A Smart English Text Zero-Watermarking Approach Based on Third-Level Order and Word Mechanism of Markov Model

Fahd N. Al-Wesabi[1, 2, *]

Abstract: Text information is principally dependent on the natural languages. Therefore, improving security and reliability of text information exchanged via internet network has become the most difficult challenge that researchers encounter. Content authentication and tampering detection of digital contents have become a major concern in the area of communication and information exchange via the Internet. In this paper, an intelligent text Zero-Watermarking approach SETZWMWMM (Smart English Text Zero-Watermarking Approach Based on Mid-Level Order and Word Mechanism of Markov Model) has been proposed for the content authentication and tampering detection of English text contents. The SETZWMWMM approach embeds and detects the watermark logically without altering the original English text document. Based on Hidden Markov Model (HMM), Third level order of word mechanism is used to analyze the interrelationship between contexts of given English texts. The extracted features are used as a watermark information and integrated with digital zero-watermarking techniques. To detect eventual tampering, SETZWMWMM has been implemented and validated with attacked English text. Experiments were performed on four datasets of varying lengths under multiple random locations of insertion, reorder and deletion attacks. The experimental results show that our method is more sensitive and efficient for all kinds of tampering attacks with high level accuracy of tampering detection than compared methods.

Keywords: Content authentication, hidden Markov model, zero-watermarking, NLP, tampering detection.

1 Introduction

Security issues of digital text in various languages and formats have assumed great importance in communication technologies, especially in terms of content authentication, integrity verification, and copyright protection. Numerous applications, such as e-commerce and e-Banking, impose many challenges during transfer of content via the internet. Most of the digital media transferred over the internet is in text form and is highly sensitive in terms of content, structure, syntax, and semantics. Malicious attackers may

[1] Department of Computer Science, King Khalid University, Muhayel Aseer, Saudi Arabia.

[2] Faculty of Computer and IT, Sana’a University, Sana’a, Yemen.

[*] Corresponding Author: Fahd N. Al-Wesabi. Email: fwesabi@gmail.com.

Received: 22 April 2020; Accepted: 08 June 2020.

CMC. doi:10.32604/cmc.2020.011151 www.techscience.com/journal/cmc

temper these digital contents during the transfer process, and thus the modified content can result in incorrect decisions [Nurul, Amirrudin, Lip et al. (2018)].

Several solutions of information security have been proposed for many proposes which includes encryption, data hiding, copyright protection, integrity verification and unauthorized access control [Fan, Huang and Hsu (2011)]. Digital watermarking is the most common techniques of information hiding for several proposes such as content authentication and copyright protection whenever, if any altering is made in the watermarked media, the original media still protected and prove its ownership. Digital watermarking uses a specific algorithm to hide information by embedding it in the digital images, audio, video or text [Singh and Chadha (2013); Kaur and Sharma (2016); Kaur and Sharma (2017)].

In last decade, the most common digital media transferred among various internet applications is in the form of text. Nevertheless, limited research focused in text solutions because text is natural language dependent and it is difficult to hide security information unlike images which security information can hide in pixels, audio in waves and video in frames [Al-Maweri, Ali, Adnan et al. (2015); Tayan, Kabir and Alginahi (2014)].

The most challenges in this area involve developing the appropriate methods to hide information in the sensitive text contents without any modification of it [Hakak, Amirrudin, Tayan et al. (2017)]. Digital Holy Qur'an in Arabic, eChecks, online exams and marking are some examples of such sensitive digital text content. Various features of Arabic alphabets such as diacritics, extended letters, and other Arabic symbols make it easy to change the main meaning of text content by making simple modifications such as changing diacritics arrangements [Dhiman and Singh (2016); Hakak, Kamsin, Tayan et al. (2017); Khizar, Abid, Mansoor et al. (2018)]. Hidden Markov model (HMM) is the most common technique of natural language processing (NLP), which is used for text analysis and extract the text features.

In this paper, the author presents an intelligent hybrid approach SETZWMWMM (Smart English Text Zero-Watermarking Approach Based on Third-Level Order and Word Mechanism of Markov Model) for content authentication and tampering detection of English text transmitted via Internet. The proposed approach is based on third level order of word Mechanism based on Markov Model. It consists of a model that operates in collaboration between zero watermarking technique and Markov model as NLP techniques. In this approach, the third order of word Mechanism has been used for text analysis in order to extract the interrelationships between the contents of the given English text and to generate a watermark key. The generated watermark will be embedded logically in the original English context without any modifications or effect on the size of original text. Embedded watermark will be used later after the transmission of text via the Internet to detect any tampering occurring on the received English text and to determine if it is authentic or not.

The major objective of SETZWMWMM approach is to achieve a high accuracy of content authentication and sensitive detection of attack tampering in English text, which has gained a great importance and needs more security and protection via the Internet.

The main contributions of SETZWMWMM are as follows.

- A hybrid text zero-watermarking and NLP approach has been developed for content authentication and tampering detection of digital English contents have been ignored by researchers in the literature for the main reasons that English text is natural language dependent and the complexity of hiding the watermark information which there is no locations to hide it within text as pixels in case of image, waves in audio and frames in video.

- Most modern techniques of information security and NLP have been integrated to improve tampering detection accuracy and embed watermark logically without need to make any modifications in the original text.
- The resilience against random attacks on text has been improved and the watermark distortion is detected to attacks of varying volume and nature.
- The approach has been developed without prior assumption in terms of size, structure, and contents of an English text documents which include character sets, numbers, and special symbols.
- Watermark capacity has been reduced and gets rid of external watermark key, it is generated as a result of text analysis process.
- Author compare the SETZWMWMM approach to other baseline approaches and performed implementation of self-developed program, and extensive experimental using various scenarios of English datasets and under main text attacks and volumes. By studying and analyzing the results, author observed that SETZWMWMM outperforms the baseline approaches in terms of tampering detection accuracy.

The rest of the paper has five core sections. Section 2 provides a literature review of the related work. Section 3 presents SETZWMWMM. Section 4 describes the implementation, simulation, and experimental. Section 5 describes the comparison and result discussion, and Section 6 has conclusion of the article.

2 Related work

In the literature, several research on text watermarking approaches and methods have been proposed for several proposes of information security. In this paper, the authors briefly review the most common classifications of text watermarking methods which are linguistic-based watermarking, structural-based watermarking, and zero-watermarking methods [Nurul, Amirrudin, Lip et al. (2018); Sameeka and Kalpesh (2018)].

2.1 Linguistic-based techniques

The linguistic-based text watermarking methods are naturally language-based techniques, which works by making some modifications to the semantics and syntactic nature of plain text in order to embed the watermark key [Nurul, Amirrudin, Lip et al. (2018); Chen, Ma and Lu (2016)]. In the linguistic and semantic-based approaches, information is hidden by making some manipulations on words and utilize them as watermark key using many methods and techniques such as synonym substitution, typos, noun-verbs, and text-meaning representational strings [Chen, Ma and Lu (2016)].

