



University of Kerbala  
College of Computer Science & Information Technology  
Computer Science Department

# **Coverless Text Steganography Methods Based on Arabic Language Features**

A Thesis

Submitted to the Council of the College of Computer Science & Information  
Technology / University of Kerbala in Partial Fulfillment of the Requirements  
for the Master Degree in Computer Science

**Written by**

**Sabaa Hamid Rashid Hassan**

**Supervised by**

**Asst. Prof. Dr. Dhamyaa Abbas Habeeb**

2024 A.D.

1445 A.H.

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

يُؤْتِي الْحِكْمَةَ مَنْ يَشَاءُ ۚ وَمَنْ يُؤْتَ الْحِكْمَةَ فَقَدْ أُوتِيَ خَيْرًا  
كَثِيرًا ۗ وَمَا يَذَّكَّرُ إِلَّا أُولُو الْأَلْبَابِ

صدق الله العظيم

سورة البقرة

الآية 269

## **Supervisor Certification**

I certify that the thesis entitled (**Coverless Text Steganography Methods Based on Arabic Language Features**) was prepared under my supervision at the department of Computer Science/College of Computer Science & Information Technology/ University of Kerbala as partial fulfillment of the requirements of the degree of Master in Computer Science.

Signature:

Supervisor Name: Asst. Prof. Dr. Dhamyaa Abbas Habeeb Nasrawi

Date:     /     /2024

## **The Head of the Department Certification**

In view of the available recommendations, I forward the thesis entitled “**Coverless Text Steganography Methods Based on Arabic Language Features**” for debate by the examination committee.

Signature:

Dr. Asst. Prof. Dr. Muhannad Kamil Abdulhameed

Head of Computer Science Department

Date:     /     /2024

## **Certification of the Examination Committee**

We hereby certify that we have studied the dissertation entitled (**Coverless Text Steganography Methods Based on Arabic Language Features**) presented by the student (**Sabaa Hamid Rashid**) and examined him/her in its content and what is related to it, and that, in our opinion, it is adequate with (**Excellent**) standing as a thesis for the degree of Master in Computer Science.

Signature:  
Name: Majid Jabbar Jawad  
Title: Prof. Dr.  
Date:    /    / 2024  
(**Chairman**)

Signature:  
Name: Ayad Hameed Mousa  
Title: Assist. Prof. Dr.  
Date:    /    / 2024  
(**Member**)

Signature:  
Name: Mohammed Mohsen Hassoun  
Title: Assist. Prof. Dr.  
Date:    /    / 2024  
(**Member**)

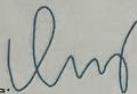
Signature:  
Name: Dhamyaa Abbas Habeeb  
Title: Assist. Prof. Dr.  
Date:    /    / 2024  
(**Member and Supervisor**)

Approved by the Dean of the College of Computer Science & Information Technology, University of Kerbala.

Signature:  
Assist. Prof. Dr. Mowafak Khadom Mohsen  
Date:    /    / 2024  
(**Dean of College of Computer Science & Information Technology**)

### **Supervisor Certification**

I certify that the thesis entitled (**Coverless Text Steganography Methods Based on Arabic Language Features**) was prepared under my supervision at the department of Computer Science/College of Computer Science & Information Technology/ University of Kerbala as partial fulfillment of the requirements of the degree of Master in Computer Science.


Signature: 

Supervisor Name: Asst. Prof. Dr. Dhamyaa Abbas Habeeb Nasrawi

Date: 4/19/2024

### **The Head of the Department Certification**

In view of the available recommendations, I forward the thesis entitled "**Coverless Text Steganography Methods Based on Arabic Language Features**" for debate by the examination committee.

Signature: 

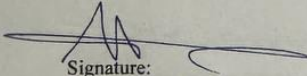
Dr. Asst. Prof. Dr. Muhannad Kamil Abdulhameed

Head of Computer Science Department

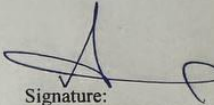
Date: / /2024

### Certification of the Examination Committee

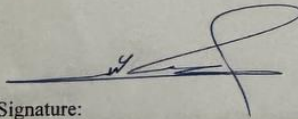
We hereby certify that we have studied the dissertation entitled (**Coverless Text Steganography Methods Based on Arabic Language Features**) presented by the student (**Sabaa Hamid Rashid**) and examined him/her in its content and what is related to it, and that, in our opinion, it is adequate with (**Excellent**) standing as a thesis for the degree of Master in Computer Science.



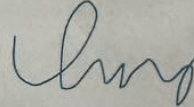
Signature:  
Name: Majid Jabbar Jawad  
Title: Prof. Dr.  
Date: / / 2024  
(Chairman)



Signature:  
Name: Ayad Hameed Mousa  
Title: Assist. Prof. Dr.  
Date: / / 2024  
(Member)



Signature:  
Name: Mohammed Mohsen Hassoun  
Title: Assist. Prof. Dr.  
Date: / / 2024  
(Member)



Signature:  
Name: Dhamyaa Abbas Habeeb  
Title: Assist. Prof. Dr.  
Date: 4 / 9 / 2024  
(Member and Supervisor)

Approved by the Dean of the College of Computer Science & Information Technology, University of Kerbala.



Signature:

Assist. Prof. Dr. Mowafak Khadom Mohsen  
Date: / / 2024  
(Dean of College of Computer Science & Information Technology)

## **Dedication**

*To my beloved husband, whose unwavering support and encouragement have been my guiding light throughout this journey. Your love and belief in me have fueled my determination to reach this milestone.*

*To my father, my family, my lovely sons, my brothers, and sisters, my friends, and all those who rejoice in my success,*

*Your unwavering support and encouragement have been my strength throughout this journey. I dedicate this Master's thesis to each one of you, for believing in me and standing by me through every challenge and triumph.*

*With heartfelt gratitude,*

*Sabaa Hamid*

## **Acknowledgement**

First and foremost, I express sincere thanks to Allah, the Most Gracious, whose boundless blessings guided me through the journey of my research, culminating in the successful completion of my thesis.

I would like to thank my supervisor, Asst.Prof.Dr. Dhamyaa A. Nasrawi, for his excellent advice, constant support, and insightful input over the course of this thesis. Their advice and support helped shape the direction and quality of this effort.

I extend my sincere appreciation to Assist. Prof. Dr. Mowafak Khadom Mohsen, the distinguished dean of the College, Assist. Prof. Dr. Muhsin Hasan, the Deputy Dean for Scientific Affairs, and Assist. Prof. Dr. Muhannad Kamil Abdulhameed, the esteemed Head of the Computer Science Department, as well as all the esteemed members of the teaching faculty, and my fellow graduate students at the College of Computer Science and Information Technology, University of Karbala. Your support and encouragement have been instrumental in my academic journey.

Sabaa Hamid



## **Abstract**

In the digital era, protecting confidential information from unauthorized access is crucial. Information can be depicted through several forms of communication, such as text, audio, video, and images with text being the most common. Steganography's primary objective is to conceal information so that it is not noticed by others by hiding it in other cover media. There is a significant difference between this method and other covert information exchange techniques. In cryptography, for instance, people detect information by looking at coded data, but they are unable to understand it. But using steganography, no one will ever know that the data is even included in the source. Traditional methods require carriers to disguise secret information which leads to carrier modifications that are challenging to avoid steganalysis. The challenges associated with the coverless text steganography method include low capacity, high perplexity, and the absence of any applications in the Arabic language. In contrast, the carrier is not modified by coverless text steganography and transfers hidden information directly via the stego cover's built-in features.

The thesis aims to improve coverless text steganography techniques in terms of hiding capacity, perplexity, success rate, extracting accuracy, security analysis, and availability. As well as extend the coverless text steganography techniques to the Arabic language, by leveraging the Arabic language statistical model and Arabic language built-in features that are utilized for information hiding.

Two new coverless text steganography methods based on Arabic language features are proposed, the first one is the generation method based on an Arabic

language statistical model using first-order Markov chain, while the second method is a search method based on built-in Arabic language features.

Three Arabic datasets are used in this thesis (SANAD (Single-Label Arabic News Articles Dataset) includes 45500 articles, the Arabic Poem Comprehensive Dataset (APCD) contains 1,831,770 poetic verses in total, the Arabic Poetry Dataset contains more than 58000 poems).

The first method uses a first-order Markov Chain to generate hidden texts without the need for cover media. A dataset of Arabic texts is selected, and a state transition diagram is created based on word frequencies. A code word is used to represent transitions in the diagram, allowing for the generation of text that hides information. The method demonstrated an improvement in hiding capacity, reaching a value of 5.5, and a reduction in perplexity to 18.51, indicating the method's effectiveness in information hiding.

The second method focuses on testing the first word of each row in a dataset against eight features—mahmoze, diacritics, isolated, two sharp edges, vowels, dotted, looping, and high frequency—to generate a byte value (1 or 0) based on the presence or absence of these features. This byte is then converted to a decimal (ASCII code) to establish a dynamic mapping protocol with the most frequent letter. The method achieved a high accuracy rate of 100%, which signifies its precision in embedding and retrieving hidden information without altering the linguistic structure of the text. Moreover, the success rate also reached 100%, underscoring the method's reliability in successfully concealing and uncovering the embedded information. Despite the high success and accuracy rates, the concealment capacity using this method was up to 0.246, which reflects the balance between maintaining the linguistic integrity of the Arabic text and the amount of information that can be concealed.

## **Declaration Associated with this Thesis**

1. Rashid, Sabaa Hamid, and Dhamyaa A. Nasrawi. "Coverless Text Information Hiding Techniques: A Review", Vol. 21 No. 1 (2024): Journal of Kerbala University, Vol 21 Issue 1 June 2024.
2. Rashid, Sabaa Hamid, and Dhamyaa A. Nasrawi. "Generation Method of Dynamic Coverless Arabic Text Information Hiding Using First-Order Markov Chain", submitted to 2024 4th International Conference of Science and Information Technology in Smart Administration (ICSINTESA). (Accepted).
3. Rashid, Sabaa Hamid, and Dhamyaa A. Nasrawi. " Coverless Text Information Hiding Based on Built-in Features of Arabic Scripts." Journal of Applied Data Sciences 5, no. 2 (2024): 653-667. doi: <https://doi.org/10.47738/jads.v5i2.243>

# Table of Contents

Dedication .....	i
Acknowledgement .....	ii
Abstract .....	iii
Table of Contents .....	vi
List of Tables .....	viii
List of Figures .....	ix
List of Abbreviations .....	x
<b>CHAPTER ONE .....</b>	<b>1</b>
1.1 Overview .....	2
1.2 Problem Statement .....	4
1.3 The Aims of the Thesis .....	5
1.4 The Objectives of the Thesis .....	5
1.5 Related works .....	5
1.5.1 Generation Method .....	6
1.5.2 Search Method .....	8
1.6 Thesis Organization .....	17
<b>CHAPTER TWO .....</b>	<b>18</b>
2.1 Overview .....	19
2.2 Concept of Coverless Text Steganography .....	19
2.3 Types of Coverless Text Steganography .....	21
2.4 Characteristics of Coverless Text Steganography .....	23
2.5 Markov Chain Model .....	24
2.5.1 Definition of Markov Chain Model .....	24
2.5.2 Markov Chains in Natural Language Processing .....	26
2.5.3 Markov Chains in Information Hiding .....	27
ava N-Grams .....	29
2.7 Arabic Language Features .....	29
2.8 Evaluation Measures .....	34
2.8.1 Hiding Capacity .....	34
2.8.2 Success Rate .....	35

2.8.3 Extracting Accuracy.....	35
2.8.4 Security Analysis .....	35
2.8.5 Perplexity.....	36
2.8.6 Availability.....	36
<b>CHAPTER THREE.....</b>	<b>38</b>
3.1 Overview .....	39
3.2 Generation Method.....	39
3.2.1 Construction First- Order Markov Chain.....	40
3.2.2 Hiding Processes.....	46
3.2.3 Extraction Processes.....	47
3.3 Search Method .....	50
3.3.1 Dynamic Mapping Protocol.....	51
3.3.2 Hiding Processes .....	58
3.3.3 Extraction Processes .....	60
<b>CHAPTER FOUR.....</b>	<b>62</b>
4.1 Overview .....	63
4.2 Arabic Datasets.....	63
4.3 Generation Method Results.....	64
4.3.1 Perplexity.....	65
4.3.2 Hiding Capacity.....	65
4.3.3 Availability.....	67
4.4 Search Method Results.....	68
4.4.1 Hiding Capacity .....	69
4.4. 2 Availability.....	69
<b>CHAPTER FIVE .....</b>	<b>72</b>
5.1 Conclusions.....	73
5.2 Future Works .....	74
<b>REFERENCES.....</b>	<b>75</b>

## List of Tables

Table 1.1: Overview of coverless text Text Steganography techniques .....	13
Table 2.1: Arabic Alphabets Forms.....	31
Table 2.2: Dots in Arabic letters .....	32
Table 2.3: : Diacritical Marks of Arabic Language .....	32
Table 2.4: The number of sharp edges in Arabic letters .....	33
Table 2.5: Arabic Letter frequency using only the Quran as input source.....	33
Table 2.6: List of evaluation metrics used for coverless text information hiding.....	37
Table 3.1: Present Vocabulary Size .....	42
Table 3.2: Example of Construct 8-bit Based Built-in Features .....	53
Table 3.3: ASCII Code and their Frequency in SANAD Dataset.....	54
Table 3.4: Dynamic English Letter Mapping Results: ASCII Frequencies in Three Dataset.....	56
Table 4.1: Summary of Selected Arabic Datasets.....	64
Table 4.2: Experimental Results of Generation Method with Experiment (1).....	65
Table 4.3: Experimental Results of Generation Method with Experiment (2).....	66
Table 4.4: Experimental Results of Generation Method with Experiment (3).....	66
Table 4.5: Experimental Results of Generation Method with Experiment (4).....	66
Table 4.6: Comparison of Proposed Generation Method with Related Thesis.....	67
Table 4.7 Average of Hiding Capacity for Proposed Search Method.....	69
Table 4.8: Comparison of proposed method with related thesis.....	70

## List of Figures

Figure 2.1: <i>Information security categories</i> .....	20
Figure 2.2: <i>A Boat Under a Sunny Sky” by Lewis Carroll</i> .....	20
Figure 2.3: Weather Forecasting with Markov Chains .....	26
Figure 2.4: State transition graph of sample text .....	28
Figure 3.1: Generation Method Using First-Order Markov Chain .....	40
Figure 3.2: Construction first-order Markov Chain .....	41
Figure 3.3: Sample of First-Order Markov Chain Data Structure .....	43
Figure 3.4: Setting of Code words in First-Order Markov Chain Data Structure .....	44
Figure 3.5: Sub Markov Diagram for Word” ﻻﻱ” .....	45
Figure 3.6: Hiding processes of Generation Method .....	46
Figure 3.7: Matching Secret Message with Code Words in First-Order Markov Chain Data Structure .....	48
Figure 3.8: Matching Secret Message with Code Words in State Transition Diagram.....	48
Figure 3.9 Extraction processes of Generation Method.....	49
Figure 3.10: Search method based on Arabic language features .....	51
Figure 3.11: Dynamic mapping protocol .....	51
Figure 3.12: Distribution of ASCII Code in SANAD Dataset.....	56
Figure 3.13: The English letters frequency (a): tabular data ;(b) histogram .....	51
Figure 3.14: Hiding Procedures of Search Method.....	58
Figure 3.15: Hiding procedure results using SANAD dataset .....	58
Figure 3.16: Hiding procedure results using APCD dataset .....	59
Figure 3.17: Hiding procedure results using Arabic Poetry dataset.....	59
Figure 3.18 Extraction Procedures of Search Method.....	60

## List of Abbreviations

<b>Abbreviation</b>	<b>Description</b>
APCD	Arabic Poem Comprehensive Dataset
BDSs	Binary Digit String
BiDSs	Binary Digital Slices
CSM	Code Square Matrix
IMDB	Internet Movie Database
LLM	Large Language Model
LSB	Least Significant Bit
LSTM	Long Short Term Memory
MSM	Minimal Square Matrix
NER	Named Entity Recognition
NLP	Natural Language Processing
POS	Part of Speech
RNN	Recurrent Neural Network
RNNs	Recurrent Neural Networks
RCV1	Reuters Corpus Volume1
SANAD	Single-Label Arabic News Articles Dataset
STBS	State Transition Binary Sequence
TFIDF	Term Frequency Inverse Term Frequency
URL	Uniform Resource Locator



# **CHAPTER ONE**

## **INTRODUCTION**

## 1.1 Overview

With advancements in signal processing and transmission systems, multimedia data has become easier to alter, access, and replicate. As a result, secure communication and network security are increasingly critical, especially as computers are more widely used in both social and professional settings. Data hiding is a technique that helps secure communication by embedding data into various media types, such as text, video, and images. The goal is to embed the data without noticeably deteriorating the host signal, ensuring that the concealed information remains undetectable and unintelligible to outsiders [1].

