithm to hide information in text documents. The authors selected email as their communication medium and a mail forwarding platform as their stego cover. The embedding process uses two secret keys, namely, a global stego key comprising numerous email addresses and a set of selected email addresses that have been modified. The value of the second key in the embedding process is built based on the bits of the original message, and the algorithm generates the bits of the secret message. In the extraction phase, each element of the first key is compared with the value of the second key to decode the message. The experimental results reveal that this algorithm outperforms the existing methods by 6.92% (Satir & Isik 2012). However, the stego file shows poor quality and is prone to attacks.

The value of the second key for the embedding process is determined based on the bits of the original message and the algorithm-generated bits of the hidden message. During the extraction phase, the various elements of the first key are contrasted with the value of the second key to decode the message. The hiding capacity of the pixel is determined from the complexity of the cover image. As can be seen in the findings, the algorithm that uses secret messages with a length of 300 characters outperforms the conventional techniques by 7.042%. However, the quality of the stego file remains lacking, thereby increasing its vulnerability to attacks; the code words also need to be recalculated whenever the content of the cover file is altered (Rahman et al. 2017).

Agarwal (2013) proposed three text steganography methods. The first method employs the missing letter puzzle theme, in which each character in a message is hidden by hiding one or more letters in the cover text. This method obtains an average Jaro score of 0.95, thereby highlighting a close similarity between the cover and stego files. The second approach embeds a message in a list of words where the ASCII value of the embedded character determines the length and starting letter of a word. The third approach conceals a message without degrading the cover by using the first and last letters of the cover text. To enhance security, the secret message is encrypted by applying a one-time pad scheme before the embedding process. The ciphertext is then embedded into the cover text (Agarwal 2013).

Satir and Isik (2014), proposed another method that uses the stego key to enhance security. The hidden content is concealed within a text that is constructed from naturally generated texts. They used email as their medium and the stego file as their forwarding mail platform. However, if only a few characters are considered or when the possibility for one character to appear is too high, then the Huffman compression method shows a low bit rate, a reduced capacity, and time complexity (n log n) (Saniei & Faez 2013).

Other text steganography techniques improve the cover capacity and provide high security by using data compression techniques. One example of these techniques has been described in (Lee & Chen 2013), who proposed a lossless text steganography method that applies Huffman compression coding. The authors applied variable Huffman coding to generate codewords for the symbols in the cover file and then hide the secret data in these codewords. In this way, their proposed method achieves an excellent hiding capacity and a reduced transmission cost. However, this method has very poor transparency and is prone to attacks (Rahman et al. 2017).

Kumar et al. (2014) proposed a high-capacity email-based text steganography method that uses combinational compression and a forwarding email platform to hide secret data in email addresses. This approach adopts a combined Burrows-Wheeler transform + moves to forwarding + LZW coding algorithm to increase the hiding capacity to 7.03%. The number of characters in an email ID is also used to represent the secret data bits while enhancing capacity. Random characters are added immediately before the "@" symbol of email addresses to increase randomness (Kumar et al. 2014).

Tutuncu and Hassan (2015) proposed an embedding method that is a hybrid of lossless compression techniques and the Vigenere cipher developed by Tutuncu and Hassan. The authors used an email environment to cover the embedded message. The distance matrix is organized after choosing the cover text with the highest repetition pattern related to the secret message. The size of the secret message is reduced by creating a hybrid of run-length encoding, Burrows-Wheeler transforms, move to the front (MTF), and arithmetic encoding lossless compression algorithms sequence. The Vigenere cipher provides another layer of security or complexity to the system to obtain the stego key (K1). The experimental results prove that this approach achieves a favorable hiding capacity and that the Vigenere cipher improves the security or complexity of the system.

Kumar, Malik, et al. (2016) proposed an email-based text steganography method that uses Huffman compression. This method improves the capacity of the cover object in order to minimize the costs of communication and uses the number of characters in an email ID to maximize their utilization and to indicate the bits of hidden messages. The experimental results show that this method outperforms other existing methods in terms of hiding capacity.

Malik et al. (2017) introduced a high-capacity text steganography method based on LZW compression and color coding that directly applies the LZW technique on the secret data and embeds the obtained bitstream into email addresses and the email message, thereby increasing the capacity by around 13.43%. However, as its primary weakness, this method uses colors to hide the data in the email by following some color coding (Obeidat 2017).