One of the syntactic-based method proposed in Reem et al. [Reem and Lamiaa (2018)]. The proposed method uses open word space to improve the capacity of Arabic text. It works by utilize each word space to hide the binary bit 0 or 1 through which physical modification of the original text is conducted. Other syntactic-based methods presented in [Mujtaba and

Asadullah (2015)] for copyright protection by considers the existence of Harakat (diacritics, i.e., Fat-ha, Kasra and Damma) in the Arabic language and reverses the Fatha for message hiding. Other English text watermarking method also make use of Unicode characters to hide the watermark information within English scripts. The ASCII code used for embedding is 00, however, and the Unicode used of multilingual for embedding are 01, 10, and 11 [Abdul, Wesam and Dhamyaa (2013)].

2.2 Structural-based techniques

The structural-based text watermarking approaches are based on content structure which alters the features or structure of the text to embed the watermark information [Nasr addin, Wan and Abdul (2016)]. This also include modifications in general formatting features of the original text to hide watermark key such as locations of letters or words, writing style, repeating some letters or altering the features of the text [Kaur (2015); Alotaibi and Elrefaei (2015)].

One of the early approaches following the structure-based approach change the locations of words in text [Bashardoost, Rahim and Saba (2017)]. Other structural-based approaches proposed in Liu et al. [Liu, Zhu and Xin (2015); Zhu, Xiang, Song et al. (2016)] for content authentication of Chinese text by merging properties of sentences and calculate its entropy. In these approaches, the contents of Chines text divided into sets of small sentences and obtain semantic code of each word, then calculate its entropy by semantic codes' frequency, and find the weight of each sentence by utilizing the sentence features such as entropy, length, relevance, and weight function. The extracted features utilized to generate the watermark by using the verbs, and nouns of the high-weight sentences.

2.3 Zero watermark -based techniques

Zero text watermark-based approaches are based on text features which is achieved by generating the watermark key from the text context. This means several text features should be obtained, extracted and utilized as a watermark information. Several techniques and solutions have been proposed based on text features includes number of words or sentences letters, first letter of each word, and appearance frequency of non-vowel ASCII letters and words [Milad (2018); Khizar, Abid, Mansoor et al. (2018); Zulfiqar, Shamim, Ghulam et al. (2018); Tayan, Yasser and Muhammed (2014); Hanaa and Maisa'a (2016); Mokhtar, Fadl and Fahd (2014); Fahd, Adnan and Kulkarni (2014)].

One of the available text zero-watermarking approaches presented in Milad et al. [Milad (2018)] which hide the watermarking information within the social media and validate it later in terms of accuracy and reliability. Other text zero-watermarking techniques proposed in Khizar et al. [Khizar, Abid, Mansoor et al. (2018)] to validate data integrity of text context over the internet of things. The watermark information is generated as a text features such as text size, data appearance frequency, and time of data capturing. The generated watermark will have to be embedded logically in the original contents before its transmission. In Zulfiqar et al. [Zulfiqar, Shamim, Ghulam et al. (2018)], a text zero-watermarking method has been developed for individual privacy protection based on certain measures such as Hurst exponent and zero crossing of the speech signals. Individual identity has to obtained to embed it as a watermark key. In the case of copyright protection of English text, a text zero-watermark methods have been proposed in Tayan et al. [Tayan, Yasser and Muhammed (2014); Hanaa and Maisa'a (2016)] which uses the appearance frequency of non-vowel ASCII letters and words.

According to combination solutions with zero-watermarking, such solutions presented in Mokhtar et al. [Mokhtar, Fadl and Fahd (2014); Fahd, Adnan and Kulkarni (2014)] which uses natural language processing and zero-watermarking for content authentication. The proposed methods trying to extract some text features to obtain the text probability properties and utilize it as a watermark key.

3 The proposed approach

This paper proposes a novel intelligent approach by integrating a zero-watermark and Hidden Markov as NLP techniques in which there is no need to embed extra information such as watermark key, or even to perform any modifications on the original text. Third level order of word mechanism of Markov model has been used as NLP technique to analyze the contents of English text and extract the interrelationships features of these text contents. I take the following set of assumptions in SETZWMWMM:

- Unlike the previous work, in which the watermarking is performed by effecting the text, content, and size, the SETZWMWMM approach embeds the watermarking logically without any effect on the text, content, and size.
- In SETZWMWMM, watermarking does not need any external information because the watermark key is produced as a result of text analysis and extracting the relationship between the content itself and then making it as a watermark.
- The SETZWMWMM approach is highly sensitive to any simple modification on the text or the meaning in the English text. The three contributions mentioned above are found somehow only in images but not in text. This is the vital point concerning to the contribution of this paper.
- In addition, the SETZWMWMM can effectively determine the place of tempering occurrence. This feature can be considered as advantage over Hash function method.

Two main processes should be performed in SETZWMWMM, which are text analysis and watermark generation process, and watermark extraction and detection process, illustrated in Figs. 1 and 2.

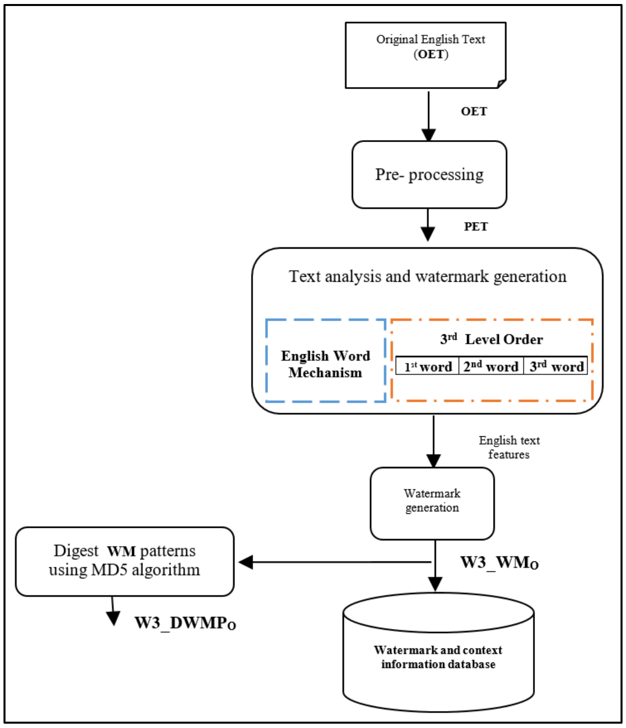

Original English Text (OET)
OET
Pre- processing
PET
Text analysis and watermark generation
English Word Mechanism
3rd Level Order
1st word
2nd word
3rd word
English text features
Watermark generation
Digest WM patterns using MD5 algorithm
W3_WMO
W3_DWMPO
Watermark and context information database