Hiding and retrieving secret messages is more efficient and reliable with electronic media than with physical objects due to the smaller size of the secret message compared to the large amount of cover text data. Computers can effectively process the data and algorithms needed for extracting hidden information, allowing the extraction process to be automated when the data is electronic [2].

Steganography is a clandestine communication technique that hides information within other information. Steganography, unlike cryptography, conceals data in a way that is imperceptible to the human eye[3]. Steganography comes from the Greek words ‘steganos,’ which means ‘covered or concealed’, and ‘graphene’, which means ‘writing’[4].

The information intended to be concealed in the cover data is referred to as *embedded data* in the context of steganography. The data that include both the embedded information and the cover signal is known as *stego data*. The method of *embedding* involves incorporating concealed

or embedded information into cover data. The original, including the text, audio, and video, are all called the *cover*. The following formula could serve as a representation of this process:

$$\textit{Cover medium} + \textit{Embedded message} + \textit{Stego key} = \textit{Stego medium}$$

The field of steganalysis belongs to the identification of steganography, message length calculation, and extraction [2],[4].

There are numerous ways to arrange steganography. For instance, the kind of carrier determines whether a file is text, image, audio, video, or a protocol file was used to incorporate hidden information. Depending on the type of key used, there are three distinct methods of hiding information: pure, secret, and public steganography. Additionally, three methods are utilized to secure information in a cover object dependent on the embedding method: insertion-based, substitution-based, and generation-based techniques [5]. Text steganography is used because text files don't include redundant data like images or audio files is thought to be the most challenging [6].

Employing the currently available information concealment techniques to conceal sensitive information necessitates a predefined carrier signal. The chosen carrier can take various forms, including pictures, texts, audio, and videos. When performing the information-concealing process, the unnecessary portion of a picture, text, audio, or video carrier is employed to conceal secret information. Traditional text steganography techniques have a significant problem because they cause particular modifications in the selected carrier despite hidden information.

The modified carrier cannot withstand all steganalysis attacks, making it a poor choice for carrying sensitive data. Therefore, in this approach, there is always a possibility that cyber criminals could access or destroy secret data [7].

Coverless steganography is a subset of steganography methods. Coverless steganography does not embed any messages. Instead, carrier media are chosen in such a way that their qualities reflect the message [3].

## **1.2 Problem Statement**

Traditional Text steganography strategies often introduce specific modifications to the chosen carrier, even when data is hidden. These alterations to the carrier render it less effective for transporting sensitive information, as it may not be resistant to all forms of steganalysis. Consequently, this vulnerability presents a risk of unauthorized access or deletion of confidential information by malicious actors. In contrast, the concept of coverless text steganography addresses these limitations by avoiding alterations to the carrier. This approach allows sensitive data to be transmitted securely between the sender and receiver without modifying the carrier itself. By leveraging or creating secret messages within the coverless carrier, this method enhances resistance to various steganalysis techniques and malicious attempts [7]. However, the coverless Text steganography technique faces several challenges, including limited capacity, high perplexity, and a lack of applications specifically tailored to the Arabic language.

### **1.3 The Aims of the Thesis**

The aim of this thesis is to extend coverless text steganography techniques to the Arabic language by utilizing the Arabic language statistical model and its features. This involves improving these techniques in terms of hiding capacity, perplexity, success rate, extraction accuracy, security analysis, and availability.

### **1.4 The Objectives of the Thesis**

To accomplish the aforesaid aims, a new coverless text steganography methods are proposed and experimentally evaluated in detail:

1. To design a new coverless text steganography within the generation method type based on an Arabic language statistical model using Markov Chains and N-grams.
2. To design a new coverless text steganography within the search method type based on Arabic language built-in features.

### **1.5 Related Works**

This section discusses the literature review on coverless text steganography techniques, their strength, weakness, and performance metrics. The next subsections are dedicated to reviewing relevant studies in two methods of coverless information hiding, the generation method and the search method.

### 1.5.1 Generation method

Wu et al. suggested a number of Markov Chain models based on coverless text steganography techniques. In 2019, they presented a single bit rule-based coverless text steganography technique[8]. The method involves navigating a dataset, selecting sentences with the same began term, establishing a Markov state transition diagram, trimming it for optimal results, and setting a binary code. The method was tested on movie reviews and news datasets, resulting in improved embedding rate (embedding rate= 2.73), and low perplexity ( $15.38\pm 6.77$ ,  $17.05\pm 15.21$ ).

This method was improved in 2019 [9] by substituting the typical fixed bit embedding with the maximal variable bit embedding in method presented by Wu et al. [8]. Experimental results showed improved embedding rate (embedding rate= 2.75) and low perplexity (15.29, 16.91) in movie reviews and news datasets compared to previous thesis.

In 2019 [10] on the basis of the half-frequency crossover rule, a new enhancement was put into place. The approach was evaluated on news and movie review datasets, including examples using 3-gram and 4-gram half-frequency crossover. The results showed an embedding rate of 2.78 and perplexity of  $15.97\pm 7.57$  and  $17.41\pm 8.91$ , respectively.

Another improvement was implemented in 2019 [10] based multi-rule language models instead of a single gram model. The embedding rate= 2.85, and the perplexity was ( $52.05\pm 35.80$ ,  $20.52\pm 13.98$ ) in the two datasets.

A text steganography method for Arabic was presented by N. Alghamdi and L. Berriche , 2019[11]. Huffman Coding is used in conjunction with Markov Chain (MC) for both the encoder and the decoder. Additionally, the stego-text's upper and lower bounds are calculated. Although less resistant to attacks, the suggested method is format independent. The average Embedding capacity before applying Huffman coding = 1.83 and after applying Huffman coding reached 6.8 by using only one dataset.

Then, two improvements were made in 2020, the first in [36] using state transition-binary sequence (STBS)-based to produce coherent semantically and smooth text. The result of perplexity ( $14.07 \pm 8.83$ ,  $13.34 \pm 9.90$ ,  $12.89 \pm 8.75$ ) is minimal compared to similar models in Twitter, IMDB, and News datasets, and the embedding rate is 2.71. The second improvement that made in 2020 [12] by index and one-bit embedding. The perplexity is small among similar models ( $16.78 \pm 7.89$ ,  $14.97 \pm 2.55$ ,  $25.92 \pm 18.59$ ) in Twitter, IMDB, and News datasets respectively, with a 2.95 embedding rate.

Zhang W. et al. 2020 [13] presented a method that uses word association to construct a word node tree, with a higher average hiding capacity, stable success rate, and resistance to detection.

A. Majumder et al. 2023 [14] proposed a new method generates a dataset for domain values, synthesizing a database while maintaining semantic integrity. The strategy outperforms previous approaches and

offers a thesis around for covert communication, assessing effectiveness through security measures and time complexity analysis.

### **1.5.2 Search Method**

The most crucial favorable implications for the progress of coverless text steganography technology suggested by Chen et al. 2015 [15] based on the Chinese mathematical phrase. The method uses tags and keywords to conceal information without altering the carrier text. It uses stego-vector generation, a large database, and an inverse method to recover confidential information. Despite its low capacity (one keyword in a 1-kilobyte) or (one Chinese character in one text) and large database requirements, it is resistant to steganalysis attacks and maintains carrier originality.

Zhou et al.2016 [16] proposed a new method to hide multiple Chinese keywords simultaneously in a text, slightly improving capacity (1.57) but requiring extensive database and low success rate.

A new coverless text steganography method was developed by Zhang et al. 2016 [17] utilizing stego-vectors from secret data and a word rank map. The method withstands steganalysis attacks and requires a large text database.

Shi et al. 2016 [6] proposed a model for detecting secret messages by searching webpages, offering high embedding capacity(10.32%-80.98%) and good imperceptibility.



Liu et al. 2017 [18] improved speech tagging and multi-keyword methods, enhancing success rate, accuracy, hiding capacity, time, and space efficiency using the "Word2Vec" language model and unmodified text carriers.

Zhang et al. 2017[19], [20] improved a method using frequently occurring terms, words rank map, and distance to hide secret information, but required an extensive text database for performance.

Liu et al. 2017 [21] proposed a coverless text steganography technique using news aggregation, converting secret messages into numbers and concealing them within visible online news pieces. The algorithm is easy, reliable, and efficient for stego messages.

Sun et al.2017 [22] presented a coverless text steganography technique utilizing NER systems. The secret message was converted into keywords, searched, and noted using a named entity recognition system. This method embeds directly without modifying the stego\_text, allowing it to withstand current steganalysis and detection techniques.

Xia et al. 2017 [23] suggested a technique that used the Unicode character's least significant bit (LSB) to coverless text information concealment. The method requires an extensive text database and low capacity texts (14 or 15 bits when the number of texts was approximately 200,000) texts.

Wu et al.2018 [24] proposed an approach using Chinese conversion strategies. They segmented secret information into keywords, obtained tag sets based on protocol and ID, and selected specific rules from each set.

This method improved success rates and hiding capacity, and resisted steganalysis.

Wu et al. 2018 [25] improved a previous steganography method using English texts, enhancing hiding capability and steganalysis resistance, but with a lower success rate compared to Chinese coverless steganography.

Long and Liu 2018 [26] improved the method suggested by Chen et al. [15] using obtaining comparable terms by applying word2vec and distance. After expanding the pool of keywords with similar terms, they were able to obtain stego-texts with location tags and keywords with a 100% success rate.

Fu et al. 2018 [27] enhanced method presented by Chen et al. [15] by incorporating a header file for tag placement, aiming to locate hidden keywords in text using binary strings and a certain number of keywords, but requiring extensive database.

Chen X. and Chen S. 2019 [28] proposed a method for improving steganalysis methods by selecting popular terms as compounds, separating secret information, reducing carrier texts, and enhancing hiding capacity.

Ji H. and Fu Z. in 2019 [29] introduced a coverless text steganography method using signal keyword preprocessing and word components as location tags. The method conceals numerous keywords in text, with high success rates but the robustness is minimal, as is the hiding capacity.

Long et al. 2019 [30] proposed a method using web texts to segment secret information into keywords and extract them using the TextRank algorithm and Word2Vec language model, achieving 99% success rate.

Wang K. and Gao Q. 2019 [31] proposed a method using character features to represent binary digits. The method achieved a high embedding rate, security, robustness, and success rate, with advantages like resisting format transformation attacks and improving semantic and statistical detection. Large-scale text corpus was crucial for implementation.

Zhou et al. 2020 [32] proposed a method based on double-tags and twice-send, significantly improved the hiding capacity by 31.38% in experimental results.

Liu Y. and Wu J. 2020 [33] proposed method utilizing Chinese Pinyin as secret labels. The secret data was encoded by mapping the part of speech (POS) of keywords to integers. The "Word2Vec" language model was used in the procedure to increase the keyword set exhibiting excellent concealment capacity, extraction accuracy, success rate, and time efficiency without requiring changes to text carriers.

Xiang et al. 2021 [34] introduced a method using multi-index segmentation to convert big data texts into secret data. The method improved robustness, security, and had a fast average hiding rate of 42.48 bits/s and the average success rate was 94.89%.

Qin et al. 2021 [35] developed a method using big data text. The method, which transmitted a mixed index with TFIDF features and topic model distribution. The method demonstrated high security and resistance

to attacks, but name entities couldn't be hidden, with an average success rate of 98.24% and average hiding capacity of 60.40 and 64.36.

Liu et al. 2021 [36] improved the methods in [15], [24]Liu et al.[32] used Part of Speech (POS) to hide keyword numbers and optimize stego-text retrieval. They used Chinese character components as locating marks and expanded the keyword set using the "Word2Vec" language model. The method enhances embedding capacity, extraction accuracy and embedding success rate.

Wang et al. 2021 [37] introduced a method using Chinese character component structures to convert Minimal Square Matrix (MSM) into Code Square Matrix (CSM) and BDSs into binary digital slices (BiDSs). The method has good imperceptibility, high robustness, 100% success rate, high capacity, and is extensible across languages.

Wen et al. 2021 [38] introduced a new method using Morse Code, lists, loops, initiators, and groupings. The method creates a character correspondence table and allows high-frequency words to be represented. Experiments showed this method improved hiding capacity, confidentiality, and text search rate.

Guan et al. 2022 [39] proposed a polynomial-based coverless text steganography technique for Chinese text. The technique uses tags to increase keyword selection and uses text vocabulary matching to determine keyword location. The method achieved a hiding success rate

of nearly 100% for any webpage text database size, and a 95% success rate even with the smallest database.

Table 1.1 below concludes the methods used in the studies and shows the strengths and weaknesses of each one.

*Table 1.1: Overview of coverless text Text Steganography techniques*

Generation Method						
References with year	Methodology	Strength	Weakness	Dataset	Metrics	Language
2019 [7-9], [40]	According to the Markov model, the probability of transition is used to create stego-text.	It preserves the characteristics and affects the algorithm.	Based on a sample library of sentences that contain the same term, it needs an extensive database.	Collection of reviews about movies and news.	Algorithm performance, embedding rate and perplexity.	English
2019 [11]	Markov Chain (MC) is implemented for encoder and decoder combined with Huffman Coding	Increasing the block size leads to increased system capacity.	It results in a stego-text that has no meaning and alters the imperceptibility attribute.	Arabic corpus (al-aqd alfareed)	high ratio of capacity	Arabic and any Unicode languages
2020 [41], [12]	According to the Markov model, the probability of transition is used to create stego-text.	The features are maintained, and the algorithm is impacted.	Based on a sample library of sentences that contain the same term, it needs an extensive database.	Collection of reviews about movies and news.	Algorithm performance, embedding rate and perplexity	English
2020 [13]	Based on build the word node tree.	Stable success rate, improved hiding capacity, and resistance to detection.	It needs an extensive database.	Online news and film reviews	Resistance to detection Capacity, security,	English
2023 [14]	unique database synthesis technique	Hiding in any language		Database from kaggle	Capacity, success rate, extraction accuracy, security, time complexity	Any language
Search Method						
2015 [15]	It uses and portrays Chinese characters as mathematical expressions.	It resists steganalysis attacks while maintaining the carrier's originality.	With low capacity, unknown the number of keywords, requiring a vast text database.	Chinese text database	Capacity	Chinese