Ekodeck and Ndoundam (2016) proposed four methods for increasing the amount of information that can be hidden in a cover PDF file. These methods were based on the Chinese remainder theorem to reduce the number of A0’s insertions between characters considerably. These methods reduce the difference between the weights of a cover file and a stego file in which the secret message is hidden by ensuring that the number of embedded A0’s is less than the number of characters or by limiting the increase in the number of inserted A0’s. Although these methods increase the difficulty for attackers to detect the integration of secret information into the cover file, they can greatly increase the number of pages in the file (Ekodeck & Ndoundam 2016).

c. Linguistic methods

Linguistic-based text steganography methods consider the use of language properties in text modification. Therefore, altering the physical format of the text is necessary when using these methods.

Mir and Hussain (2011) developed a few text steganography methods for secretly communicating messages that are specified in a textual format. They found that although using synonyms and acronyms is suitable for hiding secret steganography messages in digital content, this method is only secure when the lists are in possession of the user. If a third party obtains these lists, then the security of this method is breached. (Mir & Hussain 2011)

Shu et al. (2011) introduced a novel text steganography method that uses clear text to hide secret messages. The hidden information can be efficiently distributed via repeated extraction and can be hidden by using any of the available carrier texts. This method can be effectively improved. Based on semantics, this method eliminates the need to change the layout and modify or delete the elements of the carrier text. Therefore, this approach shows high robustness (Shu et al. 2011).

Vidhya and Paul (2015) proposed a novel linguistic steganographic method that enhances the safety of information exchange. Given its limited awareness of the local language, this method depended on the use of the Malayalam language as cover text and the use of Unicode and hiding algorithms. Two matrices are employed to index the characters in the common language and in local languages in ascending order. The Unicode extraction method selects the Malayalam text that corresponds to the English text. A diagonal encoding scheme is then applied. This method not only provides security but also a highly precise encoding process and a balanced decoding process.

Shivania et al. (2015) proposed a new method for enhancing security and data hiding capacity without distorting the cover object. This method combines the abbreviation method with the zero-distortion technique to address the limitation of the former (i.e., only a small amount of data can be embedded in a text file at a certain time). The abbreviation method can also reduce the size of the secret text. In other words, this method can abbreviate a massive secret text to reduce its size. The zero-distortion technique is then applied to ensure that the cover image is not distorted and to hide large volumes of data without significantly changing the cover image (Shivani et al. 2015).

Chang and Clark (2014) proposed a novel lexical substitution-based stego system by applying the vertex coding method with a higher data concealing capacity compared with previous systems. The vertex coding method describes the synonym replacement as a synonym graph to be mainly used for English texts. In this way, the relations between words can be clearly recognized. The selected words are replaced with the same part of speech synonyms. Therefore, the modification tends to be grammatical as it does not affect the structure of the sentence. In other words, text paraphrasing is a multi-word substitution process. The evaluation results indicate that the stego system achieves low capacity by reaching the payload capacity of around two bits per sentence within a reasonable level of security (Chang & Clark 2014).

Qi et al. (2014) applied synonym substitution that improves the security of the steganographic scheme by utilizing the advantage of the abandoned synonym in traditional steganography-based synonym substitution. The experimental results reveal that this approach has a higher capacity and robustness compared with conventional methods and achieves minimal syntax error when applied to English texts (Qi et al. 2014).

Table 2.4 summarizes the works above on text steganography, including their findings, limitations, and categories of their proposed methods.

Table 2.4 Related works on text steganography

Ref.	Technique(s)	Category	Finding(s)	Limitation(s)
(Sangita Roy & Manasmita 2011)	The special character, Line shifting, and word shifting coding techniques.	Format-based	Provides good hiding capacity and no change in the stego file.	Its low robustness
(Por et al. 2012)	Modifying inter-sentence, inter-word, end of the line and inter-paragraph spacing.	Format-based	Higher hiding efficiency. Robust to DASH attack. Also, the contents of the cover file have minimal influence on hiding efficiency.	It requires changing the content of the cover text (Obeidat 2017).
(Odeh et al. 2012)	Multi-point letters	Format-based	Enhance the capacity and robustness.	
(Bhaya 2011)	Changing font type.	Format-based	Provide perceptual transparency and complex.	Its low capacity
(Bhaya et al. 2013)	Capitalizing selected letters in the cover media and changed the font type.	Format-based	The capacity is very high and has excellent transparency.	Stego document increased by approximately 0.766% from the original size.
(Stojanov et al. 2014)	Property Coding: character scale or underline and the border of paragraph and sentence.	Format-based	High capacity.	Its low robustness and a slight increase in the size of stego-file of approximately 0.1%.
(Mahato et al. 2014)	Modifying inter-word spacing by changing the Font Size of invisible space characters.	Format-based	Very high capacity	The stego file increased the size and retyping can destroy robustness of the algorithm.
(Roslan et al. 2014)	Primitive structure; Sharp edges, dots, the typographical proportion of the Arabic letter.	Format-based	Higher capacity and higher transparency.	It is low security.