Figure 1: The proposed model of watermark generation and embedding processes

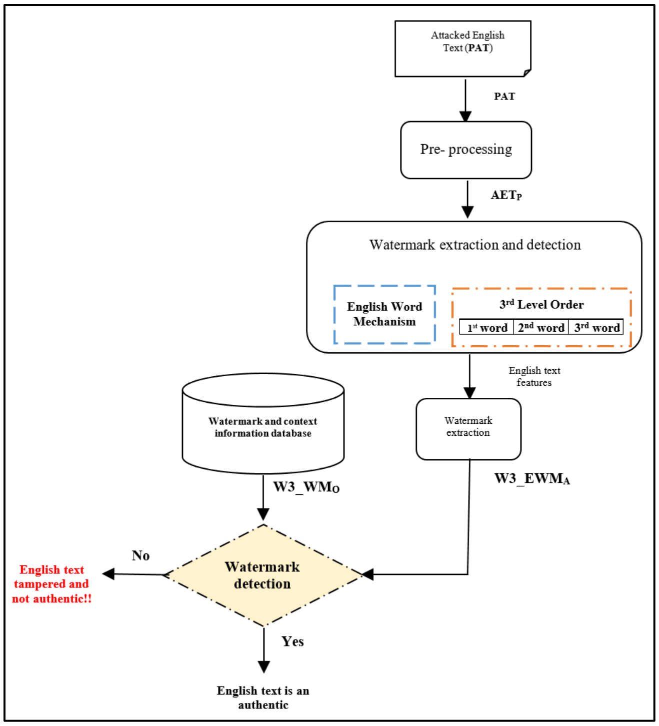

Attacked English Text (PAT)
PAT
Pre- processing
AETP
Watermark extraction and detection
English Word Mechanism
3rd Level Order
1st word
2nd word
3rd word
English text features
Watermark and context information database
Watermark extraction
W3_WMO
W3_EWMA
No
English text tampered and not authentic!!
Watermark detection
Yes
English text is an authentic

Figure 2: Watermark extraction and detection processes of SETZWMWMM

The following subsections explain in detail the watermark generation and extraction processes.

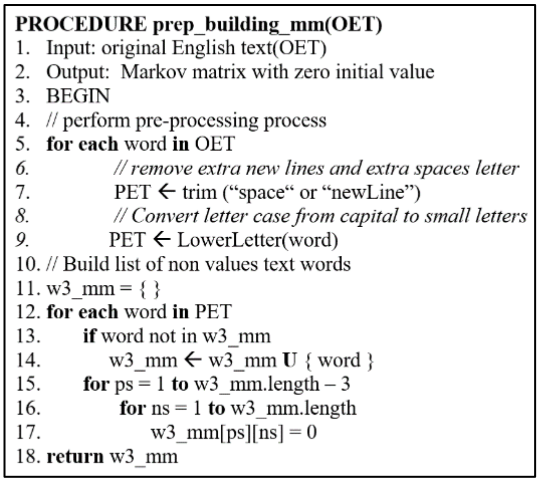
3.1 Text analysis and watermark generation process

Pre-processing process should be performed before watermark generation and embedding process in order to remove any extra spaces and new lines in the given English text. Pre-processed original English text document (PET) is required as input for the watermark generation and embedding algorithm. The output of this algorithm is original watermark pattern ($W3\_WM_O$). The generated watermark will be stored in watermark database beside the basic information of English text document such as author name, document size and identity, and last modified date. The three main sub-algorithms included in this process are pre-processing and building a Markov matrix algorithm, text analysis algorithm and watermark generation algorithm.

3.1.1 Pre-processing and building a Markov matrix algorithm

The original English text (OET) is required as input for Pre-processing process to remove extra spaces and new lines. Building a Markov matrix is the starting point of English text analysis and watermark generation process using Markov model. A Markov matrix that represents the possible states and transitions available in a given text is constructed without reputations. In this approach, each unique triple of words within a given English text represents a present state, and each unique word represents a transition in Markov matrix. During the building process of Markov matrix, the proposed algorithm initializes all transition values by zero to use these cells later to keep track of the number of times that the i^{th} triple of words is followed by the j^{th} word within the given English text document. Pre-processing and building Markov matrix algorithm executes as presented in Algorithm. 1.

Algorithm 1. Pre-processing and building Markov algorithm of SETZWMWMM



```
PROCEDURE prep_building_mm(OET)
Input: original English text(OET)
Output:  Markov matrix with zero initial value
BEGIN
// perform pre-processing process
for each word in OET
        // remove extra new lines and extra spaces letter
        PET ← trim (“space“ or “newLine”)
        // Convert letter case from capital to small letters
       PET ← LowerLetter(word)
// Build list of non values text words
w3_mm = { }
for each word in PET
    if word not in w3_mm
       w3_mm ← w3_mm U { word }
    for ps = 1 to w3_mm.length – 3
       for ns = 1 to w3_mm.length
          w3_mm[ps][ns] = 0
return w3_mm
```

where, OET: is an original English text, PET: is a pre-processed English text, w3_mm: states and transitions matrix with zeros values for all cells, ps: refers to current state, ns: refers to next state.

According to the above, a method is presented to construct two-dimensional matrix of Markov states and transitions named $\mathrm{w3\_mm}[i][j]$, which represents the backbone of Markov model for English text analysis.

The length of $w3\_\mathrm{mm}[i][j]$ matrix of SETZWMWMM is dynamic in which the number of states varies based on the context of a given English text, which is equal to the number of unique triple of words.

3.1.2 Text analysis algorithm

After the Markov matrix was constructed, the NLP for text analysis process should be performed to find the interrelationships between the contexts of the given English text, and generate watermark patterns. The following example of English text sample describes the mechanism of the transition process of present state to other next states.

"The quick brown fox jumps over the brown fox who is slow jumps over the brown fox who is dead."

When using the third level order of word mechanism of Hidden Markov model, every unique triple of words is a present state. Text analysis is processed as the text is being read to obtain the interrelationship between the present state and the next states. Fig. 3 below illustrates the available transitions of the above sample text and results of text analysis.

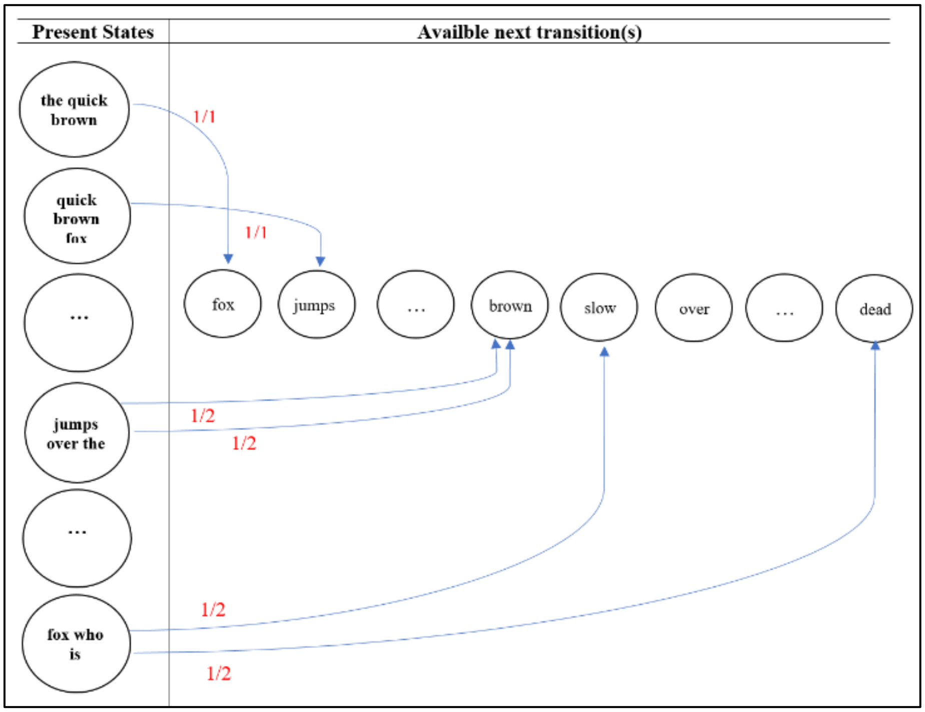


Figure 3: States representation and text analysis results of English text sample of SETZWMWMM