<b>2016 [16], [42]</b>	It retrieves stego-text that contains both the number of keywords and the private information.	It enhances the capacity, knows the total number of keywords, resists steganalysis attacks	It demands an extensive text database, consumes time, limited improvement and a low success rate.	Sogou Labs	Capacity, success rate	Chinese
<b>2016 [43]</b>	Using the rank map to generate stego-vectors from the secret message.	Resisting all kinds of existing steganalysis methods, no cover modification.	Capacity is inadequate, an extensive text database is needed.	Text big data	Capacity, robustness	English
<b>2016 [44]</b>	Based on features of huge amount Internet webpages	Sending only a URL, no original text change, high embedding capacity, and good imperceptibility.	It needs enormous webpages.	Set of webpages	Capacity, validity of the algorithm	Chinese
<b>2017 [18]</b>	Using multiple-keywords to tag parts of speech.	It improves success rate, extraction accuracy, hiding capacity, time, and space efficiency.	It needs an extensive text database.	Chinese corpus of Sogou Lab	Capacity, success rate, extraction accuracy, security, time efficiency.	Chinese
<b>2017[19],[20]</b>	Using a hash of frequently occurring terms and the words rank map.	Safe, can escape from almost all steganalysis methods, no cover modification.	It has a low capacity, needs an extensive text database, needs to change the private keys periodically.	News websites	Capacity	English
<b>2017 [21]</b>	Based on news aggregation, the secret message M is transformed into a large integer.	The algorithm is easy to use, reliable against any steganalysis, not need an extensive database.	The database needs regular intervals updates, and capacity is based on the amount of news.	Sina News	Capacity, robustness, security.	Chinese
<b>2017 [22]</b>	Named entities are used to mark the locations of the hidden information based on big data.	Reliable against any steganalysis, no modification, high security.	It needs an extensive text database.	CoNLL03, ACE05	Robustness, security.	English
<b>2017 [23]</b>	Based on the LSBs of the Unicode characters on the covers.	Resisting current detecting techniques, no modification in cover.	It has low capacity, needs an extensive text database.	Databases of 200,000 texts	Capacity	Chinese
<b>2018 [24]</b>	Based on two tag selecting strategies to determine the tags	Improving success rate and hiding capacity, resisting the steganalysis.	It needs an extensive text database, low capacity, need an index of the database	Sougou Lab.	Capacity, success rate, security.	Chinese
<b>2018 [25]</b>	coverless text steganography based on English texts.	Improving hiding capacity, resisting the steganalysis.	Low success rate, needs an extensive database, need an index of the database	Reuters Corpus Volume1 (RCV1)	Capacity, success rate, security.	English
<b>2018 [26]</b>	Method of text coverless text steganography based on word2vec	High success rate, improving hiding capacity, resists steganalysis, used	It fails if the keyword does not exist in the text big data, needs an extensive database	Sogou Lab	Capacity, success rate.	Chinese

	to obtain similar keywords.	similar words when text retrieval fails.				
<b>2018 [27]</b>	Use tags to position the keywords as much as possible.	It hides more information, resists all steganalysis methods	It needs an extensive database. weaknesses of success rate	Text from the Internet.	Capacity, success rate.	Chinese
<b>2019 [28]</b>	Based on the compound and selection of words	High algorithm efficiency, resists all steganalysis methods	It needs an extensive database, low security.	Text from the Internet.	Capacity, success rate, security	Chinese
<b>2019 [29]</b>	Based on keywords that are used with location tags to build the index.	resists all steganalysis methods, high success rate for tiny texts with a restriction.	It has weak robustness and low hiding capacity.	Sougou Labs.	Capacity, success rate, security, robustness.	Chinese
<b>2019 [30]</b>	Based on web text, consider existing large internet text as big data.	The presence of many webpages.	Extraction accuracy will be decreased with increasing secret information length.	Chinese corpus of Sogou Lab	Capacity, success rate, extraction accuracy.	Chinese
<b>2019 [31]</b>	Based on the parity of stroke numbers in Chinese characters.	It resists attacks, no need for additional information, applied in many languages.	It needs a large text corpus.	webpages	Embedding rate, success rate, availability.	Chinese
<b>2020 [32]</b>	Based on the double tags in a text by odd-even judgment.	Improving the success rate, hiding capacity, and efficiency of algorithm.	It needs an extensive database.	20 million selected carriers	Capacity, success rate.	Chinese
<b>2020 [33]</b>	The Pinyin combinations of two words are used to choose the hidden tags.	It shows improvement in success rate, extraction accuracy, hiding capacity, time, and space efficiency.	It needs an extensive database.	Chinese corpus of Sogou Lab	Capacity, success rate, extraction accuracy, security, time efficiency.	Chinese
<b>2021 [34]</b>	Based on segmenting the secret information into several keywords	Improving robustness, resisted attacks, high security, high success rate, fast average hiding rate	It needs big data text, cannot hide some entity names, and does not reach a 100% success rate.	Chinese corpus of Sogou Lab	Robustness, success rate, security, hiding rate.	Chinese
<b>2021 [35]</b>	Based on the big data on the Internet. secret information contains the TF-IDF and topic model.	Resisting attacks, showing high security.	needs big data text, cannot hide some entity names, not reach a 100% success rate.	Sougou Lab	Robustness, success rate, security, hiding rate, hiding capacity	Chinese
<b>2021 [45]</b>	POS is utilized to hide the number of keywords after pretreatment.	It shows improvement in success rate, extraction accuracy, hiding capacity, time, and space efficiency.	It needs an extensive database.	Chinese corpus of Sogou Lab	Capacity, success rate, extraction accuracy, security, time efficiency.	Chinese
<b>2021 [37]</b>	Using structures' encoding to express various BiDSs to guarantee that a	No matter how long a text is, no modification, good imperceptibility, high		Collected from the Internet science,	Capacity, success rate, robustness.	Chinese

	secret message can be effectively concealed.	robustness, success rate, and capacity, resist steganalysis methods, extensibility to other languages.		novels, news, and e-commerce.	
<b>2021 [38]</b>	Based on a combination of Morse Code.	Improve hiding capacity, confidentiality, efficiency and text search rate.		5076 English texts from the Internet	Capacity, efficiency testing, English
<b>2022 [39]</b>	Hiding a secret message using polynomial encryption.	It shows an improvement in success rate, hiding capacity, security, and extraction accuracy	It needs an extensive database.	Collected from the Internet:	Capacity, success rate, extraction accuracy, security Chinese

In previous research on coverless text steganography, several challenges and limitations have been identified. First, there is a significant lack of studies addressing these techniques in the Arabic language, which limits our understanding of how to effectively apply them to Arabic texts. This gap poses challenges in developing accurate and reliable models for the Arabic language. Additionally, current methods suffer from high perplexity, indicating substantial uncertainty in the generated texts. The capacity for text steganography in these methods is also often limited, reducing the effectiveness of these techniques in practical applications that require the concealment of large amounts of data. These limitations highlight the need for further research aimed at improving these aspects, particularly concerning Arabic texts.



## **1.6 Thesis Organization**

The rest chapter of this thesis are divided as follows:

Chapter Two: This chapter details the theory used in this thesis.

Chapter Three: This chapter focused on the proposed method.

Chapter Four: In this chapter, the results are obtained and discussed.

Chapter Five: This chapter involves conclusions and suggestions for future thesis.

## **CHAPTER TWO**

### **THEORETICAL BACKGROUND**

## **2.1 Overview**

This chapter presents details of coverless text steganography and the theoretical background of this thesis, including an overview of coverless text steganography concept, coverless Text steganography types, and the coverless text steganography characteristics.

The datasets used in this thesis and some preprocessing needed are demonstrated in this chapter.

Markov Chain model, N-Gram, and Arabic language features are explained in detail in this chapter.

Lastly, present the evaluation metrics that were employed to assess the two suggested coverless text steganography system.

## **2.2 Concepts of Coverless Text Steganography**

Information hiding is regarded as a vital discipline by information security experts. Information concealing is a science that keeps confidential information hidden from outsiders by using secret communication between the source and the destination. The classification of information hiding technologies shown in Figure 2.1.

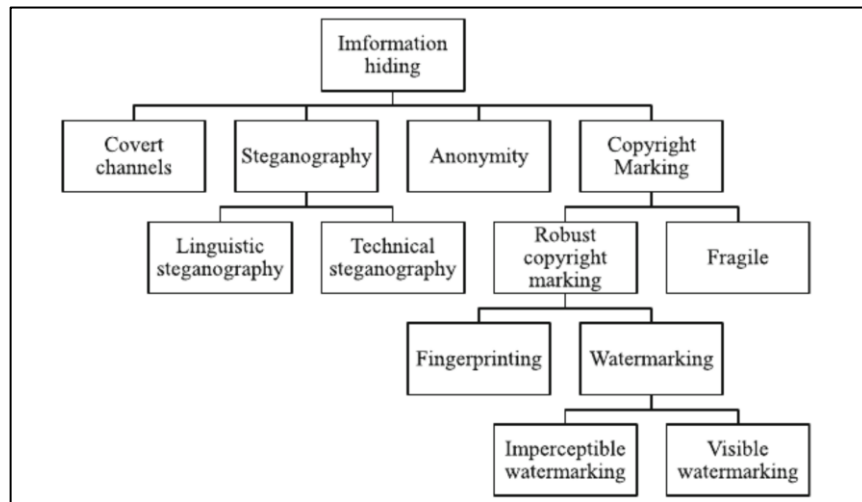


Figure 2.1: Information hiding technologies [46]

Researchers recently coined the term “coverless information hiding” to address the problem with the typical text steganography methods now in use. It is essential to distinguish between the term “coverless” and the lack of a carrier signal. Coverless text steganography refers to the fact that the hidden information does not need to be embedded in a specified carrier. Because the original carrier signal remains unchanged, this method of secret text steganography makes coverless text steganography approaches more resistant to existing steganalysis attacks.

Although coverless information concealment is not a novel concept, it has been implemented daily. An acrostic poem is one of the finest examples that prove that the coverless concealment of data is not an entirely novel concept for people. Lewis Carroll wrote an acrostic poem ‘Alice Pleasance Liddell’ [48] is the individual name that is concealed in this acrostic poem[7], see Figure 2.2

**A boat beneath a sunny sky,  
Lingering onward dreamily  
In an evening of July--**

**Children three that nestle near,  
Eager eye and willing ear,  
Pleased a simple tale to hear--**

**Long has faded that sunny sky:  
Echoes fade and memories die.  
Autumn frosts have slain July.**

**Still she haunts me, phantomwise,  
Alice moving under skies  
Never seen by waking eyes.**

**Children yet, the tale to hear,  
Eager eye and willing ear,  
Lovingly shall nestle near.**

**In a Wonderland they lie,  
Dreaming as the days go by,  
Dreaming as the summers die:**

**Ever drifting down the stream--  
Lingering in the golden gleam--  
Life, what is it but a dream?**

Figure 2.2: “A Boat Under a Sunny Sky” by Lewis Carroll [48]

Texts have less redundant information than images and videos, making it more challenging to apply the techniques of text information-hiding. Nevertheless, texts offer some benefits, including easy encoding, a vast amount of data and a short amount of space occupied, and frequent use. Various scholars have expressed interest in text information concealment technology due to its significance in wireless transmission, covert communication, copyright complications, and other areas. Text steganalysis technology advanced concurrently with the advancement of text steganography and automated text-generation technologies also continues flourishing. This has become a significant challenge to Text Steganography[34]. In general, coverless text steganography techniques

have three main characters (no modification, no embedding, resistant to steganalysis attacks)[27],[24].

## **2.3 Types of Coverless Text Steganography**

The methods of coverless text information hiding, according to Xiang et al. [27], generally fall into two categories:

- Generation method [42][49]. This method automatically generates paraphrases; it is a new and helpful source of transformations for linguistic steganography. Text generation is a critical task in natural language processing (NLP), involving the automatic production of coherent and contextually relevant text by machines. Various techniques have been developed over the years, ranging from traditional methods like Markov chains to advanced deep learning models such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) [50].

In thesis we employed one of the text generation methods, specifically the Markov model. This approach was chosen due to its ability to model the probability of transitions between words based on a given corpus.

- Search method [6]: The suggested method search for accessible carriers on the Internet that are compatible with the secret message's statistical characteristics expecting that a suitable carrier webpage will be found after applying this method. Following the selection of a carrier, the secret message's distribution within the

carrier is scrutinized, and communication relations are added to the carrier's address.

In contrast, Wang et al. [37] present three primary categories of these methods :

- **Steganography by Cover Search:** This method scans the Internet for an appropriate website that can serve as a carrier. This page must contain every character from the hidden message. The URL will be updated with the resulting string after those characters' places on the website are encoded[44].
- **Steganography by Cover Generation:** As the carrier, Chang and Clark [49] create a new text using the N-gram model. Another common example is the NiceText system, which transforms secret messages into different sentences using a sizable code vocabulary [51]. Luo and Huang [52] Chinese poetry is created using the RNN Encoder-Decoder architecture. High-quality text coverings can be created based on the RNN to conceal a hidden bitstream [53].
- **Steganography by Cover Index:** A text corpus from the Internet with books, news, articles, and other contents is developed by Zhou et al.[29] . After breaking down texts into words and creates indexes for each word "label + keyword", it embeds these into secret messages, providing qualified texts. The natural language processing NLP technology's imperceptibility is weak when a secret message is long. Besides, additional semantic problems, including syntax mistakes and poor readability, mainly if a lengthy text is generated.

In this thesis, the first classification, which consists of two types—generation methods and search methods—was adopted because there was no need to use the index.

## **2.4 Characteristics of Coverless Text Steganography**

Traditional text steganography techniques have a significant problem because they cause particular modifications in the selected carrier despite hidden information. The modified carrier cannot withstand all steganalysis attacks, making it a poor choice for carrying sensitive data. Therefore, in this techniques, there is always a possibility that cyber criminals could access or destroy secret data [7].

Coverless text steganography is a solution to the low security and minimal concealment of traditional text steganography techniques. It involves sharing secret information between sender and recipient without changing the carrier. It is possible to choose or produce a secret message from the shared carrier. Due to their coverless nature, the carriers are resistant to a wide range of harmful attacks and most steganalysis techniques [39].

In general, coverless text steganography techniques have three main characters (no modification, no embedding, resistant to steganalysis attacks) [24],[27],[54].



## 2.5 Markov Chain Model

This section presents details of Markov Chain model including a definition, applications in natural language processing (NLP) and information hiding.

### 2.5.1 Definition of Markov Chain Model

In machine learning, Markov models are dynamic and stochastic models [55]. The discrete random process known as the Markov Chain is named after Andre Markov. It clarifies many conditions. It is a discretized random process with no aftereffects. In other words, given the current situation, it is impossible to forecast the process's future state based on its past. Depending on the probability distribution, the system can either maintain its current state or transition to a new one at each step of the Markov Chain. The sequence of random variables that comprise the Markov Chain exhibit the Markov property.

The value of  $x_n$  represents time n. If the conditional probability distribution of  $x_{n+1}$  to the previous states is only a function of  $x_n$ , then equation (1) can be expressed as follows:

$$\begin{aligned} P(X_{n+1} = x_{n+1} | X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) \\ = P(X_{n+1} = x_{n+1} | X_n = x_n) \end{aligned} \quad (1)$$

Where  $X_n$  represents the state of the process at time n ;  $x_n$  is a specific value of the state  $X_n$ ;  $X_{n+1}$  is the state at time n+1 ;  $x_{n+1}$  is its specific value;  $X_1, X_2, \dots, X_n$  are the previous states up to time n ; and

$x_1, x_2, \dots, x_n$  are their corresponding specific values. One could think of this equation as the Markov property [41].

A basic understanding of general processes, state-space, and the concept of state are necessary in order to comprehend the Markov model. A "state" is a group of variables, each with a specific value given to it. These variables are typically used to characterize physical settings; for example, cloudy, wet, and sunny are some of the weather states that exist. Processes change states within the state-space, which is made up of all conceivable states. The term "process" describes the transition between states[14].

In a first-order Markov Chain, every following state in the Chain depends only on the state that came before it. Systems in which the subsequent state is dependent upon two or more previous states are known as Markov Chains of second or higher order.

By considering events as states that transition into new ones or return to their original states, Markov Chains are used to determine the probability that they will occur. Using the example of weather forecasting with Markov Chains shown in Figure 2.3, if it is sunny today, there is a 30% chance that it will be rainy tomorrow; however, there is a 20% chance that the next day will be sunny if it is raining now. Thus, if it is raining today, there is an 80% chance that it will rain tomorrow as well[56].

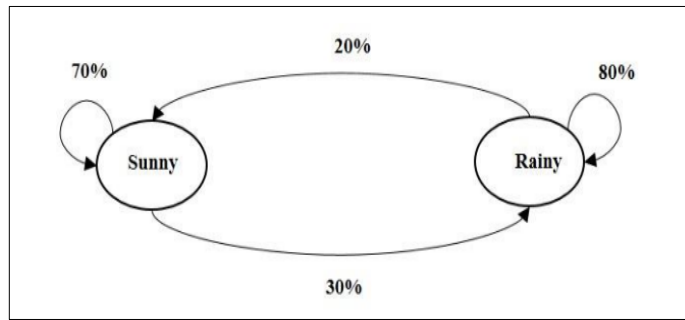


Figure 2.3: Weather Forecasting with Markov Chains[56]

### 2.5.2 Markov Chains in Natural Language Processing

Natural language processing (NLP) is a subfield of artificial intelligence that investigates how human language may be interpreted and processed. NLP aims to uncover effective strategies for reading, interpreting, comprehending, and making sense of human language. Since the 1950s, entity extraction, information retrieval, document indexing, topic modeling, and translation have all been largely reliant on computer comprehension of language. It is used in contemporary computing to manage search engines, identify spam, and boost analytics efficacy in nimble and scalable ways [57].

Stochastic models, such as Markov models, are ideal for natural languages since they are probabilistic and rely on word order to convey meaning in a specific context.

One popular method in machine learning for processing natural language is the use of Markov models. A stochastic modelling technique called Markov Chains is used to represent dynamic systems in which the future state is dependent upon the present state. When producing natural

language, the Markov Chain—which produces a series of words to form a whole sentence—is widely employed [57].

Based on a statistical examination of the transition between words in a sample text, Markov Chains can be used to construct sentences in a given language. The Markov Chain's state space can be composed of various word sequences [56].

NLP has several uses, including machine translation, questions answered, natural language generation, part-of-speech (POS) tagging, text summarization, named entity recognition (NER), and more [57].

The application of the Markov Chain model to natural language generation and information concealment was the primary goal of this thesis.

### **2.5.3 Markov Chains in Information Hiding**

Statistical natural language processing and text steganography have been integrated by numerous academics, and numerous natural language processing methods have been employed to automatically produce steganographic text. The Markov Chain model has been widely used for automatic steganographic text production in recent years, mostly due to its suitability for modelling natural language. For the purpose of determining the transition probability, the majority of these thesis employ the Markov Chain model to determine how frequently each phrase appears in the training set.