to be continued…

… continuation

(Mohamed 2014)	Using single letters without any noticeable change in the target word.	Format-based	Provides a relatively secure algorithm and Resist traditional attacking methods.	Provides a low capacity.
(Al-Asadi & Bhaya 2016)	Based on changing the font type and font color.	Format-based	Achieves capacity and security.	It has low robustness.
(Kumar, Singh, et al. 2016)	Hidden data within white spaces by altering the font types and styles.	Format-based	Achieves capacity and transparency.	It has low robustness when changing the font or retyping the text
(Souvik Roy & Venkateswaran 2013)	The frequency of letters and Vedic Numeric Code			
(Dulera et al. 2011)	Random character sequencing and feature coding methods	Format-based	Increased randomness, thus aiding in higher security	quality is low due to lower overhead on the stego file.
(Majumder & Changder 2013)	Generating the summary of a textual file.	Format-based	Higher capacity	Provides low security. However, it has low quality (Iyer & Lakhtaria 2016)
(Satir & Isik 2012)	Apply the LZW data compression algorithm.	Data compression-based	Provides a significant increment with regard to capacity.	The quality of the stego file is very poor. As a result, the stego file becomes more prone to attack (Rahman et al. 2017).
(Agarwal 2013)	Employ three methods of text steganography include: the theme of the missing letter puzzle, wordlist as well as start and end letters of words.	Format-based	Improve security.	
(Satir & Isik 2014)	Based on Huffman lossless compression algorithm	Data compression-based	It has a higher capacity compared to (Satir & Isik 2012), can be applied to any language, has optimal prefix code and strong against OCR programs and retyping.	The quality of the stego file is very poor. As a result, the stego file becomes more prone to attack.
(Lee & Chen 2013)	Used a lossless compression coding, which termed variable Huffman coding.	Data compression-based	High hiding capacity, reduced transmission cost.	quality is very poor and prone to attack (Rahman et al. 2017).

to be continued

… continuation

(Kumar et al. 2014)	Combined Burrows-Wheeler transform + moves to forwarding + LZW coding algorithm	Data compression-based	The hiding capacity is 7.03%.	poor quality
(Tutuncu & Hassan 2015)	A hybrid of Run Length Encoding, Burrows-Wheeler Transform, Move to Front (MTF), Run Length Encoding, and Arithmetic Encoding lossless compression algorithms sequence.	Data compression-based	Achieved the hiding capacity and also improved security.	
(Kumar, Malik, et al. 2016)	Based on employing the number of characters used in email id to indicate the hidden secret data.	Data compression-based	Provides a high hiding capacity.	
(Malik et al. 2017)	Based on employing an LZW compression technique and color coding-based approach.	Data compression-based	Produces a high hiding capacity.	Produces low security due to change the color of cover text which attracts attention (Obeidat 2017).
(Ekodeck & Ndoundam 2016)	Based on the Chinese Remainder Theorem	Format-based	Produces a high hiding capacity and the security improved.	The inconvenient with the files that have big size.
(Mir & Hussain 2011)	AES algorithm and synonyms	Linguistic method	Improves security.	
(Shu et al. 2011)	An algorithm based on multi-text.	Linguistic method	Improves security and higher robustness.	
(Vidhya & Paul 2015)	Unicode extraction and diagonal encoding indexing.	Format-based	Achieved greater security.	(Vidhya & Paul 2015)
(Shivani et al. 2015)	Abbreviation method and Zero Distortion Technique	Linguistic method	Increase security and improves data hiding capacity.	
(Chang & Clark 2014)	Replacing selected words with the same part of speech synonyms.	Linguistic method	Provides a reasonable level of security	Achieved a small capacity
(Qi et al. 2014)	Employ the benefit of an abandoned synonym in traditional steganography-based synonym substitution.	Linguistic method	Achieved higher capacity, robustness	A minimal creating syntax error in English text.