Author assumes that "*fox who is*" is present state, and the available next transitions are "*slow*" and "*dead*". Now, I present a method to construct a two-dimensional matrix of

Markov states and transitions named w3_mm[i][j], which represents the backbone of Markov model to English text analysis.

The length of w3_mm[i][j] matrix of SETZWMWMM is dynamic, which the number of states is varies based on the context of a given English text. In this algorithm, the number of appearances of possible next states transitions with non-zero-values for each current state of triple of words will be calculated and constructed as transition probabilities by Eq. (1).

$$w3\_mm[ps][ns] = \sum_{i,j=1}^{n-3} totalNumberofTransition(i,j) \tag{1}$$

where, n: is total number of states, i: is ith current state, and j: is jth next state transition.

Let PET be the pre-processed text, w3_mm[ps][ns] represents the Markov matrix to store values of the number of times that the ith triple of words (present state) is followed by the jth word (next state transitions) in the given English text. The watermark generation algorithm is presented formally and executed as illustrated in Algorithm. 2.

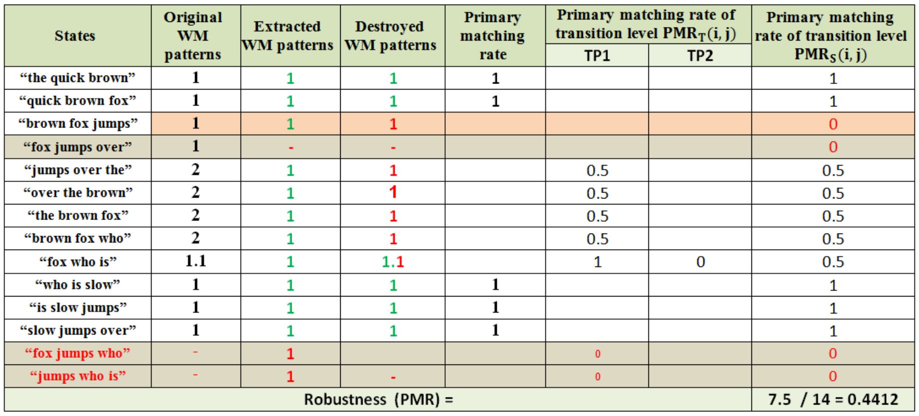
Algorithm 2: Text analysis algorithm of SETZWMWMM

```
PROCEDURE ETA(PET)

Input: PET, w3_imm[ps][ns]
Output: w3_mm[pw][nw]
BEGIN
prep_bulding_mm(PET)
pw = first_word(PET)
pd2 = PET – [pw] // begin with 2nd word
w3_mm[pw][nw] = w3_imm[ps][ns]
for each word in pd2
    w3_mm[pw][nw] = w3_mm[pw][nw] + 1
    pw = nw
return w3_mm[pw][nw]
```

where, w3_mm[ps][ns] refers to the initial matrix of Markov model with zero values, pw refers to pervious word, and nw refers to next word.

The results of text analysis algorithm based on third level order of word mechanism of Markov model proceeds as illustrated in Fig. 4.



States	Original WM patterns	Extracted WM patterns	Destroyed WM patterns	Primary matching rate	Primary matching rate of transition level $PMR_T(i, j)$		Primary matching rate of transition level $PMR_S(i, j)$
					TP1	TP2	
"the quick brown"	1	1	1	1			1
"quick brown fox"	1	1	1	1			1
"brown fox jumps"	1	1	1				0
"fox jumps over"	1	-	-				0
"jumps over the"	2	1	1		0.5		0.5
"over the brown"	2	1	1		0.5		0.5
"the brown fox"	2	1	1		0.5		0.5
"brown fox who"	2	1	1		0.5		0.5
"fox who is"	1.1	1	1.1		1	0	0.5
"who is slow"	1	1	1	1			1
"is slow jumps"	1	1	1	1			1
"slow jumps over"	1	1	1	1			1
"fox jumps who"	-	1			0		0
"jumps who is"	-	1	-		0		0
Robustness (PMR) =							7.5 / 14 = 0.4412

Figure 4: Text analysis process of given English text sample using SETZWMWMM

3.1.3 Watermark generation algorithm

After English text analysis has been performed and probability features were extracted, watermark key is generated by finding all non-zero values in Markov matrix. All of these non-zero values will be concatenated sequentially to generate the original watermark pattern W3_WM$_O$, as given in Eq. (2) and illustrated in Fig. 5.

$$W3\_WM_O \,\&= w3\_mm[ps]\,[ns], \quad \textit{for } i\,,\; j = \textit{non-zeros values resulted in } w3\_mm \tag{2}$$

1 – 1 – 1 – 1 – 2 – 2 – 2 – 2 – 1.1 – 1 – 1 – 1

Figure 5: The generated original watermark patterns W3_WM$_O$ in a decimal form using SETZWMWMM

The generated watermark W3_WM$_O$ is stored in WM database beside basic information of English text document. The generated watermark sequential patterns are then digested by using MD5 Hash algorithm to find a secure watermark form and reduce the capacity of watermark information, and they are denoted as W3_DWMP$_O$, notational as given in Eq. (3) and illustrated in Fig. 6.

$$W3\_DWMP_O = MD5(W3\_WM_O) \tag{3}$$

b7ec0e4dc73abb2c22575c61c54aabce

Figure 6: Digested original watermark patterns W3_DWMP$_O$

Algorithm of watermark generation based on third level order of word mechanism of Markov model is presented formally and executed as illustrated below in Algorithm. 3.

Algorithm 3: Watermark generation algorithm of SETZWMWMM

```
PROCEDURE wm_gen(PET)

Input: pre-processed text (PET)
Output: W3_DWMP$_O$, W3_WM$_O$
BEGIN
ETA (PET)
for ps = 1 to w3_mm[ps][ns].length - 3,
    for ns = 1 to w3_mm[ps][ns].length,
        if w5_mm[ps][ns] != 0
            W3_WM$_O$ &= w3_mm[ps] [ns]
W3_DWMP$_O$ = MD5(w3_WM$_O$)
return W3_DWMP$_O$, W3_WM$_O$
```

where, W3_WM$_O$: original watermark, W3_DWMP$_O$: digested original watermark, and MD5: hash algorithm.