Subsequently, the transition probability can be employed to encrypt the text and accomplish the goal of including confidential data during the writing process [58].

Most of these academics' studies compute the frequency of each phrase in the training set and get the transition probability. Then, based on the transition probability, encode the words and embedding secret information in the generated text [58]. The components of Markov Chain (states, transition probability, and code word) obtained from the sample text is shown in Figure 2.4.

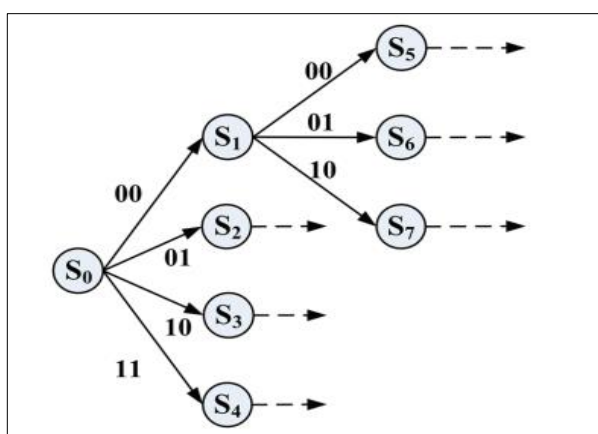


Figure 2.4: State transition graph of sample text [8].

When the secret data was translated to binary coding, it produced a steganographic sentence that started with " $S_0$ " and matched the coding in the state transition graph element by element. assuming that every state transition has two bits of information. The state transition graph indicates that the text  $S_0S_1S_7$  can be recovered when the hidden information is 0010.

The original secret information recovered, the transitions  $S_0 \rightarrow S_1 \rightarrow S_7$  are followed, allowing the binary coding "0010" to be recovered. The state transition graph acts as a map that ensures the correct retrieval of the steganographic text by following the specific path determined by the binary code [12].

## 2.6 N-Grams

Markov models known as "n-grams" estimate words based on a predetermined window of earlier words.

The most fundamental kind of language model is the n-gram model. A group of n words is known as an n-gram. However, the term "n-gram" refer to a probabilistic model that can assign probabilities to complete sequences and predict the probability of a word given the n-1 preceding words.

This is the concept of taking current or previous words to the required word by a specified number of words, where the letter N indicates a specific number, and the word "grams" refers to words. Thus, we have what's called *unigram* for dealing with each word separately, or a *bigram* system which means dealing with two words, or a *trigram* system for dealing with three words [59].

## 2.7 Arabic Language Features

Arabic language spoken by roughly 330 million people[60], is the fifth most spoken language in the world [61],[62]. Arabic Internet material expands amid daily Internet activities [63]. Arabic has 28 characters and

is written in a cursive form akin to Urdu and Farsi. The main features of Arabic language explained below:

- *Shape*: The different character forms that are used to write an Arabic sentence are another characteristic of the Arabic script. Writing in Arabic is done from right to left, and some do not connect to the characters that come after them. An Arabic character's shape in a sentence depends on where it falls within the word. A character can produce one of four shapes when it is isolated, connected to its position in the word either joined from both sides (middle form), from the right (ending form), or connected from the left (beginning form).[64]. Table 2.1 shows a sample of Arabic alphabets and their forms [65].

Table 2.1: Sample of Arabic Alphabets Forms[65]

Name	Unicode	Shapes			
		Isolated	Final	Medial	Initial
HAMZA	0621	ء			
ALEF WITH MADDA ABOVE	0622	أ	آ		
WAW WITH HAMZA ABOVE	0624	ؤ	آؤ		
ALEF WITH HAMZA BELOW	0625	إ			
YEH WITH HAMZA ABOVE	0626	ئ	آئ	أئ	أئ
ALEF	0627	ا	آ		
BEH	0628	ب	ب	ب	ب
TEH MERBUTA	0629	ة	ة		
THE	062A	ت	ت	ت	ت
THEH	062B	ث	ث	ث	ث
JEEM	062C	ج	ج	ج	ج
HAH	062D	ح	ح	ح	ح
KHAH	062E	خ	خ	خ	خ
DAL	062F	د	د		
THAL	0630	ذ	ذ		
RAH	0631	ر	ر		
ZAIN	0632	ز	ز		
SEEN	0633	س	س	س	س
SHEEN	0634	ش	ش	ش	ش
SAD	0635	ص	ص	ص	ص
DHAD	0636	ض	ض	ض	ض

- **Dot:** One, two dots may be positioned above or below certain Arabic characters or three only above. Arabic features fifteen pointed letters, five of which are multipoint, in contrast to English's lack of multipoint letters, see Table 2.2.

*Table 2.2: Dots in Arabic letters*

No. of dots	Arabic letters
0	ا, ح, د, ر, س, ص, ط, ع, ك, ل, م, و, ه, ء, و, ئ
1	ب, ج, خ, ذ, ز, ض, ظ, غ, ف, ن
2	ت, ق, ي
3	ث, ش

- **Diacritics:** Arabic symbols such as FATHAH, DHAMMAH, KASRAH, SUKON, and SHADDAH use diacritical marks, which are short vowels. TANWEEN can alternatively be formed from TWO DHAMMAS, TWO KASRAH, or TWO FATHAH. These diacritical marks can be positioned above or below the characters and are written as strokes. A word's meaning can be altered by altering a character's diacritical mark. Arabic readers are used to inferring meaning from context when reading undiacritical texts, see Table 2.3.

*Table 2.3: Diacritical Marks of Arabic Language*

Name	diacritical marks
FATHAH	◌َ
DHAMMAH	◌ُ
KASRAH	◌ِ
SUKON	◌ْ
SHADDAH	◌ّ
TWO FATHAH	◌ً
TWO DHAMMAS	◌ٌ
TWO KASRAH	◌ٍ

- **Kashida:** The extended character that appears between words to write (—) is another distinctive aspect of Arabic script, referred to as Kashida, such as ‘كتب’ extended letter ‘ك’ to ‘كـ’ [66].



- *Sharp Edges*: Having a lot of sharp edges is another distinctive feature of the Arabic characters found by [67]. The Arabic letters range from one sharp edge (that is, و) to five sharp edges (that is, ك). The sharp letters in Arabic are categorized into five groupings according to the number of edges [68], see Table 2.4.

Table 2.4: The number of sharp edges in Arabic letters [68]

No. sharp edges	Arabic letters
1	و ة ف ه م
2	ا ب ت ث ذ ر ز ي ل ط ظ ن
3	غ ع ء ج ح خ
4	س ش
5	ك

- *Letter Frequency*: The frequency of Arabic letters based on the Holy Quran [69] explained in Table 2.5. As show the letter (ا) is the higher frequency and (ؤ) is the lower.

Table 2.5: Arabic Letter frequency using only the Quran as input source [69]

Rank	Letter	Frequency	Percentage	Rank	Letter	Frequency	Percentage
1	ا	43,542	13.17	19	ذ	4,932	1.49
2	ل	38,191	11.55	20	ح	4,140	1.25
3	ن	27,270	8.25	21	ج	3,317	1.00
4	م	26,735	8.08	22	ى	2,592	0.78
5	و	24,813	7.50	23	خ	2,497	0.76
6	ي	21,973	6.64	24	ة	2,344	0.71
7	ه	14,850	4.49	25	ش	2,124	0.64
8	ر	12,403	3.75	26	ص	2,072	0.63
9	ب	11,491	3.47	27	ض	1,686	0.51
10	ت	10,520	3.18	28	ز	1,599	0.48
11	ك	10,497	3.17	29	ء	1,578	0.48
12	ع	9,405	2.84	30	آ	1,511	0.46
13	أ	9,119	2.76	31	ث	1,414	0.43
14	ف	8,747	2.64	32	ط	1,273	0.38
15	ق	7,034	2.13	33	غ	1,221	0.37
16	س	6,012	1.82	34	ئ	1,182	0.36
17	د	5,991	1.81	35	ظ	853	0.26
18	إ	5,108	1.54	36	ؤ	673	0.20

- *Letter with loop*: A further characteristic that pertains to written Arabic letter with or without loops was also discovered. Notably, there is a loop in nine Arabic letters [64]:

( 'م', 'ص', 'ض', 'ط', 'ظ', 'ف', 'ق', 'و', 'ه' ).

## 2.8 Evaluation Measures

The available coverless text steganography methods are evaluated and contrasted based on a variety of critical factors such as hiding capacity, algorithm efficiency, success rate, ability to resist steganalysis, and theoretical and real-world significance. The important metrics used to evaluate the performance of coverless text steganography methods is described in the following subsections.

### 2.8.1 Hiding Capacity and Embedding Rate

In the subject of information hiding, there are two forms of capacity definition: the hiding capacity and the embedding rate.

- *Hiding capacity*: refers to the maximum amount of data that can be hidden within the cover data using a specific steganographic technique or algorithm. It indicates the upper limit of how much information can be concealed without causing noticeable changes to the cover data. Hiding capacity is usually measured in bits or bytes. A higher hiding capacity implies that more data can be hidden, but it also presents a greater risk of detection and potential alteration of the cover data [12][39][32].

$$Hiding\ Capacity(HC) = \frac{\text{the length of the secret message}}{\text{the total number of the carrier texts}} \quad (2)$$

### 2.8.2 Success Rate

The success rate is a measure employed to evaluate the performance of the embedding algorithm. The success rate can be calculated by equation 3 [39].

$$\text{Success Rate (SR)} = \frac{\text{the number of successfully hidden messages}}{\text{the total number of secret messages}} \quad (3)$$

### 2.8.3 Extracting Accuracy

Evaluating the performance of the extracting algorithm (distance between the secret message and the extracted message). Accuracy can be calculated by equation 4 [36].

$$\alpha = 1 - \frac{D}{L_m} \quad (4)$$

where  $D$  is the minimal number of editing procedures required to change the extracted message, which includes character replacement, into the hidden message, character addition, and character deletion.  $L_m$  is the maximum length possible between the retrieved and secret messages.

### 2.8.4 Security Analysis

Overall, the provided security analysis of the resilience of the proposed coverless text steganography method to various steganalysis techniques. It underscores the method's ability to effectively use natural text as carriers, maintain the original probability distribution, and ensure that the hidden information remains inaccessible to unauthorized parties[35][39].

## 2.8.5 Perplexity

The degree of confusion in natural language processing could be used to assess the quality of a language production model. The closer the statistical distribution of the output text and the training text was, the lower the perplexity. The following formula expresses the perplexity [8-11][9][40] [41] :

$$\text{Perplexity} = 2^{-\frac{1}{m}} \sum_{i=1}^m \log p(s_i) \quad (5)$$

When  $s$  is the generated sentence,  $p(s)$  denotes the probability distribution for words in  $s$ , and probability is derived using the training texts' language model. The parameter  $m$  indicates the total number of words in  $s$ .

## 2.8.6 Availability

A large scale-corpus will prompt the availability of these methods [31]. Table 2.6 presents a summary and comparison of 6 evaluation metrics types used in developing coverless text steganography methods over the selected papers. The table shows that the most common measure used is hiding capacity, then the success rate and security analysis, respectively.

No.	Metrics	References
1	Hiding capacity, Embedding rate	All papers about coverless except[22]
2	Success rate	[39] , [21-27], [16],[14] [18],[42] ,[32] ,[30-34]
3	Extracting accuracy	[18], [39],[30],[33],[36],[14]
4	Security Analysis	[39],[24],[54],[18],[21],[22],[25],[28],[30-33],[14]
5	Perplexity	[7-10] ,[37-38]
6	Availability	[31],[14]

In this thesis, we focus on measuring capacity and perplexity.

**CHAPTER THREE**

**PROPOSED METHODOLOGY**

### **3.1 Overview**

This chapter presents the details of the proposed thesis which involves two methods, the first method is coverless text steganography within the generation method type based on the Arabic language statistical model using the Markov Chain with N-grams. The second method is coverless text steganography within the search method type based on Arabic language built-in features.

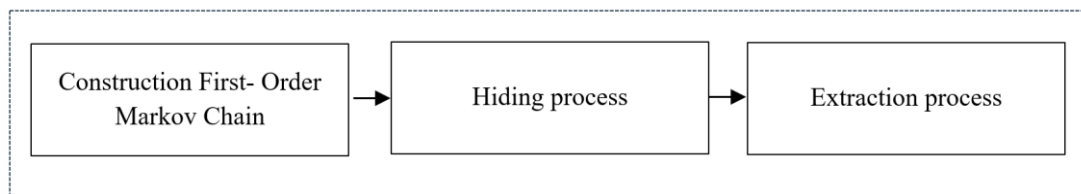
### **3.2 Generation Method**

The idea behind the generation approach is to automatically produce paraphrases, which is a new and practical way to change the language for linguistic steganography.

Coverless text steganography techniques, particularly in Arabic text, present a new approach to embedding information without relying on explicit cover objects.

The proposed method leverages first order Markov Chain model to embed information within the text without the need for a separate cover medium. In a first-order Markov Chain, every following state in the Chain depends only on the state that came before it. The algorithm used the dynamic code word on the embedding process based on the transition probability of the first-order Markov Chain model. The first-order Markov Chain means that implement  $N\text{-Grams}=1$ , which every following state in the Chain depends only on the one state that came before it.

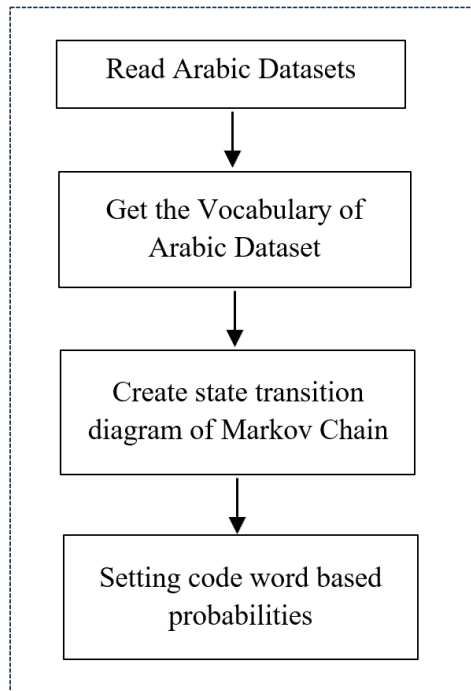
The proposed method begins by selecting an Arabic dataset, then create state transition diagram based on these selected sentences according to the word with the highest frequency and the most branches. A code word is assigned to represent the transitions in the diagram. Using this code word and the original input sequence, a text is generated to conceal the information. The extraction process utilizes the same diagram and selected sentences. Finally, hidden information is extracted and compared with the original dataset to uncover the concealed content. Figure 3.1 shows the steps of the proposed generating method using first-order Markov Chain in detail. The three parts that make up the suggested method are the construction first- order Markov Chain, the hiding process, and the extracting process, as shown in Figure 3.1



*Figure 3.1: Generation Method Using First-Order Markov Chain*

### **3.2.1 Construction First- Order Markov Chain**

In this step, implementing the construction first-order Markov Chain dynamically based on the selected dataset. This step is pre-agreed upon beforehand between the sender and receiver. Figure 3.2 shows the details of this step.



*Figure 3.2: Construction first-order Markov Chain*

- **Read Arabic Dataset step**

In this step, the dataset containing the information to be concealed is selected. This method utilizes three Arabic datasets. A preprocessing step was implemented, which involves tokenization—the process of breaking down sentences into individual words known as tokens. The resulting set of Arabic tokens was then passed to the next step to extract the vocabulary of the Arabic dataset.

- **Get The Vocabulary of Arabic Dataset step**

The vocabulary of the Arabic dataset consists of the unique Arabic words (excluding frequency). Table 3.1 presents the vocabulary size for each dataset.



Table 3.1 Present Vocabulary Size

Dataset	Vocabulary size
SANAD	905,531
APCD	751,409
Arabic Poetry Dataset	537,721

- **Create State Transition Diagram of Markov Chain step**

To calculate the transition probability and create state transition diagram, this method uses a Markov Chain model to calculate the frequency of each word or token in the training set.

Based on the transition probability, encode the words and embed secret information in the generated text. The components of Markov Chain (*states, transition probability, and code word*) are represented as a nested dictionary data structure in Python. Initially, the code words are 0, see the sample text in Figure 3.3.