2.7.5 Discussion and Analysis of Related Work

The ultimate challenge in text steganography is to achieve a favorable hiding capacity, which is often limited by the lack of redundant data in textual documents compared with digital media, including image, audio, and video files. Different text steganography methods have been proposed in the literature as shown in above table. These methods are employed in text steganography to enhance hiding capacity and maintain transparency. Although some of these techniques can achieve a favorable hiding capacity, they suffer from limited transparency. Meanwhile, other methods demonstrate a favorable transparency yet show limited capacity as will be explained further in the following paragraphs.

Table 2.4 shows that numerous text steganography techniques have been applied to enhance the capacity of the cover media and to provide high security such as described in (Shivani et al. 2015; Tutuncu & Hassan 2015; Lee & Chen 2013; Satir & Isik 2012; Dulera et al. 2011). For instance, the method proposed by Malik et al. (2017) and Majumder and Changder (2013) can achieve high capacity yet offer low security, while that proposed by Chang and Clark (2014), and Mohamed (2014) can achieve a reasonable level of security yet offer a low capacity. However, a few techniques can be classified as having low capacity, although the provide high transparency such as that presented by (Bhaya 2011). Kumar, Malik, et al. (2016) proposed another method that shows higher hiding capacity compared with other methods.

As mentioned in Table 2.4, several algorithms for text steganography demonstrate high transparency in producing high-quality stego file and provide high security at the same time. Examples of these algorithms include those introduced in (Bhaya et al. 2013; Souvik Roy & Venkateswaran 2013). Meanwhile, other methods such as proposed by Roslan et al. (2014) and Majumder and Changder (2013), provide high capacity and transparency yet offer a low level of security. Several other techniques, such as those proposed by (Stojanov et al. 2014; Banerjee et al. 2013), achieve high transparency and capacity yet increase the overhead on stego files.

Table 2.4, shows a few methods that can improve security, such as those introduced by (Vidhya & Paul 2015; Mir & Hussain 2011).

Certain methods have been designed to improve robustness and provide high capacity, such as those introduced by Qi et al. (2014), Por et al. (2012), and Odeh et al. (2012). Besides, the work by Shu et al. (2011) developed a highly robust method that also provides a high level of security. Meanwhile, other works, such as Al-Asadi and Bhaya (2016), Mahato et al. (2014) and Sangita Roy and Manasmita (2011), proposed methods with low robustness and high capacity. Some techniques, such as that proposed by Stojanov et al. (2014), demonstrate low robustness and high capacity, while those techniques proposed by Ekodeck and Ndoundam (2016), and Satir and Isik (2014) show high capacity, improved security, and an increased overhead on the stego file.

These findings indicate that when designing or developing a steganography technique, one must consider the primary requirements (e.g., hiding capacity, transparency, and robustness), maintain the quality of the stego medium, and evaluate these methods by using most of the evaluation criteria proposed in the literature.

2.7.6 Proposed Classification For Text Steganography

This study classifies text steganography methods based on how they embed secret data. Specifically, these methods are classified based on their formatting, linguistics, and data compression techniques. This classification is justified and explained in further detail in the following paragraphs.

The main reason for this suggestion is that the existing methods have worked towards highlighting characteristics that are contributed to providing more significant opportunities to hide textual information without arousing any suspicion. For instance, Unicode character, in current classification is included under open space coding. Although, the main problem with the open space approaches is suffering from the robustness because of the absence of extra security layers (Aman et al. 2017). Furthermore, some text editor programs automatically delete additional white spaces and thus destroy the hidden information (Khairullah 2018). Whereas, many recent studies that are utilized invisible Unicode character proved it achieved a high level of robustness and resistance to attack especially with DASH attack (Hosmani et al. 2015; Por et al. 2012).

Meanwhile, font format is classified under feature coding of the format-based method as described in the existing classification. Whereas some current studies focused on manipulation in font features and some features are provided a little of capacity such as the work that presented in (Bhaya 2011) and this conflicts with what exists. Therefore, this study suggests that Unicode character and font format classify as individual kinds under the format-based methods.

On the other hand, Linguistic methods are dealing with the nature and properties of language. It deals both with the study of particular languages, and the search for general features common to all languages or large groups of languages. Linguistics consists of subareas such as phonetics, phonology, morphology, syntax, semantics, and pragmatics (Akmajian et al. 2010; Radford et al. 2009). In line with the subareas of the linguistics that this study proposes the inclusion of random and statistical generation method under linguistic methods.

A nowadays group of researchers have done toward employing lossless data compression techniques (Malik et al. 2017; Satir & Isik 2014, 2012), that are contributed to providing more considerable scope to hide textual information with minimal arousing suspicion. This leads to achieving the hiding capacity and transparency which are considered as the primary requirements in steganography. Therefore, this study suggests that data compression is regarded as a branch in text steganography classification.