3.2 Watermark extraction and detection process

Before the detection of pre-proceed attacked English text (AET$_P$), attacked watermark patterns (W3_EWM$_A$) should be generated, and matching rate of patterns and watermark distortion should be calculated by SETZWMWMM for detecting any tampering with the authentication of the given contents.

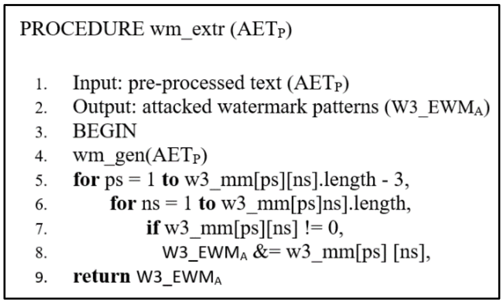
Two core algorithms are involved in this process, which are watermark extraction and watermark detection. However, W3_EWM$_A$ will be extracted from the received (AET$_P$) and matched with WM$_O$ by detection algorithm.

AET$_P$ should be provided as an input for the proposed watermark extraction algorithm. The same process of watermark generation algorithm should already have been performed to obtain the watermark pattern for (AET$_P$).

3.2.1 Watermark extraction algorithm

AET$_P$ is the main input required to run this algorithm. However, the output of this algorithm is EWM$_A$. The watermark extraction algorithm is presented formally and executed as illustrated in Algorithm. 4.

Algorithm 4: Watermark extraction algorithm of SETZWMWMM



```
PROCEDURE wm_extr (AET_P)

Input: pre-processed text (AET_P)
Output: attacked watermark patterns (W3_EWM_A)
BEGIN
wm_gen(AET_P)
for ps = 1 to w3_mm[ps][ns].length - 3,
    for ns = 1 to w3_mm[ps]ns].length,
        if w3_mm[ps][ns] != 0,
            W3_EWM_A &= w3_mm[ps] [ns],
return W3_EWM_A
```

where, AET$_P$: refers to pre-processed attacked English text, W3_EWM$_A$: refers to attacked watermark patterns.

3.2.2 Watermark detection algorithm

W3_EWM$_A$ and W3_WM$_O$ are the main inputs required to run watermark detection algorithm. However, the output of this algorithm is to notify whether the English text document is authentic or tampered. Detection process of extracted watermark is achieved in two main phases:

- *Primary matching* is achieved for W3_WM$_O$ and W3_EWM$_A$. If W3_EWM$_A$ and W3_WM$_O$ patterns appear identical, then, an alert will appear as "This English text is an authentic and no tampering occurred". Otherwise, notification will be "This English text is tampered", then continue to the next phase.
- *Secondary matching* is achieved by matching the transition of each state in a whole generated pattern. This means W3_EWM$_A$ of each state is compared with equivalent transition of W3_WM$_O$ as given by Eqs. (4) and (5) below.

$$\text{W3\_PMR}_\text{T}(i,j)\left|\frac{W3\_WM_O[i][j]-(W3\_WM_O[i][j]-W3\_EWM_A[i][j])}{W3\_WM_O[i][j]}\right| \quad \text{for all i,j states and transition} \qquad (4)$$

where,

- W3_PMR$_\text{T}$: represents pattern matching rate value in transition level, $(0 <$ W3_PMR$_T <=1)_T$
- i, j: refers to indexes of states and transitions respectively, i= 0. total number of non-zeros states, and j= 0. total number of non-zeros transitions in the given English text.
- W3_WM$_\text{O}$: refers to original watermark value in transition level.

- $\mathrm{W3\_EWM_A}$: refers to attacked watermark value in transition level.

$$W3\_PMR_S(i) = \left|\frac{\sum_{j=1}^{n-3}\left(\mathrm{W3}_{PMR_T(i,j)}\right)}{Total\ StatePatternCount(i)}\right| \quad \text{for all i} \tag{5}$$

where, n: is a total number of non zeros transitions of every state represented in matrix of Markov model, and W3_PMR$_S$ refers to value of pattern matching rate in state level, (0< W3_PMR$_S$ <=1).

After pattern-matching rate of every state that is produced, author finds the weight of every state from all the states in Markov matrix by using Eq. (6).

$$\mathrm{W3\_Sw} = \left|\frac{\mathrm{W3}\_PMR_S(i) * Transitions\ frequency(i)}{total\ number\ of\ transitions}\right| \tag{6}$$

The final W3_PMR of OET$_P$ and AET$_P$ are calculated by Eq. (7).

$$W3\_PMR = \left|\frac{\sum_{i=1}^{n-3}\mathrm{W3}\_PMRS(i)}{N}\right| \tag{7}$$

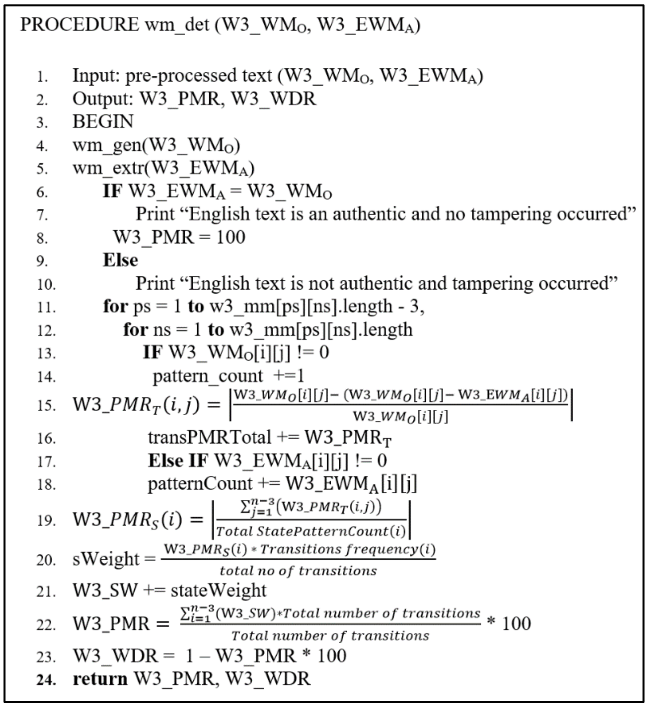
where, N: is a total number of non-zeros values in W3_mm.

Watermark distortion rate represents the amount of tampering occurred on contents of attacked English context which is denoted by W3_WDR and calculated by Eq. (8).

$$W3\_WDR = 100 - W3\_PMR \tag{8}$$

The steps involved in watermark detection algorithm are illustrated in Algorithm 5.