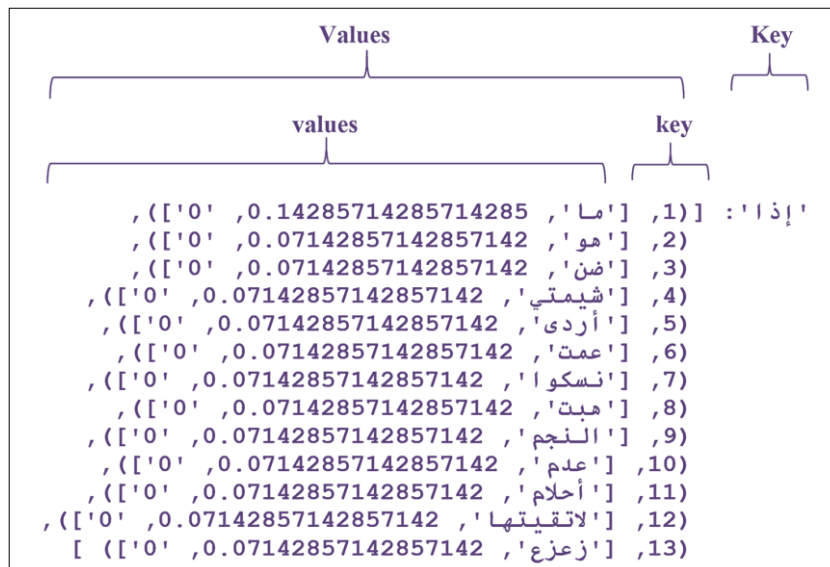


Figure 3.3: Sample of First-Order Markov Chain Data Structure

To understand the Markov Chain data structure, the *Key* represents the current word, while *Values* represent the list of the next words. In terms of the state transition diagram of the Markov Chain, the components are (*states*, *transition probability*, and *code word*). In Figure 3.3, the length of the list of next words means the number of branches in the diagram, in this example, there are 13 branches.

- **Setting Code Word Based Probabilities step**

This step involves two sub-steps:

- Arrange the words in descending order according to their branches, as well as arrange the list of the next words in descending order based on their probabilities.
- Set the code word based on number of branches ( $n$ ) using the following equation:

$$code\_word[i] = \begin{cases} binary(i, \lfloor \log_2 n \rfloor) & i < 2^{\lfloor \log_2 n \rfloor} \\ '0' & i \geq 2^{\lfloor \log_2 n \rfloor} \end{cases} \quad (6)$$

Where  $binary(i, \lfloor \log_2 n \rfloor)$  denotes the binary representation of the number  $i$ , and  $i$  range from 0 to  $n - 1$ .

The length of code words depends on the number of branches, for example,  $n=8$ , which means the length of code words=3 using equation 6, they are ['000', '001', '010', '011', '100', '101', '110', '111'], show Figure 3.4 and Figure 3.5.

Values	Key
values	key
<pre> 'إذا': [   ([ '000' , 0.14285714285714285 , 'ما' ] , 1) ,   ([ '001' , 0.07142857142857142 , 'هو' ] , 2) ,   ([ '010' , 0.07142857142857142 , 'ضن' ] , 3) ,   ([ '011' , 0.07142857142857142 , 'شيمتي' ] , 4) ,   ([ '100' , 0.07142857142857142 , 'أردى' ] , 5) ,   ([ '101' , 0.07142857142857142 , 'عمت' ] , 6) ,   ([ '110' , 0.07142857142857142 , 'نسكوا' ] , 7) ,   ([ '111' , 0.07142857142857142 , 'هبت' ] , 8) ,   ([ '0' , 0.07142857142857142 , 'النجم' ] , 9) ,   ([ '0' , 0.07142857142857142 , 'عدم' ] , 10) ,   ([ '0' , 0.07142857142857142 , 'أحلام' ] , 11) ,   ([ '0' , 0.07142857142857142 , 'لاتقيتها' ] , 12) ,   [ ([ '0' , 0.07142857142857142 , 'زعزع' ] , 13) </pre>	

Figure 3.4: Setting of Code words in First-Order Markov Chain Data Structure

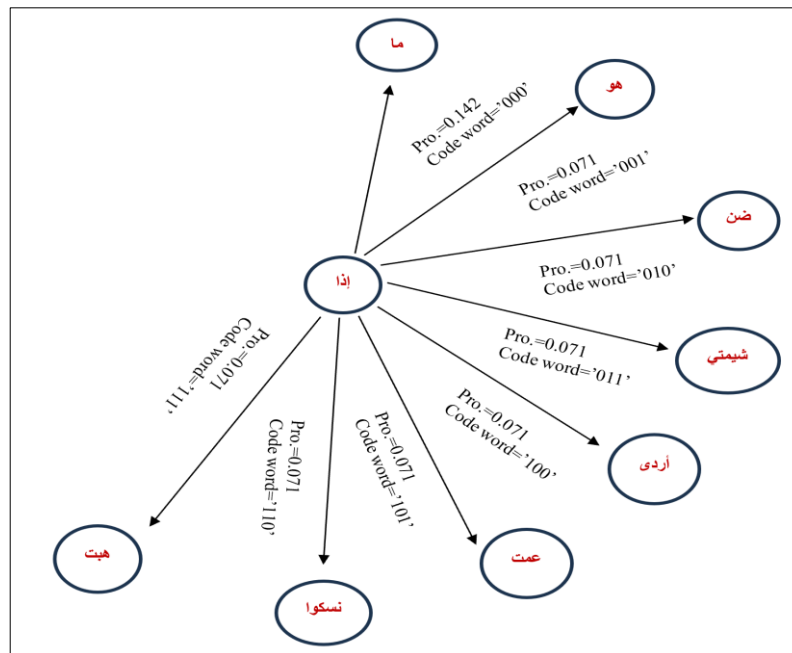


Figure 3.5. Sub Markov Diagram for Word "إذا"

### 3.2.2 Hiding Processes of Generation Method

The hiding procedure was started by tokenization the secret message. Each character in a secret message token was converted to the binary. Then, using pre-agreed construction first-order Markov Chain was dynamically based on the selected dataset. The successful matching with the code word was implemented to generate the stego-text that was sent to the receiver. Figure 3.6 shows the hiding procedures.

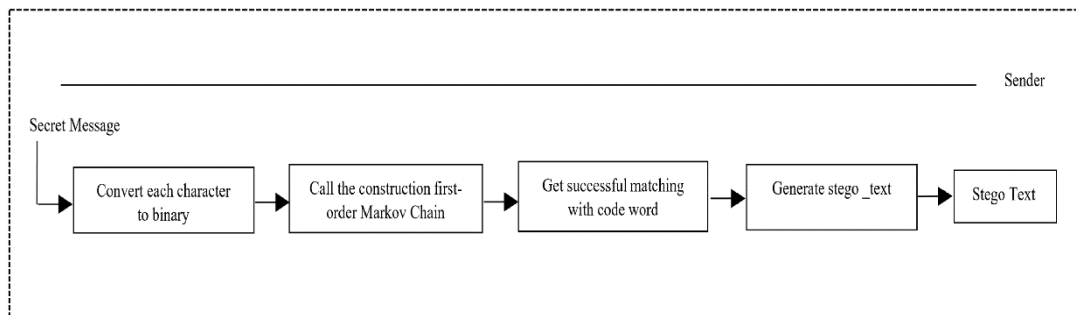


Figure 3.6: Hiding Processes of Generation Method

#### Algorithm (3.1) Hiding Processes of Generation Method

**Input:** Secret message M

**Output:** Stego- text

**Step 1:** Convert each character in the secret message M into binary.

**Step 2:** Call construction first- order Markov Chain

**Step 3:** Match each binary code of the secret message with the corresponding code word from the Markov Chain to generate the stego-text

**Step 4:** Return stego-text.

**End.**

Example: assume the secret message = '010000', by tracing the secret message with a code word in first-order Markov Chain data structure and

state transition diagram, the stego-text is “إذا ضن ذو القربى”, see Figures 3.7, 3.8

```

,(['000',0.14285714285714285,'ما'],1),
,(['001',0.07142857142857142,'هو'],2),
,(['010',0.07142857142857142,'ضن'],3),
,(['011',0.07142857142857142,'شيمتي'],4),
,(['100',0.07142857142857142,'أردى'],5),
,(['101',0.07142857142857142,'عمت'],6),
,(['110',0.07142857142857142,'نسكوا'],7),
,(['111',0.07142857142857142,'هبت'],8),
,(['0',0.07142857142857142,'النجم'],9),
,(['0',0.07142857142857142,'عدم'],10),
,(['0',0.07142857142857142,'أحلام'],11),
,['0',0.07142857142857142,'لاتقيتها'],12),
[(['0',0.07142857142857142,'ززع'],13)

,{'0',1.0,'ذو']:1} : 'ضن'
,{'0',1.0,'القربى']:1} : 'ذو'
,{'0',1.0,'عليهم']:1} : 'القربى'

```

Figure 3.7: Matching Secret Message with Code Words in First-Order Markov Chain Data Structure

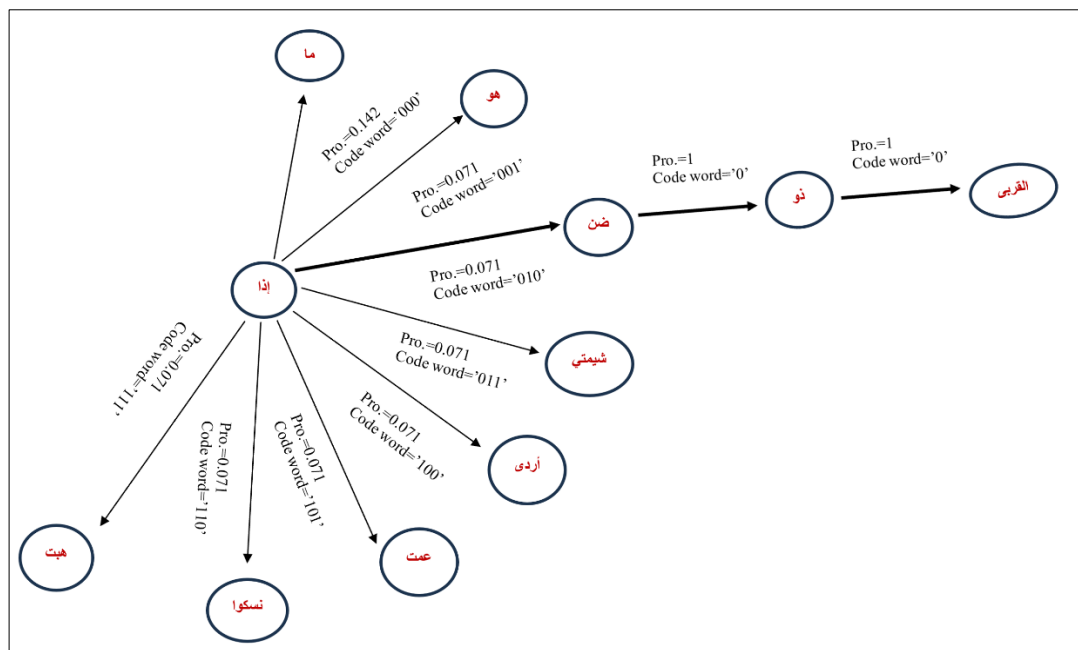


Figure 3.8: Matching Secret Message with Code Words in State Transition Diagram

### 3.2.3 Extraction Processes of Generation Method

Conversely, the extraction process for uncovering the hidden information involves reversing the initial hiding procedure. This is achieved by reconstructing the first-order Markov Chain and then retrieving the corresponding code words from the state transition diagram. These extracted code words are subsequently converted back into their respective characters. Figure 3.9 shows the extraction procedures.

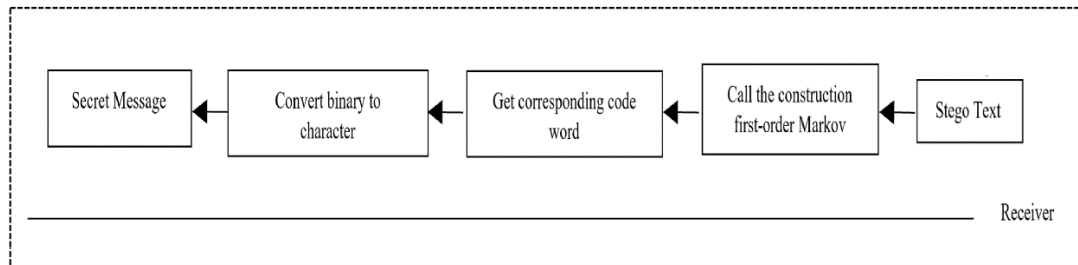


Figure 3.9: Extraction Processes of Generation Method

*Example:* For instance, consider the stego-text 'إذا ضن ذو القربى'. By tracing this text through the state transition diagram and the first-order Markov Chain data structure, we can extract the corresponding secret message. In this case, the secret message '010000' is successfully retrieved, as illustrated in Figures 3.7 and 3.8.

#### Algorithm (3.2) Extraction Processes of Generation Method

**Input:** Stego-text

**Output:** Secret message M'

**Step1:** Receive the stego-text

**Step2:** Call construction first- order Markov Chain

**Step3:** Match Code Words: Trace the stego-text using the state transition diagram to retrieve the corresponding code words.

**Step4:** Convert the binary code words back to the original characters to reveal the secret message.

**Step5:** Return the secret message M'.

**End.**

### **3.3 Search Method**

The principle of the search method is to locate accessible carriers from the dataset that match the statistical characteristics of the secret message, assuming a suitable carrier dataset will be identified.

The Arabic language features offer a new method for data concealment. A new coverless text steganography method was proposed based on built-in features of Arabic scripts. The first word of each row in the dataset is tested based on eight features to get one byte containing 1 or 0. That is a result of the presence or absence of the following features: mahmoze, diacritics, isolated, two sharp edges, vowels, dotted, looping, and high frequency. Then, each byte is converted to a decimal number (ASCII code) to implement a dynamic mapping protocol with the most frequent letter.

In the hiding process, each character in the secret message is converted to ASCII code and successfully matched in the dataset. Thus, after matching, the candidate text is sent to the receiver. In contrast, the pre-agreed dynamic mapping protocol was implemented in a receiver to extract secret messages.

This section provides a detailed explanation of the proposed coverless text steganography within the search method type based on Arabic language features. The three parts that make up the suggested method are the dynamic mapping protocol, the hiding procedure, and the extracting procedure, as shown in Figure 3.10

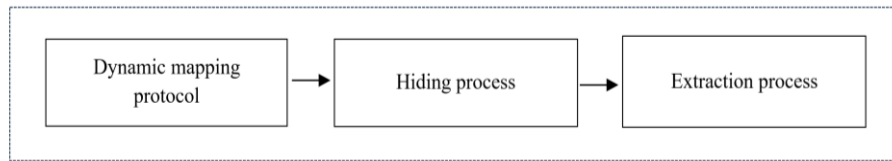


Figure 3.10: Search method based on Arabic language features

### 3.3.1 Dynamic Mapping Protocol

In this part, implementing the dynamic mapping protocol is the pre-agreed protocol upon beforehand between the sender and receiver. Figure 3.11 shows the details of this part.

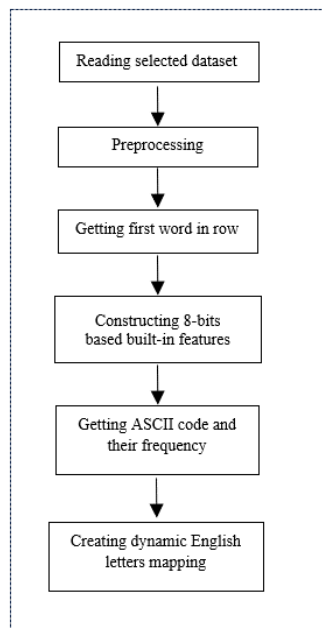


Figure 3.11: Dynamic mapping protocol

- **Reading Selected Dataset Step**

In this step the dataset is read for preparing the cover for embedding the secret message. Three Arabic datasets are used in this thesis (SANAD (Single-Label Arabic News Articles Dataset) includes 45500 articles,



Arabic Poem Comprehensive Dataset (APCD) contains 1,831,770 poetic verses in total, Arabic Poetry Dataset contains more than 58000 poems).

- **Preprocessing Step**

Many processes in datasets were implemented in the Arabic news articles dataset, including 7 folders in multiple topics: culture, finance, medicine, politics, religion, sports, and tech. All text files in 7 folders were converted to CSV files. Meanwhile, the preprocessing in the other two datasets involved extracting just poetic verses from total poem metadata.

- **Get First Word in Row Step**

The proposed method is implemented on the first word of each row in the selected dataset. In SANAD, 45,500 words were extracted, while in the APCD and Arabic poetry dataset, 1,831,770 and 58021 words were extracted, respectively.