2.8 RELATED CONCEPTS

The following sections discuss the associated concepts that are used to facilitate the development of a solution to issues in text-based steganography. This study specifically focuses on four core concepts, namely, linguistics, Unicode language, hiding strategy, and data compression. These four concepts are discussed as follows.

2.8.1 Linguistics

Linguistics can be defined as the science of language, including its structure and constituents (e.g., phonetics, phonology, morphology, syntax, semantics, and pragmatics). Linguistics is studied for two primary purposes. The first purpose relates to the natural context of language, with some researchers attempting to establish a theory of language with which a certain language can be described. The second purpose is to examine all forms of language while aiming to achieve a scientific understanding of those methods by which a certain language is organized to fulfill those functions that facilitate communication among humans.

Figure 2.10 shows the core branches of linguistics. Given that each language has an identifiable structure, their structural elements comprise four components, namely, sounds, words, sentence structure, and meaning. Each language has words, and words are likely the most accessible linguistic units to the layman. The presentations of English words is examined within the context of linguistics in English. As mention by Akmajian et al. (2010) that the words play an integral role in the human ability to use language creatively. Human vocabulary is a dynamic system, far from being a static repository of memorized information. Consequent can add words at will and also can even expand their meanings into new domains.

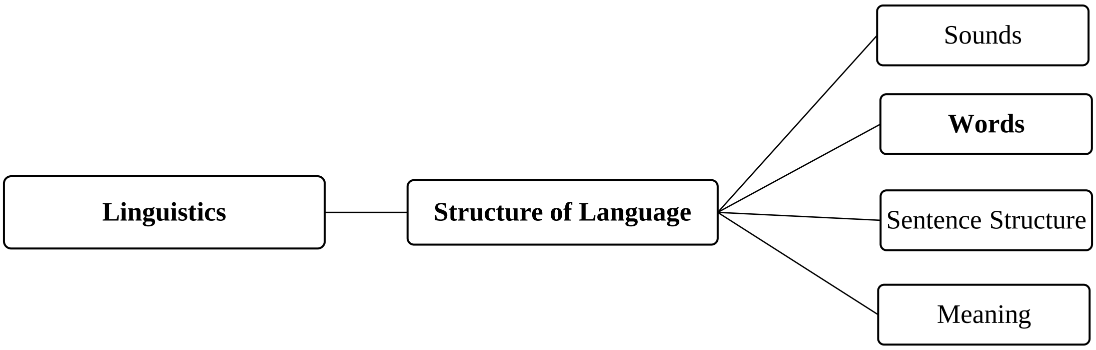


Figure 2.10 Core branches of linguistics

A particularly interesting observation is that many words with variable lengths are used in English writing. Examples of these words include a one-letter word (e.g. a and I), two-letter words (e.g. am, is and on), three-letter words (e.g. are, can, has and was) and four-letter words (e.g. were, here, dear, done and into).

Moreover, the language used in the text Steganography, as each language has its hold individual properties which are fully different from other languages. For instance, the letter shape in the English language does not rely on its position in the word, while Arabic or Persian letters own different forms relying on letter position in the words (Odeh et al. 2012).

Table 2.5 lists a set of Arabic or Persian letter which can be represented by four shapes on the basis of its position in a word.

Table 2.5 Some Arabic letters shape

Name	Isolated	Initial	Middle	Final
baa	ب	بـ	ـبـ	ب
taa	ت	تـ	ـتـ	ت
thaa	ث	ثـ	ـثـ	ث
Jeem	ج	جـ	ـجـ	ـج
haa	ح	حـ	ـحـ	ـح
khaa	خ	خـ	ـخـ	ـخ
seen	س	سـ	ـسـ	س
sheen	ش	شـ	ـشـ	ش
saad	ص	صـ	ـصـ	ص
daad	ض	ضـ	ـضـ	ض
taa	ط	ط	ـطـ	ط
thaa	ظ	ظـ	ـظـ	ظ
ayn	ع	عـ	ـعـ	ـع
ghayn	غ	غـ	ـغـ	ـغ
faa	ف	فـ	ـفـ	ف
qaaf	ق	قـ	ـقـ	ق
kaaf	ك	كـ	ـكـ	ك
laam	ل	لـ	ـلـ	ل
meem	م	مـ	ـمـ	م
noon	ن	نـ	ـنـ	ن
haa	ه	هـ	ـهـ	ـه
yaa	ي	يـ	ـيـ	ي