Algorithm 5: Watermark detection algorithm of SETZWMWMM



```
PROCEDURE wm_det (W3_WM$_O$, W3_EWM$_A$)

Input: pre-processed text (W3_WM$_O$, W3_EWM$_A$)
Output: W3_PMR, W3_WDR
BEGIN
wm_gen(W3_WM$_O$)
wm_extr(W3_EWM$_A$)
    IF W3_EWM$_A$ = W3_WM$_O$
        Print "English text is an authentic and no tampering occurred"
     W3_PMR = 100
    Else
        Print "English text is not authentic and tampering occurred"
    for ps = 1 to w3_mm[ps][ns].length - 3,
       for ns = 1 to w3_mm[ps][ns].length
          IF W3_WM$_O$[i][j] != 0
           pattern_count  +=1
W3_$PMR_T(i,j) = \left|\frac{\mathrm{W3}\_WM_O[i][j] - (\mathrm{W3}\_WM_O[i][j] - \mathrm{W3\_E}WM_A[i][j])}{\mathrm{W3}\_WM_O[i][j]}\right|$
          transPMRTotal += W3_PMR$_T$
          Else IF W3_EWM$_A$[i][j] != 0
          patternCount += W3_EWM$_A$[i][j]
W3_$PMR_S(i) = \left|\frac{\sum_{j=1}^{n-3}(\mathrm{W3}\_PMR_T(i,j))}{Total\ StatePatternCount(i)}\right|$
sWeight = $\frac{\mathrm{W3}\_PMR_S(i) * Transitions\ frequency(i)}{total\ no\ of\ transitions}$
W3_SW += stateWeight
W3_PMR = $\frac{\sum_{i=1}^{n-3}(\mathrm{W3}\_SW)*Total\ number\ of\ transitions}{Total\ number\ of\ transitions} * 100$
W3_WDR =  1 – W3_PMR * 100
return W3_PMR, W3_WDR
```

where, W3_SW: refers to weight value of states correctly matched, W3_WDR: refers to value of watermark distortion rate (0<W3_WDR$_S$<=100).

4 Implementation and simulation

To evaluate the tampering detection accuracy of SETZWMWMM, several scenarios of simulation and experiments are performed. This section depicts an implementation, simulation and experimental environment, experiment parameters, experimental scenarios of standard datasets and results discussion.

4.1 Implementation environment and setup

Self-developed program has been developed to test and evaluate the performance of SETZWMWMM. Implementation environment of SETZWMWMM are: CPU: Intel Core i7-4650U/2.3 GHz, RAM: 8.0 GB, Windows 10 – 64 bit, PHP Programming language with VS Code IDE.

4.2 Simulation and experimental parameters

A series of experiments and simulation scenarios of SETZWMWMM have been conducted using standard datasets with different sizes. experiments and simulation scenarios performed under predefined attacks with their volumes randomly on multiple locations of these datasets. The experimental and simulation parameters and their associated values that used to perform the experiments are given as follows in Tab. 1.

Table 1: Experimental and simulation parameters

Parameters	Value
English dataset size	[ESST, 179], [EMST, 421], [EHMST, 559] and [ELST, 2018]
Attack type	Insertion, deletion and rephrasing
Attack volumes	5%, 10%, 20% and 50%
Tampering detection accuracy	High with close to 100 Low with close to 0
W3_PMR	(High when W3_PMR>70, Mid when 40<W3_PMR<70, and Low when W3_PMR<40)
W3_WDR	(High when W3_WDR>70, Mid when 40<W3_WDR<70, and Low when W3_WDR<40)

4.3 Performance metrics

Tampering detection accuracy refers to the performance of SETZWMWMM, which is evaluated using the following metrics:

- Tampering detection accuracy (w3_PMR and w3_WDR) under very low volume (5%), low volume (10%), mid volume (20%) and high volume (50%) of all addressed attacks with all scenarios of Arabic dataset sizes.
- Desired tampering detection accuracy values close to 100%.

- Accuracy evaluation of tampering detection under all attacks with various volumes.
- Tampering detection accuracy comparison of SETZWMWMM approach with others and results evaluation of dataset size effect, attacks type effect, and attacks volumes effect against tampering detection accuracy.

4.4 Simulation, experiments and results discussion with SETZWMWMM

In this subsection, author evaluates the tampering detection accuracy of SETZWMWMM. The character set covers all English characters, spaces, numbers, and special symbols. Experiments are conducted on different volumes of datasets and various kinds of attacks with their rates as identified above in Tab. 1.

4.4.1 Tampering detection accuracy evaluation under all volumes of all attacks

To evaluate the overall tampering detection accuracy of SETZWMWMM, several experiments scenarios are performed under all types of attacks and their volumes as show in Tab. 2. The results are illustrated in Fig. 7.

Table 2: Tampering detection accuracy of SETZWMWMM under all volumes of all attacks

Attack Volume	Insertion	Deletion	Reorder
5%	94.13	91.44	81.35
10%	89.82	84.76	69.17
20%	81.82	73.5	51.82
s50%	65.01	42.28	26.55

From Tab. 2 above and Fig. 7 below, it seems that SETZWMWMM approach gives sensitive results of tampering detection under all attacks in which the structure, syntax and semantic of English text contents maybe effected. As a comparison of tampering based on attack types, results show that insertion attack has most sensitive tampering detection in all scenarios of attack volumes.

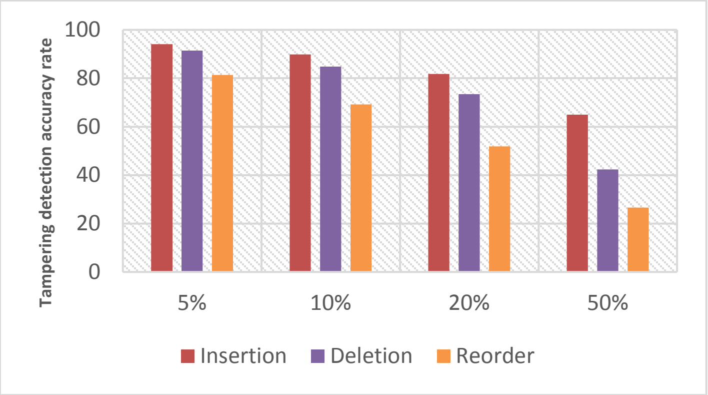


Figure 7: Tampering effect under all volumes of all attacks against all dataset sizes

5 Comparison and result discussion

The tampering detection accuracy results are critically analyzed. This subsection displays an effect study, and a comparison between SETZWMWMM and baseline approaches RACAAT and ZWAFWMMM. It also shows a discussion of their effect under the major factors i.e., dataset size, attack types and volumes.

5.1 Baseline approaches

Tampering detection accuracy of SETZWMWMM is compared with RACAAT (Robust Approach for Content Authentication of Arabic Text) and ZWAFWMMM (Zero-Watermarking Approach based on Fourth level order of Arabic Word Mechanism of Markov Model) [Fahd, Khalid and Nadhem (2020)]. Comparison is performed under all performance metrics to find which approach gives the best accuracy of tampering detection. Baseline approaches and their working parameters are stated in Tab. 3.