- **Constructing 8-Bit Based Built-In Features Step**

In this step, the resulting words from the previous step were tested based on eight features to construct one byte containing 1 or 0 resulting from the presence or absence of these features. The features include: mahmoze, diacritics, isolated, two sharp edges, vowels, dotted, looping, and high frequency. Six features are implemented on the first character of the word (isolated, two sharp edges, vowels, dotted, looping, and high frequency), while the other features (mahmoze, diacritics) are implemented on the end character and whole word, respectively. The following functions explain built-in features in detail:

1. **Is\_mahmoze:** This function checks if a word ends with a mahmoze letter; mahmoze letters mean that they contain HAMZA [‘أ’, ‘ؤ’, ‘ئ’, ‘ء’].
2. **Has\_diacritics:** This function checks if a word has diacritics [‘َ’, ‘ُ’, ‘ِ’, ‘ّ’, ‘ٍ’, ‘ٌ’, ‘ٍ’, ‘ٌ’, ‘ٍ’, ‘-’].
3. **Is\_isolated:** It checks if a word starts with isolated letters, that is, cannot connected with the next one [‘أ’, ‘ر’, ‘ذ’, ‘د’, ‘ز’, ‘و’, ‘ؤ’].
4. **Is\_2\_sharp edges:** It checks if a word starts with 2\_sharp edges letters [‘ن’, ‘ظ’, ‘ط’, ‘ل’, ‘ي’, ‘ز’, ‘ذ’, ‘ر’, ‘ا’, ‘ت’, ‘ث’, ‘د’, ‘ب’, ‘ا’].
5. **Is\_vowels:** It checks if a word starts with vowel letters [‘أ’, ‘إ’, ‘آ’, ‘و’, ‘ي’, ‘ؤ’].
6. **Is\_dotted:** It checks if a word starts with dotted letters: [‘ب’, ‘ت’, ‘ث’, ‘ج’, ‘خ’, ‘ذ’, ‘ز’, ‘ش’, ‘ظ’, ‘ض’, ‘غ’, ‘ق’, ‘ف’, ‘ن’, ‘ي’].
7. **Is\_looping:** It checks if a word starts with looping letters: [‘ؤ’, ‘ة’, ‘ص’, ‘ض’, ‘ط’, ‘ظ’, ‘ف’, ‘ق’, ‘و’, ‘ه’].
8. **Is\_high-frequency:** It checks if a word starts with high-frequency letters, In the proposed method, seven high-frequency letters were selected [‘أ’, ‘ل’, ‘ن’, ‘م’, ‘ي’, ‘و’, ‘ه’].

Table 3.2 explains examples of Arabic words and their 8-bit based built-in features.

Table 3.2: *Example of Construct 8-bit Based Built-in Features*

Words	Built-in features							
	mahmoze	diacritics	isolated	2 sharp edges	vowels	dotted	looping	high
استضافت	0	0	1	1	1	0	0	1
اليغسوب	0	1	1	1	1	0	0	1
زار	0	0	1	1	0	1	0	0

- **Getting ASCII Code and Their Frequency Step**

In this step, the resulting 8 bits are converted to decimal numbers (ASCII code) and find their frequency in the Arabic news dataset. The distribution of ASCII code in the selected dataset is very significant in hiding capacity. Table 3.3 shows the ASCII code and their frequency in the dataset, while Figure 3.12 shows the distribution.

*Table 3.3: ASCII Code and their Frequency in SANAD Dataset*

No.	Code	Frequency	No.	Code	Frequency
1	8	10961	24	121	52
2	57	6691	25	85	43
3	48	5762	26	93	40
4	20	4608	27	132	37
5	0	4427	28	185	30
6	6	2769	29	72	29
7	29	2013	30	68	27
8	4	1597	31	65	26
9	1	1312	32	157	25
10	43	940	33	136	25
11	21	936	34	70	23
12	18	714	35	112	19
13	17	701	36	129	18
14	36	370	37	134	17
15	3	253	38	131	15
16	22	227	39	149	9
17	64	172	40	67	9
18	52	138	41	82	7
19	148	135	42	81	4
20	84	123	43	176	4
21	146	68	44	145	4
22	107	63	45	100	2
23	128	53	56	195	1

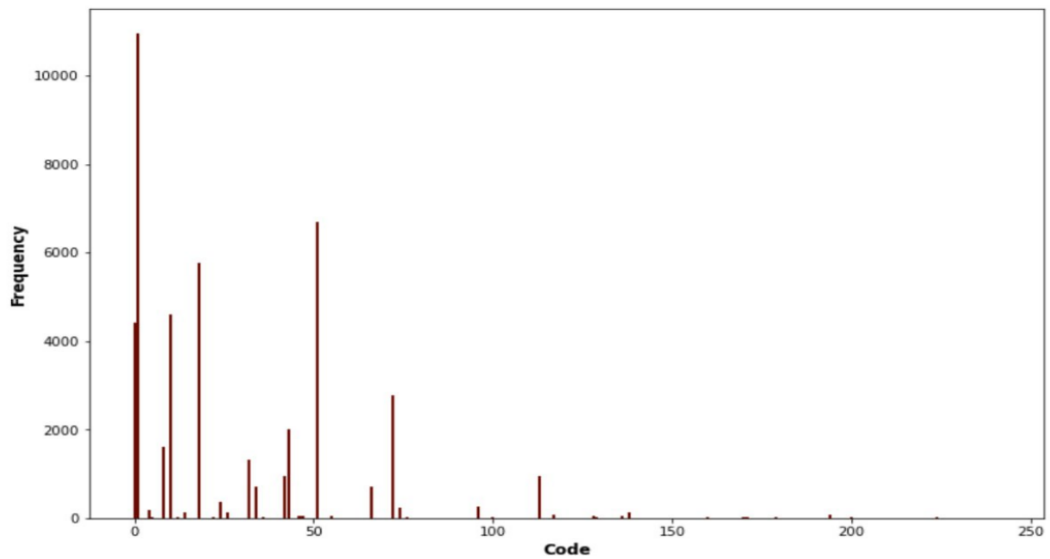


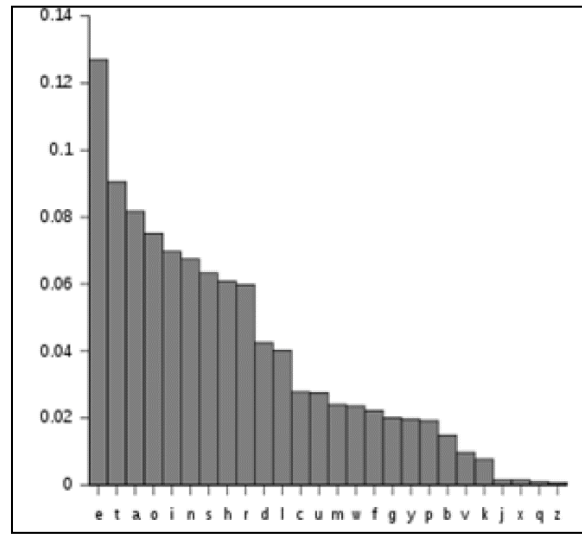
Figure 3.12: Distribution of ASCII Code in SANAD Dataset

- **Creating Dynamic English Letter Mapping Step**

As shown in Figure 3.11, the distribution of ASCII code lies in the out range of English letters in the ASCII code. This required making a mapping to ensure that the English message was successfully hidden in the selected dataset.

The proposed method implemented dynamic English letter mapping to map the most frequent ASCII code with the most frequent English letters. The term “dynamic” denotes that mapping is changed dynamically based on the selected dataset itself. Figure 3.12 shows the frequency of English letters [12].

Letter	Frequency	Letter	Frequency
e	12.7020%	m	2.4060%
t	9.0560%	w	2.3600%
a	8.1670%	f	2.2280%
o	7.5070%	g	2.0150%
i	6.9660%	y	1.9740%
n	6.7490%	p	1.9290%
s	6.3270%	b	1.4920%
h	6.0940%	v	0.9780%
r	5.9870%	k	0.7720%
d	4.2530%	j	0.1530%
l	4.0250%	x	0.1500%
c	2.7820%	q	0.0950%
u	2.7580%	z	0.0740%



(a)

(b)

Figure 3.13: The English letters frequency (a): tabular data ;(b) histogram [70].

The results of dynamic English letter mapping with the most frequent ASCII code in the selected news datasets and two datasets about Arabic poetry are explicated in Table 3.4.

Table 3.4: Dynamic English Letter Mapping Results: ASCII Frequencies in Three Datasets

SANAD				APCD				Arabic Poetry Dataset			
ASCII code	Freq.	English letters	Mapping	ASCII code	Freq.	English letters	Mapping	ASCII code	Freq.	English letters	Mapping
8	10,961	e	(8:'e')	43	485,718	e	(43:'e')	57	14,620	e	(57:'e')
57	6,691	t	(57:'t')	6	252,772	t	(6:'t')	0	8,086	t	(0:'t')
48	5,762	a	(48:'a')	8	226,772	a	(8:'a')	17	4,913	a	(17:'a')
20	4,608	o	(20:'o')	0	167,059	o	(0:'o')	29	4,639	o	(29:'o')
0	4,427	i	(0:'i')	20	125,386	i	(20:'i')	43	4,331	i	(43:'i')
6	2,769	n	(6:'n')	17	120,149	n	(17:'n')	20	4,018	n	(20:'n')
29	2,013	s	(29:'s')	1	104,239	s	(1:'s')	6	3,942	s	(6:'s')
4	1,597	h	(4:'h')	29	99,979	h	(29:'h')	1	3,718	h	(1:'h')
1	1,312	r	(1:'r')	4	55,470	r	(4:'r')	4	2,409	r	(4:'r')
43	940	d	(43:'d')	3	39,046	d	(3:'d')	48	1,833	d	(48:'d')
21	936	l	(21:'l')	48	36,294	l	(48:'l')	3	1,689	l	(3:'l')
18	714	c	(18:'c')	57	34,384	c	(57:'c')	18	1,104	c	(18:'c')
17	701	u	(17:'u')	21	26,130	u	(21:'u')	21	1,075	u	(21:'u')
36	370	m	(36:'m')	18	21,351	m	(18:'m')	36	302	m	(36:'m')
3	253	w	(3:'w')	36	8,892	w	(36:'w')	52	299	w	(52:'w')
22	227	f	(22:'f')	22	7,103	f	(22:'f')	22	272	f	(22:'f')
64	172	g	(64:'g')	52	5,530	g	(52:'g')	185	132	g	(185:'g')

52	138	y	(52:'y')	171	4,618	y	(171:'y')	132	83	y	(132:'y')
148	135	p	(148:'p')	128	1,563	p	(128:'p')	171	73	p	(171:'p')
84	123	b	(84:'b')	134	1,525	b	(134:'b')	128	65	b	(128:'b')
146	68	v	(146:'v')	148	1,511	v	(148:'v')	148	48	v	(148:'v')
107	63	k	(107:'k')	132	1,426	k	(132:'k')	129	47	k	(129:'k')
128	53	j	(128:'j')	136	1,034	j	(136:'j')	145	24	j	(145:'j')
121	52	x	(121:'x')	176	694	x	(176:'x')	134	19	x	(134:'x')
85	43	q	(85:'q')	129	617	q	(129:'q')	131	15	q	(131:'q')
93	40	z	(93:'z')	157	453	z	(157:'z')	176	14	z	(176:'z')

### 3.3.2 Hiding Processes of Search Method

On the sender side, the hiding procedure was started by preprocessing the secret message by removing special characters, converting them to lowercase, converting numbers to words, and tokenizing. Each character in a secret message token was converted to the corresponding ASCII code. Then, using pre-agreed dynamic English letters mapping. This protocol maps the most frequent ASCII code with the most frequent English letters. The successful matching was collected as the stego-text sent to the receiver. Figure 3.13 shows the hiding procedures.

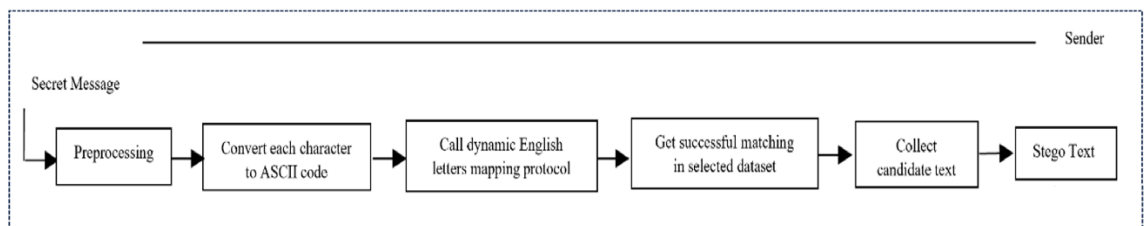


Figure 3.14: Hiding Procedures of Search Method

*Example:* assume the secret message = ‘hope’, Figure 3.15,3.16, and 3.17 explain the details of the hiding procedure in three datasets.

Secret Message	ASCII Code	Mapping	Matching in dataset	Built-in Features (Binary)	Built-in Features (ASCII)	Collected stego-text
0	h	104 (4: h)	خصصت	00000100	4	...خصصت مجلة تراث الصائره عن نادي تراث الإمارات م
1	o	111 (20: o)	تقيم	00010100	20	...تقيم القاديه لينا كابلوت معرضاً في مطلع العام
2	p	112 (148: p)	بدأ	10010100	148	...بدأ التشكيلي الإماراتي إبراهيم العوضي تجربته ا
3	e	101 (8: e)	أصدر	00001000	8	...أصدر قسم الدراسات والبشر في دائرة الثقافة والإ

Figure 3.15: Hiding procedure results using SANAD dataset

Secret Message	ASCII Code	Mapping	Matching in dataset	Built-in Features (Binary)	Built-in Features (ASCII)	Collected stego-text
0	h	104 (29: h)	يتوب	00011101	29	يتوب إليها كل صيف وجانب كما رد ددهاه الفاض نصيحها
1	o	111 (0: o)	على	00000000	0	على غير ذنب أن أكون جنيته سوى قول باع كادني فنجبدا
2	p	112 (128: p)	عزاء	10000000	128	عزاء لم تجل الخطاب بهجتها حتى اجتلاها جنادي بنينار
3	e	101 (43: e)	وإن	00101011	43	وإن تنظراني اليوم أفض لباده وتسوجبا منا على وتحندا

Figure 3.16: Hiding procedure results using APCD dataset

Secret Message	ASCII Code	Mapping	Matching in dataset	Built-in Features (Binary)	Built-in Features (ASCII)	Collected stego-text
0	h	104 (1: h)	منظرها	00000001	1	...منظرها امام بابك الكبير اصرخ في الضلام استجير
1	o	111 (29: o)	يا	00011101	29	...يا ضياء الحقول ياغوده الفلاح في الساجيلت من اس
2	p	112 (171: p)	ولجء	10101011	171	...ولجء فمرا لحياه الدلس فرحل مثلما تاتي ويبقى ال
3	e	101 (57: e)	انا	00111001	57	...انا لا ازال و في يدي كبحي ياليل اين تفرق الشر

Figure 3.17: Hiding procedure results using Arabic Poetry dataset

As seen in Figures 3.15, 3.16, and 3.17 the mapping protocol was dynamic based on the selected dataset, the candidate text was different from one dataset to another.

#### Algorithm (3.3) Hiding Processes in Search Method

**Input:** Secret message M

**Output:** Stego-text

**Step 1:** preprocessing to the secret message.

**Step 2:** Iterate over each character in the secret message and convert to ASCII code

**Step 3:** Call dynamic English letters mapping

**Step 3:** Get successful matching in selected dataset

- a. If a match is found:
- i. Select the matching text from the dataset.
- ii. Append text to the Stego-text.

**Step 4:** Return the Stego-text.

**End.**

### 3.3.3 Extraction Processes of Search Method

On the receiver's side, the extraction process is the reverse of the embedding procedure. It begins with the stego-text, where the pre-agreed dynamic English letters mapping is applied. This mapping is then used to identify and extract the corresponding characters, ultimately reconstructing the entire secret message. The detailed extraction steps are illustrated in Figure 3.18.

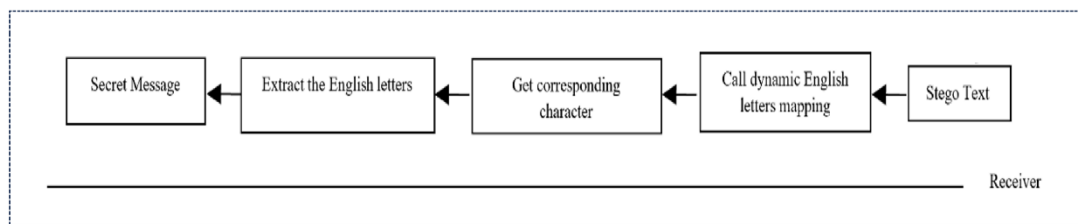


Figure 3.18 Extraction Procedure of Search Method

*Algorithm (3.4) Extraction processes in search method*

**Input:** Stego-text

**Output:** Secret Message M'

**Step 1:** Receive the stego-text.

**Step 2:** Call dynamic English letters mapping protocol.

**Step 3:** If a match is found:

- i. Extract the corresponding part of the secret message from mapping English letter.
- ii. Append characters to the secret message M'.

**Step 4:** Return the secret message M'.

**End.**



**CHAPTER FOUR**  
**RESULTS AND DISCUSSION**

## 4.1 Overview

In this chapter, the experimental results are given and interpreted for two proposed methods that are presented in this thesis.

In this thesis, three databases (SANAD, APCD, and Arabic Poetry Dataset) were built from Kaggle, totaling about 829.61 MB. Unfortunately, there is no researches on coverless text steganography in Arabic to compare with them, but this thesis conducts several experiments to assess the performance of the proposed steganographic method in terms of hiding capacity, perplexity, success rate, extracting accuracy, security analysis, availability, and validity of the algorithm, respectively.

Three experiments are executed to evaluate the proposed hiding methods with different secret message lengths.

The results of these experiments using two proposed methods were explained in detail in the next subsections.

## 4.2 Arabic Datasets

Three Arabic datasets are used in this thesis:

- SANAD (Single-Label Arabic News Articles Dataset), the articles were collected from three popular news websites: AlKhaleej, AlArabiya, and Akhbarona. SANAD includes (45500 articles in 7 categories) a wide range of topics, culture, finance, medical, politics, religion, sports, and tech[71].

- Arabic Poem Comprehensive Dataset (APCD), the Arabic dataset is primarily scraped from Al-diywan and Al-mawsuea Al-shieria. Once both are combined, there are 1,831,770 poetic verses in total[72].
- Arabic Poetry Dataset contains more than 58K poems in the dataset dating from the sixth to the current era. In addition, the poem metadata includes the poet’s name, the poem’s title, and its category. The source of the extracted dataset was adab.com[73], and two datasets about Arabic poetry were used from Kaggle. More details are explained in Table 4.1.

*Table 4.1: Summary of Selected Arabic Datasets*

<b>Dataset</b>	<b>No. of rows</b>	<b>No. of words</b>	<b>Size /MB</b>	<b>Max char./row</b>
SANAD	45,500	16,800,368	180.81	33,359
APCD	1,831,770	16,967,324	554.12	128
Arabic Poetry Dataset	58021	8,518,526	94.68	47,033

### **4.3 Generation Method Results**

In this method, the secret message was hidden as bit level, that mean at each time one or more bits can be hidden in specific text based on the probability distribution of current words. The length of the code word and the number of branches in the state diagram were limited by the equation explained previously.

Three experiments are executed for the evaluation of the proposed generation method. Embedding English text with sizes 32, 64, and 320 bits.

The results of these experiments using the proposed generation method were explained in detail in the next subsections. The most important metrics used in this method are:

### 4.3.1 Perplexity

Two important metrics was used in this method, perplexity is a typical metric used in natural language processing to rate language models; the lower the language model, the greater the perplexity number. This thesis used equation (5) explained in Chapter 2 to compute the perplexity.

### 4.3.2 Hiding Capacity

While the second metric is hiding capacity which explains the quantity of data that is embedded in text. The proposed method can hide more bits depending on the number of branches. This thesis used equation (2) explained in Chapter 2 to compute the hiding capacity. The idea is “How many bits are hidden in generated stego-text with N words?”.

The perplexity and the hiding capacity are calculated, and the results are shown in Table 4.2 -Table 4.5,

*Table 4.2 Experimental Results of Generation Method with Experiment (1)*

Metrics	Arabic Dataset		
	SANAD	APCD	Arabic Poetry Dataset
Perplexity	10.8	30	5.53
Capacity (bits/stego-text)	4.571	6.4	5.33
No. of words in stego-text	7	5	6

*Table 4.3 Experimental Results of Generation Method with Experiment (2)*

Metrics	Arabic Dataset		
	SANAD	APCD	Arabic Poetry Dataset
Perplexity	22.7	5.95	15
Capacity (bits/stego-text)	4.571	6.4	5.81
No. of words in stego-text	14	10	11

*Table 4.4 Experimental Results of Generation Method with Experiment (3)*

Metrics	Arabic Dataset		
	SANAD	APCD	Arabic Poetry Dataset
Perplexity	15.5	30.4	30.9
Capacity (bits/stego-text)	4	6.8	5.71
No. of words in stego-text	80	47	56

The average of perplexity and the hiding capacity are calculated and explained in Table 4.5

*Table 4.5 Average of Perplexity, Hiding Capacity, and Stego-Text Length of Proposed Generation Method*

Metrics	Arabic Dataset			Average
	SANAD	APCD	Arabic Poetry Dataset	
Perplexity	16.3	22.11	17.143	<b>18.51</b>
Capacity (bits/stego-text)	4.38	6.53	5.61	<b>5.5</b>
No. of words in stego-text	33.666	20.666	24.333	<b>37.443</b>

Compared to generation methods with English datasets, the results explained in Table 4.6

Table 4.6: Comparison of Proposed Generation Method with Related Thesis

Methods	Perplexity	Hiding Capacity
[8]	15.38±6.77, 17.05±15.21	2.73
[9]	15.29, 16.91	2.75
[10]	15.97±7.57 and 17.41±8.91	2.78
[40]	52.05±35.80, 20.52±13.98	2.85
[11]	-----	1.82 without compression
[41]	14.07 ± 8.83, 13.34 ± 9.90, 12.89 ± 8.75	2.71
[12]	16.78 ± 7.89, 14.97 ± 2.55, 25.92 ± 18.59	2.95
proposed method	18.51	5.5

### 4.3.3 Availability

The size of the dataset significantly affects the method's availability as well as the frequency of dataset vocabulary. The number of branching affects the code word length which means the number of hidden bits. The proposed method can be modified to include using second-order or higher Markov Chains, which rely on multiple past states to determine the next state.

### 4.4 Search Method Results

In this method, the secret message was hidden as *character level*, that mean at each time one character could be hidden in a specific text. Four experiments are executed for evaluation of the proposed search method.

- *Experiment (1):* Embedding English text with a size less than 5 char. The secret message =” say”, length = 3 char.
- *Experiment (2):* Embedding English text with a size less than 15 char.. The secret message=” good friends”, length =11 char.
- *Experiment (3):* Embedding English text with a size less than 200 char. The secret message=” say i seek refuge in the lord of daybreak from the evil of that which he created and from the evil of darkness when it settles and from the evil of the blowers in knots and from the evil of an envier when he envies”, length =170 char.
- *Experiment (4):* Embedding English text with a size less than 350 char. The secret message =” computer science is the study of algorithms data structures and computational systems it encompasses a wide range of topics including programming languages software engineering artificial intelligence and computer hardware from developing new algorithms to designing user interfaces computer scientists play a crucial role in shaping our digital world”, length=304 char.

The results of these experiments using the proposed search method are explained in detail in the next subsections.

#### **4.4.1 Hiding Capacity**

Hiding capacity refers to the number of characters or keywords that can be hidden in a single text. The proposed approach conceals characters/words. Furthermore, throughout the embedding procedure, the stego-texts are not required to incorporate the secret message's length. Thus, it is helpful in improving hiding capacity. Many parameters determine the hiding capacity in the proposed method, such as secret

message length, number of successful matchings, their frequencies, and the size of the selected dataset.

For example, the decimal number 8 results in 10,961 matchings, which means it can hide the English letter (e) 10,961 times, see Table 4.7.

*Table 4.7 Average of Hiding Capacity for Proposed Search Method*

Experiments	Hiding Capacity			Average
	Arabic Dataset			
	SANAD	APCD	Arabic Poetry Dataset	
1	0.15	0.21	0.4	0.25
2	0.22	0.28	0.26	0.25
3	0.23	0.25	0.26	0.24
4	0.24	0.24	0.26	0.24
	<b>0.21</b>	<b>0.24</b>	<b>0.29</b>	<b>0.246</b>

#### 4.4.2 Availability

The dataset size has an important impact on the method's availability. The proposed method can be extended to other built-in features such as the part of speech POS, n-grams, Name Entity Recognition NER, and so on.

Table 4.8 summarizes the comparison of the proposed steganographic method in terms of hiding capacity, success rate, extracting accuracy, security analysis, and availability of the proposed search method with related thesis.



Table 4.8: Comparison of Proposed Search Method with Related Works

Methods	Hiding Capacity	Success Rate	Extracting Accuracy	Security Analysis	Availability
[2]	Low capacity, 14 or 15 bits in 200,000 texts			Resisting current detecting techniques, no modification in cover.	
[23]	High capacity 6.25% in best case and reached 25% in the improved method	High success rate		Resisting the format transformation attack	
[34]		Improve success rate	enhances extraction accuracy by increasing the number of keywords	withstands current steganalysis techniques	
[8]	high capacity	100%			
Our proposed method	Char./Word	100%	100%	Stronger defense against format conversion attack and statistical detection.	Can be extended to other built-in features

As shown from above tables, this method, a new coverless text steganography method based on Arabic script's built-in properties has been proposed. The process involved extracting the first word in each row, which was tested based on eight features to get one byte converted to a decimal number (ASCII code). Then, a dynamic mapping protocol was implemented with the most frequently used letter. The proposed method withstands existing detection methods because it involves no modification or generation. Also, there is an enhancement in hiding capacity, which can conceal a (character /word).

The suggested approach, despite being influenced by factors like dataset size, matches, secret message length, and frequency, outperforms other studies in terms of extraction accuracy, availability, security, and concealment success rate.

**CHAPTER FIVE**  
**CONCLUSION AND FUTURE THESIS**

## 5.1 Conclusions

The experiments detailed in this thesis have led to several conclusions, including:

1. In proposed search method there is an enhancement in hiding capacity, which can conceal a (character /word) because tested based on eight features to get one byte which converted to a decimal number (ASCII code). Then, a dynamic mapping protocol was implemented with the most frequently used letter.
2. A potential limitation of the proposed methods including
  - The size of the chosen dataset, the number of successful matches, the length of the secret message, and its frequencies affect the hiding ability and success rate. Despite limitations the proposed method has a high hiding success rate, high security, high availability, and excellent extraction accuracy compared to previous thesis.
  - In proposed generation method, the number of branching affects the code word length which represents the number of hidden bits.
3. The results showed that the first-order Markov method improves the concealment capacity to reach 5.5 and reduces perplexity to 18.51. Regarding the use of Arabic language features, it achieved high accuracy by 100% and achieved a high success rate of 100% with a concealment capacity of up to 0.246.

## 5.2 Future works

Here are some suggestions for future thesis on this topic:

1. Using other methods to generate texts, such as deep learning, Large Language Model (LLM).
2. Studying the ability of investigate the use of high-order Markov model in convert information concealing.
3. Studying the improve hiding capacity by increasing the number of selected features to 16, aiming to conceal 2 bytes.
4. Embedding an Arabic message utilizing 11 bits to encode Arabic letters.
5. Studying the ability of applying the proposed methods with other languages, such as English, French, etc.

## REFERENCES

- [1] M. K. Khan, "Research advances in data hiding for multimedia security," *Multimedia Tools and Applications*, vol. 52, no. 2–3, pp. 257–261, Apr. 2011. doi: 10.1007/s11042-011-0741-1.
- [2] K. Rabah, "Steganography-The Art of Hiding Data," *Inf. Technol. J.*, vol. 3, no. 3, pp. 245–269, 2004, doi: 10.3923/itj.2004.245.269.
- [3] M. Douglas, K. Bailey, M. Leeney, and K. Curran, "An overview of steganography techniques applied to the protection of biometric data," *Multimed. Tools Appl.*, vol. 77, no. 13, pp. 17333–17373, 2018, doi: 10.1007/s11042-017-5308-3.
- [4] M. Mahajan and N. Kaur, "Adaptive Steganography: A survey of Recent Statistical Aware Steganography Techniques," *Int. J. Comput. Netw. Inf. Secur.*, vol. 4, no. 10, pp. 76–92, 2012, doi: 10.5815/ijcnis.2012.10.08.
- [5] S. S. Baawi, D. A. Nasrawi, and L. T. Abdulameer, "Improvement of 'text steganography based on unicode of characters in multilingual' by custom font with special properties," in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, Jul. 2020. doi: 10.1088/1757-899X/870/1/012125.
- [6] S. Shi, Y. Qi, and Y. Huang, "An Approach to Text Steganography Based on Search in Internet," 2016, doi: 10.1109/ICS.2016.51.
- [7] S. Ali, "A State-of-the-Art Survey of Coverless Text Information Hiding," *Int. J. Comput. Netw. Inf. Secur.*, vol. 10, no. 7, pp. 52–58, Jul. 2018, doi: 10.5815/ijcnis.2018.07.06.
- [8] N. Wu, P. Shang, J. Fan, Z. Yang, W. Ma, and Z. Liu, "Research on Coverless Text Steganography Based on Single Bit Rules," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jul. 2019. doi: 10.1088/1742-6596/1237/2/022077.
- [9] N. Wu, P. Shang, J. Fan, Z. Yang, W. Ma, and Z. Liu, "Coverless Text Steganography Based on Maximum Variable Bit Embedding Rules," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jul. 2019. doi: 10.1088/1742-6596/1237/2/022078.
- [10] N. Wu, W. Ma, Z. Liu, P. Shang, Z. Yang, and J. Fan, "Coverless text steganography based on half frequency crossover rule," in *Proceedings - 2019*

*4th International Conference on Mechanical, Control and Computer Engineering, ICMCCE 2019*, Institute of Electrical and Electronics Engineers Inc., Oct. 2019, pp. 726–729. doi: 10.1109/ICMCCE48743.2019.00168.

- [11] N. Alghamdi and L. Berriche, “Capacity investigation of Markov chain-based statistical text steganography: Arabic language case,” *ACM Int. Conf. Proceeding Ser.*, pp. 37–43, 2019, doi: 10.1145/3314527.3314532.
- [12] N. Wu *et al.*, “Coverless text hiding method based on improved evaluation index and one-bit embedding,” *C. - Comput. Model. Eng. Sci.*, vol. 124, no. 3, pp. 1035–1048, 2020, doi: 10.32604/cmescs.2020.010450.
- [13] W. Zhang, X. Wang, C. Zhang, and J. Zhang, “Coverless Text Steganography Method Based on Characteristics of Word Association,” in *International Conference on Communication Technology Proceedings, ICCT*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 1139–1144. doi: 10.1109/ICCT50939.2020.9295910.
- [14] A. Majumder, S. Kundu, and S. Changder, “A unique database synthesis technique for coverless data hiding,” *J. Vis. Commun. Image Represent.*, vol. 96, p. 103911, Oct. 2023, doi: 10.1016/j.jvcir.2023.103911.
- [15] X. Chen, H. Sun, Y. Tobe, Z. Zhou, and X. Sun, “Coverless information hiding method based on the Chinese mathematical expression,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2015, pp. 133–143. doi: 10.1007/978-3-319-27051-7\_12.
- [16] Z. Zhou, Y. Mu, N. Zhao, Q. M. Jonathan Wu, and C. N. Yang, “Coverless information hiding method based on multi-keywords,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2016, pp. 39–47. doi: 10.1007/978-3-319-48671-0\_4.
- [17] J. Zhang, J. Shen, L. Wang, and H. Lin, “Coverless text information hiding method based on the word rank map,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2016, pp. 145–155. doi: 10.1007/978-3-319-48671-0\_14.
- [18] Y. Liu, Institute of Electrical and Electronics Engineers, and IEEE Circuits and Systems Society, *Multi-keywords Carrier-free Text Steganography Based on Part of Speech Tagging*.