Table 3: Compared baseline approaches

Approach	Tampering nature and locations	Attack types	Attacks volumes	Objectives
ZWAFWMMM RACAAT	Random, multiple	Insertion, deletion and reorder	5%, 10%, 20% and 50%	Content authentication and tampering detection

5.2 Comparison and results study of attack type effect

Tab. 4 shows a comparison of the different attack types effect on tampering detection accuracy of SETZWMWMM, ZWAFWMMM and RACAAT approaches against all dataset sizes and all scenarios of attacks volumes.

Table 4: Attack type effect on tampering detection accuracy of SETZWMWMM, ZWAFWMMM and RACAAT approaches

Attack Type	Approach		
	ZWAFWMMM	RACAAT	SETZWMWMM
Insertion	80.02	74.28	81.67
Deletion	69.35	59.99	71.97
Reorder	44.88	37.23	49.81

Tab. 4 and Fig. 8 shows how the tampering detection accuracy of SETZWMWMM, ZWAFWMMM and RACAAT approaches are influenced by type of tampering attacks. In all cases of insertion, deletion and reorder attacks, SETZWMWMM outperforms ZWAFWMMM and RACAAT approaches with high rate of tampering detection accuracy. This means that, the SETZWMWMM approach is a strongly recommended and applicable for content authentication and tampering detection of English text under all attack types.

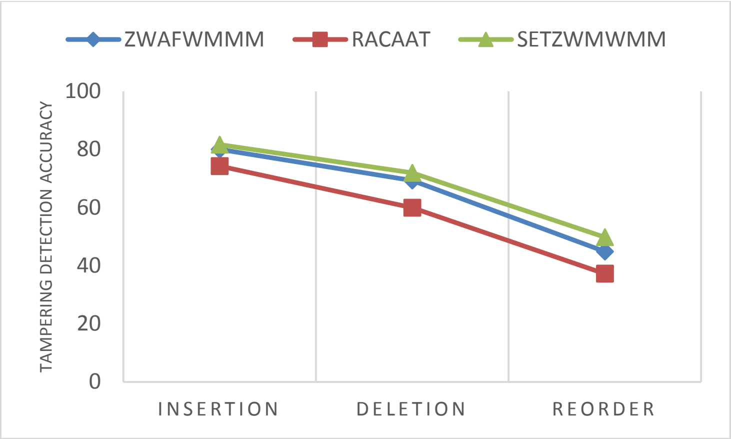


Figure 8: A compression of attack type effect on tampering detection accuracy of SETZWMWMM, ZWAFWMMM and RACAAT approaches

5.3 Comparison and results study of attack volume effect

Tab. 5 shows a comparison of the different attack volume effect on tampering detection accuracy against all dataset sizes and all scenarios of attacks volumes. The comparison is performed using SETZWMWMM, ZWAFWMMM and RACAAT approaches.

Table 5: Attack volume effect on tampering detection accuracy of SETZWMWMM, ZWAFWMMM and RACAAT approaches

Attack volume	Approach		
	ZWAFWMMM	RACAAT	SETZWMWMM
5%	82.09	83.60	85.97
10%	72.74	74.33	78.08
20%	57.71	59.39	63.01
50%	13.66	37.56	42.57

Tab. 5 and Fig. 9 shows how the tampering detection accuracy are influenced by low, mid and high attack volumes. In all cases of SETZWMWMM, ZWAFWMMM and RACAAT approaches, it can be seen that if the attack volume increases, the tampering detection accuracy also increases. However, if the attack volume decreases, the tampering detection accuracy also decreases.

In all cases of low, mid and high attack volumes, it seen SETZWMWMM outperforms both ZWAFWMMM and RACAAT approaches in terms of tampering detection accuracy in all scenarios of low, mid and high volumes of all attacks. This means that SETZWMWMM approach is a strongly recommended and applicable for content authentication and tampering detection of English text documents under all volumes of all attacks.

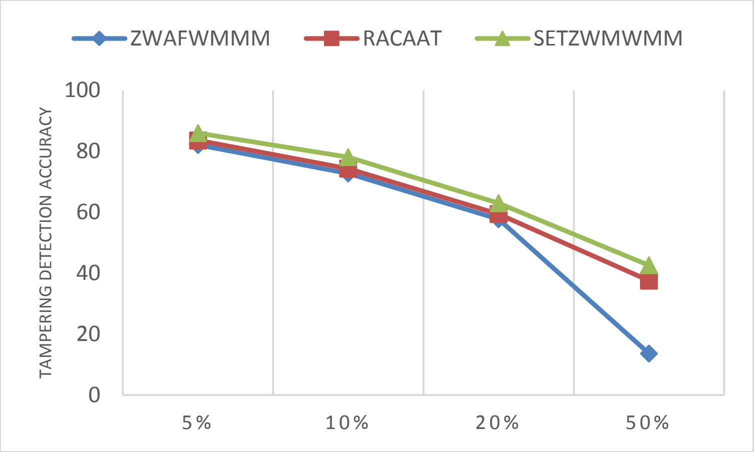


Figure 9: A compression of attack volume effect on tampering detection accuracy of SETZWMWMM, ZWAFWMMM and RACAAT approaches

5.4 Comparison and results study of dataset size effect

In this subsection, author presents an evaluation of the different dataset size effects on watermark robustness against all attack types under their different volumes. Tab. 6 shows a comparison of that effect using SETZWMWMM, ZWAFWMMM and RACAAT approaches.

Table 6: Dataset size effect on tampering detection accuracy of SETZWMWMM, ZWAFWMMM and RACAAT approaches

Dataset size	Approach		
	ZWAFWMMM	RACAAT	SETZWMWMM
[ESST]	67.27	69.53	72.65
[EMST]	63.80	68.13	71.88
[EHMST]	59.23	65.11	68.02
[ELST]	54.47	62.07	64.94

The comparative results as shown in Fig. 10 reflect the tampering detection accuracy of SETZWMWMM approach. The results show that in proposed SETZWMWMM approach, the highest effects of dataset size that lead to the best tampering detection accuracy with insertion and deletion attacks systematically are ordered as ESST, ELST, EMST, and EHMST, respectively. However, it is differing in case of reorder attacks. This means that, the tampering detection accuracy increased with decreasing document size and decreased with increasing document size. On the other hands, results show that SETZWMWMM approach outperforms both ZWAFWMMM and RACAAT approaches in terms of tampering detection accuracy under all scenarios of mid and large dataset sizes (ESST, EMST, EHMST, and ELST).