- [19] J. Zhang, H. Huang, L. Wang, H. Lin, and D. Gao, “Coverless text information hiding method using the frequent words hash,” *Int. J. Netw. Secur.*, vol. 19, no. 6, pp. 1016–1023, 2017, doi: 10.6633/IJNS.201711.19(6).18.
- [20] J. Zhang, Y. Xie, L. Wang, and H. Lin, “Coverless text information hiding method using the frequent words distance,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2017, pp. 121–132. doi: 10.1007/978-3-319-68505-2\_11.
- [21] C. Liu, G. Luo, and Z. Tian, “Coverless information hiding technology research based on news aggregation,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2017, pp. 153–163. doi: 10.1007/978-3-319-68505-2\_14.
- [22] C. Yuan, Z. Xia, and X. Sun, “Coverless image steganography based on SIFT and BOF,” in *Journal of Internet Technology*, Taiwan Academic Network Management Committee, 2017, pp. 435–442. doi: 10.6138/JIT.2017.18.2.20160624c.
- [23] Z. Xia and X. Li, “Coverless Information Hiding Method Based on LSB of the Character’s Unicode,” *J. Internet Technol.*, vol. 18, no. 6, pp. 1353–1360, 2017, doi: 10.6138/JIT.2017.18.6.20160815b.
- [24] Y. Wu and X. Sun, “Text coverless information hiding method based on hybrid tags,” *J. Internet Technol.*, vol. 19, no. 3, pp. 649–655, 2018, doi: 10.3966/160792642018051903003.
- [25] Y. Wu, X. Chen, and X. Sun, “Coverless steganography based on english texts using binary tags protocol,” in *Journal of Internet Technology*, Taiwan Academic Network Management Committee, 2018, pp. 599–606. doi: 10.3966/160792642018031902028.
- [26] Y. Long and Y. Liu, “Text Coverless Information Hiding Based on Word2vec,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2018, pp. 463–472. doi: 10.1007/978-3-030-00015-8\_40.
- [27] Z. Fu, H. Ji, and Y. Ding, “Label model based coverless information hiding method,” *J. Internet Technol.*, vol. 19, no. 5, pp. 1509–1514, 2018, doi: 10.3966/160792642018091905022.

- [28] X. Chen and S. Chen, "Text coverless information hiding based on compound and selection of words," *Soft Comput.*, vol. 23, no. 15, pp. 6323–6330, Aug. 2019, doi: 10.1007/s00500-018-3286-7.
- [29] H. Ji and Z. Fu, "Coverless information hiding method based on the keyword," 2019.
- [30] Y. Long, Y. Liu, Y. Zhang, X. Ba, and J. Qin, "Coverless Information Hiding Method Based on Web Text," *IEEE Access*, vol. 7, pp. 31926–31933, 2019, doi: 10.1109/ACCESS.2019.2901260.
- [31] K. Wang and Q. Gao, "A Coverless Plain Text Steganography Based on Character Features," *IEEE Access*, vol. 7, pp. 95665–95676, 2019, doi: 10.1109/ACCESS.2019.2929123.
- [32] X. Zhou *et al.*, "A novel coverless text information hiding method based on double-tags and twice-send," 2020.
- [33] Y. Liu, J. Wu, and G. Xin, "Multi-keywords carrier-free text steganography method based on Chinese Pinyin," 2020.
- [34] L. Xiang, J. Qin, X. Xiang, Y. Tan, and N. N. Xiong, "A robust text coverless information hiding based on multi-index method," *Intell. Autom. Soft Comput.*, vol. 29, no. 3, pp. 899–914, 2021, doi: 10.32604/iasc.2021.017720.
- [35] J. Qin, Z. Zhou, Y. Tan, X. Xiang, and Z. He, "A big data text coverless information hiding based on topic distribution and tf-idf," *Int. J. Digit. Crime Forensics*, vol. 13, no. 4, pp. 40–56, Jul. 2021, doi: 10.4018/IJDCF.20210701.oa4.
- [36] Y. Liu, J. Wu, and X. Chen, "An improved coverless text steganography algorithm based on pretreatment and POS," *KSII Trans. Internet Inf. Syst.*, vol. 15, no. 4, pp. 1553–1567, Apr. 2021, doi: 10.3837/tiis.2021.04.020.
- [37] K. Wang, X. Yu, and Z. Zou, "A Coverless Text Steganography by encoding the Chinese Characters' Component Structures," *Int. J. Digit. Crime Forensics*, vol. 13, no. 6, Nov. 2021, doi: 10.4018/IJDCF.20211101.oa4.
- [38] Y. Wen, J. Zhang, Y. Xia, H. Lin, and G. Sun, "Coverless Information Hiding Method Based on Combination Morse Code and Double Cycle Application of Starter," in *Communications in Computer and Information Science*, Springer Science and Business Media Deutschland GmbH, 2021, pp. 299–311. doi: 10.1007/978-3-030-78618-2\_24.



- [39] B. Guan, L. Gong, and Y. Shen, “A Novel Coverless Text Steganographic Algorithm Based on Polynomial Encryption,” *Secur. Commun. Networks*, vol. 2022, 2022, doi: 10.1155/2022/1153704.
- [40] N. Wu, Z. Liu, W. Ma, P. Shang, Z. Yang, and J. Fan, “Research on coverless text steganography based on multi-rule language models alternation,” in *Proceedings - 2019 4th International Conference on Mechanical, Control and Computer Engineering, ICMCCE 2019*, Institute of Electrical and Electronics Engineers Inc., Oct. 2019, pp. 803–806. doi: 10.1109/ICMCCE48743.2019.00184.
- [41] N. Wu *et al.*, “STBS-Stega: Coverless text steganography based on state transition-binary sequence,” *Int. J. Distrib. Sens. Networks*, vol. 16, no. 3, Mar. 2020, doi: 10.1177/1550147720914257.
- [42] Z. Zhou, Y. Mu, C. N. Yang, and N. Zhao, “Coverless multi-keywords information hiding method based on text,” *Int. J. Secur. its Appl.*, vol. 10, no. 9, pp. 309–320, 2016, doi: 10.14257/ijasia.2016.10.9.30.
- [43] J. Zhang, J. Shen, L. Wang, and H. Lin, “Coverless text information hiding method based on the word rank map,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2016, pp. 145–155. doi: 10.1007/978-3-319-48671-0\_14.
- [44] S. Shi, Y. Qi, and Y. Huang, “An Approach to Text Steganography Based on Search in Internet,” *Int. Comput. Symp.*, 2016, doi: 10.1109/ICS.2016.51.
- [45] Y. Liu, J. Wu, and X. Chen, “An improved coverless text steganography algorithm based on pretreatment and POS,” *KSII Trans. Internet Inf. Syst.*, vol. 15, no. 4, pp. 1553–1567, Apr. 2021, doi: 10.3837/tiis.2021.04.020.
- [46] H. H. Liu and C. M. Lee, “High-capacity reversible image steganography based on pixel value ordering,” *Eurasip J. Image Video Process.*, vol. 2019, no. 1, 2019, doi: 10.1186/s13640-019-0458-z.
- [47] N. F. Omran, N. R. Mahmoud, and A. A. Ali, “Securing Messages by Using Coverless Steganography : A Survey,” vol. 13, no. 02, pp. 335–345, 2022.
- [48] “A Boat Beneath a Sunny Sky by Lewis Carroll | Poetry Foundation.” Accessed: Sep. 07, 2023. [Online]. Available: <https://www.poetryfoundation.org/poems/43907/a-boat-beneath-a-sunny-sky>

- [49] C.-Y. Chang and S. Clark, "Human Linguistic Steganography Using Automatically Generated Paraphrases."
- [50] W. Liang, "A survey of text generation models," *Appl. Comput. Eng.*, vol. 45, no. 1, pp. 35–39, 2024, doi: 10.54254/2755-2721/45/20241023.
- [51] M. Chapman and G. Davida, "A Hiding the Hidden: Software System for Concealing Ciphertext Innocuous Text as."
- [52] Y. Luo and Y. Huang, "Text steganography with high embedding rate: Using recurrent neural networks to generate Chinese classic poetry," *IH MMSec 2017 - Proc. 2017 ACM Work. Inf. Hiding Multimed. Secur.*, pp. 99–104, 2017, doi: 10.1145/3082031.3083240.
- [53] Z. L. Yang, X. Q. Guo, Z. M. Chen, Y. F. Huang, and Y. J. Zhang, "RNN-Stega: Linguistic Steganography Based on Recurrent Neural Networks," *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 5, pp. 1280–1295, 2019, doi: 10.1109/TIFS.2018.2871746.
- [54] H. Ji and Z. Fu, "Coverless information hiding method based on the keyword," 2019., 2019.
- [55] G. F. Luger, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, vol. 5th. 2005. [Online]. Available: <http://www.amazon.com/dp/0321545893>
- [56] N. Privault, *Understanding Markov Chains*.
- [57] I. J. I. Technology, C. Science, T. Almutiri, and F. Nadeem, "Markov Models Applications in Natural Language Processing : A Survey," no. April, pp. 1–16, 2022, doi: 10.5815/ijitcs.2022.02.01.
- [58] "Automatically Generate Steganographic Text Based on Markov Model and Huffman Coding".
- [59] J. H. M. D. Jurafsky, "N-Gram Language Models N-Gram Language Models," 2020.
- [60] I. Al-Huri, "Arabic Language: Historic and Sociolinguistic Characteristics English Literature and Language Review Arabic Language: Historic and Sociolinguistic Characteristics," *English Lit. Lang. Rev.*, vol. 1, no. 4, pp. 28–36, 2015, doi: 10.13140/RG.2.2.16163.66089/1.

- [61] R. Thabit, N. I. Udzir, S. Md Yasin, A. Asmawi, N. A. Roslan, and R. Din, “A comparative analysis of arabic text steganography,” *Applied Sciences (Switzerland)*, vol. 11, no. 15. MDPI AG, Aug. 01, 2021. doi: 10.3390/app11156851.
- [62] A. Taha, A. S. Hammad, and M. M. Selim, “A high capacity algorithm for information hiding in Arabic text,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 32, no. 6, pp. 658–665, Jul. 2020, doi: 10.1016/j.jksuci.2018.07.007.
- [63] B. Al-Salemi, M. Ayob, G. Kendall, and S. A. M. Noah, “Multi-label Arabic text categorization: A benchmark and baseline comparison of multi-label learning algorithms,” *Inf. Process. Manag.*, vol. 56, no. 1, pp. 212–227, 2019, doi: 10.1016/j.ipm.2018.09.008.
- [64] N. Alifah Roslan, N. Izura Udzir, R. Mahmud, and A. Gutub, “Systematic literature review and analysis for Arabic text steganography method practically,” *Egypt. Informatics J.*, vol. 23, no. 4, pp. 177–191, 2022, doi: 10.1016/j.eij.2022.10.003.
- [65] A. F. A. AL-Nasrawi, Dhamyaa and W. Al-Baldawi, “From Arabic Alphabets to Two Dimension Shapes in Kufic Calligraphy Style Using Grid Board Catalog Dhamyaa A. AL-Nasrawi, Ahmed F. Almukhtar and Wafaa S. AL-Baldawi,” *Commun. Appl. Sci.*, vol. 3, no. 2, pp. 42–59, 2015.
- [66] A. A., F. Ridzuan, and S. Ali, “Text Steganography using Extensions Kashida based on the Moon and Sun Letters Concept,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 8, 2017, doi: 10.14569/ijacsa.2017.080838.
- [67] N. A. Roslan, R. Mahmud, and N. I. Udzir, “Sharp-edges method in Arabic text steganography,” *J. Theor. Appl. Inf. Technol.*, vol. 33, no. 1, pp. 32–41, 2011.
- [68] E. A. Khan, “Using arabic poetry system for steganography,” *Asian J. Comput. Sci. Inf. Technol.*, vol. 4, no. 6, pp. 55–61, 2014, doi: 10.15520/ajcsit.v.
- [69] “A study of Arabic letter frequency analysis.” [Online]. Available: <https://www.intellaren.com/articles/en/a-study-of-arabic-letter-frequency-analysis>
- [70] E. Agyepong, W. J. Buchanan, and K. Jones, “Detection of Algorithmically Generated Malicious Domain,” no. July, pp. 13–32, 2018, doi: 10.5121/csit.2018.80802.
- [71] “SANAD.” [Online]. Available:

<https://www.kaggle.com/datasets/haithemhermessi/sanad-dataset?select=Tech>

- [72] “APCD,” kaggle. [Online]. Available: <https://www.kaggle.com/datasets/mohamedkhaledelsafty/best-arabic-poem-comprehensive-dataset>
- [73] “Arabic Poetry Dataset,” kaggle. [Online]. Available: <https://www.kaggle.com/datasets/fahd09/arabic-poetry-dataset-478-2017>

## الخلاصة

في العصر الرقمي، يُعدّ حماية المعلومات السرية من الوصول غير المصرح به أمرًا بالغ الأهمية. المعلومات يمكن أن تُعبر عنها بوسائل اتصال متعددة مثل النصوص، والصوت، والفيديو، والصور، مع كون النصوص الأكثر شيوعًا. تهدف تقنية الإخفاء (Steganography) إلى إخفاء المعلومات بحيث لا يلاحظها الآخرون، من خلال إدراجها في وسط ناقل آخر. يختلف هذا الأسلوب عن تقنيات تبادل المعلومات السرية الأخرى مثل التشفير، حيث يمكن اكتشاف وجود المعلومات المُشفرة لكن يصعب فهمها. أما في الإخفاء، فلا يمكن لأحد أن يعرف أن البيانات موجودة أصلاً داخل المصدر.

من التحديات التي تواجه طريقة الإخفاء النصي بدون غطاء تقليدي هي انخفاض السعة العالية، وارتفاع مستوى التعقيد (perplexity)، وغياب التطبيقات باللغة العربية. بالمقابل، لا يتطلب الإخفاء النصي بدون غطاء تعديل الوسيط الناقل بل ينقل المعلومات المخفية مباشرة عبر ميزات داخلية في النص.

تهدف هذه الأطروحة إلى تحسين تقنيات الإخفاء النصي بدون غطاء من حيث سعة الإخفاء، نسبة النجاح، دقة الاستخراج، تحليل الأمان، توفر وفعالية الخوارزمية. وكذلك، توسيع تطبيقات هذه التقنيات لتشمل اللغة العربية من خلال الاستفادة من النموذج الإحصائي للغة العربية والميزات اللغوية التي يمكن استخدامها لإخفاء المعلومات.

تُقدّم في هذه الأطروحة طريقتان جديدتان للإخفاء النصي بدون غطاء؛ الأولى تعتمد على نموذج إحصائي للغة العربية باستخدام سلاسل ماركوف من الدرجة الأولى، والثانية تعتمد على ميزات مدمجة في اللغة العربية.

تم استخدام ثلاث مجموعات بيانات عربية في هذه الأطروحة: مجموعة بيانات أخبار عربية تحتوي على 45,500 مقال، مجموعة شاملة من الشعر العربي تحتوي على 1,831,770 بيت شعر، ومجموعة بيانات شعرية تحتوي على أكثر من 58,000 قصيدة.

تستخدم الطريقة الأولى سلاسل ماركوف من الدرجة الأولى لتوليد نصوص مخفية دون الحاجة إلى وسيلة نقل خارجية. تم اختيار مجموعة من النصوص العربية وإنشاء مخطط انتقال يعتمد على تكرار الكلمات. يُستخدم رمز معين لتمثيل الانتقالات في المخطط، مما يسمح بتوليد نص يخفي المعلومات. أظهرت هذه الطريقة تحسناً في سعة الإخفاء حيث وصلت إلى 5.5، وانخفاضاً في التعقيد ليصل إلى 18.51، مما يشير إلى فعالية الطريقة في إخفاء المعلومات.

أما الطريقة الثانية تركز على الكلمة الأولى في كل صف من مجموعة بيانات بناءً على ثمانية ميزات محددة—الهمزة، التشكيل، الحروف المنفصلة، الحروف ذات الحافتين الحادتين، الحركات، النقط، الحروف ذات الحلقة، والتكرار العالي—لتوليد قيمة بايت (1 أو 0) بناءً على وجود أو غياب هذه الميزات. يتم بعد ذلك تحويل هذه القيمة إلى عدد عشري (كود ASCII لإنشاء بروتوكول ترميز ديناميكي مع الحرف الأكثر تكرارًا). حققت هذه

الطريقة معدل دقة عالي جداً بنسبة 100%، مما يعكس دقتها في تضمين واسترجاع المعلومات المخفية دون تغيير في البنية اللغوية للنص. علاوة على ذلك، حققت الطريقة أيضاً معدل نجاح بنسبة 100%، مما يبرز موثوقيتها في إخفاء وكشف المعلومات المضمنة بنجاح. ومع ذلك، فإن القدرة على الإخفاء باستخدام هذه الطريقة بلغت 0.246، مما يعكس التوازن بين الحفاظ على سلامة النص اللغوية وكمية المعلومات التي يمكن إخفاؤها.



جامعة كربلاء  
كلية علوم الحاسوب وتكنولوجيا المعلومات  
قسم علوم الحاسوب

## طرائق إخفاء النصوص غير المغطاة اعتماداً على خصائص اللغة العربية

رسالة ماجستير  
مقدمة الى مجلس كلية علوم الحاسوب وتكنولوجيا المعلومات / جامعة كربلاء وهي جزء من متطلبات  
نيل درجة الماجستير في علوم الحاسوب

كتبت بواسطة

سبأ حامد رشيد حسن

بإشراف

أ.م.د. ضمياء عباس حبيب