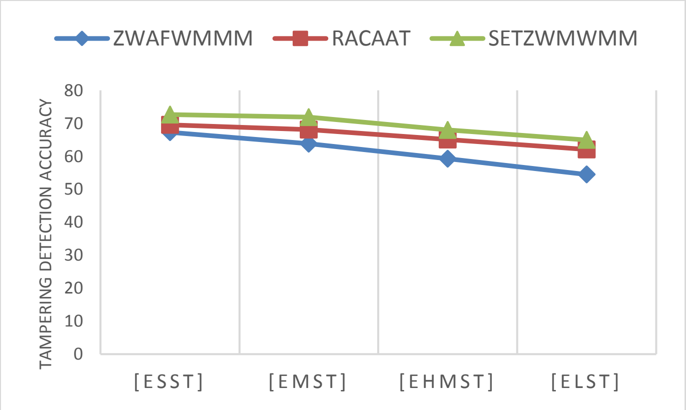


Figure 10: A compression of dataset size effect on tampering detection accuracy of SETZWMWMM, ZWAFWMMM and RACAAT approaches

6 Conclusion

Based on Third level order and word mechanism of Hidden Markov model, a novel hybrid approach of NLP and zero-watermarking has been developed which is abbreviated as SETZWMWMM for content authentication and tampering detection of English text transmitted via Internet. SETZWMWMM uses combination of zero watermarking technique and NLP techniques for text analysis in order to find interrelationships between the contents of a given English text and generated watermark key. The generated watermark is embedded logically in the original English context without modifications and effect on the size of original text. Embedded watermark is used later after the transmission of text via Internet to detect any tampering occurred on the received English text and ensures whether if it is authentic or not. SETZWMWMM approach is implemented in PHP using VS code IDE. The simulation and experiments are performed on various standard datasets under different volumes of insertion, deletion and reorder attacks. SETZWMWMM approach has been compared with ZWAFWMMM and RACAAT approaches. Comparison results show that SETZWMWMM outperforms ZWAFWMMM and RACAAT in term of general tampering detection accuracy under all attack types and volumes. For the future work, author will intend to improve the tampering detection accuracy using another techniques and mechanisms.

Funding Statement: The author extends his appreciation to the Deanship of Scientific Research at King Khalid University for funding this work under grant number (R. G. P. 2/55/40 /2019), Received by Fahd N. Al-Wesabi. www.kku.edu.sa

Conflicts of Interest: The author declares that he has no conflicts of interest to report regarding the present study.

References

Abdul, S.; Wesam, S.; Dhamyaa, A. (2013): Text steganography based on Unicode of characters in multilingual. *International Journal of Engineering Research and Applications*, vol. 3, no. 4, pp. 1153-1165.

Al-Maweri, N.; Ali, R.; Adnan, A.; Ramli, A.; Ahmad, S. (2015): State-of-the-art in techniques of text digital watermarking: challenges and limitations. *Journal of Computer Science*, vol. 12, no. 2, pp. 62-80.

Alotaibi, R.; Elrefaei, L. (2015): Arabic text watermarking? A review. *International Journal of Artificial Intelligence Applications*, vol. 6, no. 4, pp. 1-16.

Bashardoost, M.; Rahim, M.; Saba, T.; Rehman, A. (2017): Replacement attack: a new zero text watermarking attack. *3D Research*, vol. 8, no. 1.

Chen, J., F.; Ma, H.; Lu, Q. (2016): Text watermarking algorithm based on semantic role labeling. *Proceeding 3rd International Conference Digital Information Processing, Data Mining, Wireless Commun*ication, pp. 117-120.

Dhiman, S.; Singh, O. (2016): Analysis of visible and invisible image watermarking review. *International Journal of Computer Applications*, vol. 147, no. 3, pp. 36-38.

Fahd, N. Al-Wesabi; Adnan, Z.; Kulkarni, U. (2014): A zero text watermarking algorithm based on the probabilistic patterns for content authentication of text documents. *International Journal of Computer Engineering & Technology*, vol. 4, no. 1, pp. 284-300.

Fahd, N.; Khalid, M.; Nadhem, N. (2020): A zero watermarking approach for content authentication and tampering detection of Arabic text based on fourth level order and word mechanism of Markov model, *Elsevier Journal of Information Security and Applications*, vol. 52, pp. 1-15.

Fan, C. H.; Huang, H. Y.; Hsu, W. H. (2011): A robust watermarking technique resistant JPEG compression. *Journal of Information Science and Engineering*, vol. 27, pp. 163-180.

Hakak, S.; Amirrudin, K.; Tayan, O.; Yamani, I.; Amin, G. (2017): Approaches for preserving content integrity of sensitive online Arabic content: a survey challenges. *ELSEVIER Information Processing and Management*, vol. 56, no. 2, pp. 367-380.

Hakak, S.; Kamsin, A.; Tayan, O.; Idris, M.; Gilkar, G. (2017): Approaches for preserving content integrity of sensitive online Arabic content: a survey and research challenges. *Information Processing Management*, pp. 1-14.

Hanaa, M.; Maisa'a, A. (2016): Comparison of eight proposed security methods using linguistic steganography text. *International Journal of Computing & Information Sciences*, vol. 12, no. 2, pp. 243-251.

Kaur, B.; Sharma, S. (2017): Digital watermarking and security techniques: a review. *International Journal of Computer Science and Technology*, vol. 8, no. 2, pp. 44-47.

Kaur, M. (2015): An existential review on text watermarking techniques. *International Journal of Computer Applications*, vol. 120, no. 18, pp. 29-32.

Kaur, M.; Sharma, V. (2016): Encryption based LSB steganography technique for digital images and text data. *International Journal of Computer Science and Network Security*, vol. 16, no. 9, pp. 90-97.

Khizar, H.; Abid, K.; Mansoor, A.; Alavalapati, G. (2018): Towards a formally verified zero watermarking scheme for data integrity in IoT based-wireless sensor networks. *ELSEVIER Future Generation Computer Systems*, vol. 167, pp. 1-16.

Liu, Y.; Zhu, Y.; Xin, G. (2015): A zero-watermarking algorithm based on merging features of sentences for Chinese text. *Journal of the Chinese Institute of Engineers*, vol. 38, no. 3.

Milad, T. (2018): ANiTH: A novel intelligent text hiding technique. *IEEE Dataport*, vol. 10.

Mokhtar, M.; Fadl, M.; Fahdm N. Al-Wesabi. (2014): Combined Markov model and zero watermarking techniques to enhance content authentication of Arabic text documents. *International Journal of Computational Linguistics Research*, vol. 5, no. 1, pp. 26-42.

Mujtabam, S.; Asadullahm, S. (2015): A novel text steganography technique to Arabic language using reverse Fat5Th5Ta. *Pakistan Journal of Engineering, Technology and Sciences*, vol. 1, no 2, pp. 106-113.

Nasr, A.; Wan, A.; Abdul, R.; Khairulmizam, S.; Sharifah, M. (2016): Robust digital text watermarking algorithm based on Unicode extended characters. *Indian Journal of Science and Technology*, vol. 9, no. 48, pp. 1-14.

Nurul, S.; Amirrudin, K.; Lip, Y.; Hameedur, R. (2018): A review of text watermarking: theory, methods, and applications. *IEEE Access*, vol. 6, pp. 8011-8018.

Reem, A.; Lamiaa, A. (2018): Improved capacity Arabic text watermarking methods based on open word space. *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 2, pp. 236-248.

Sameeka, S.; Kalpesh, P. (2018): Securing web contents through invisible text watermarking for copyright protection. *International Journal of Engineering Development and Research*, vol. 6, no. 3, pp. 257-261.

Singh, P.; Chadha, R. (2013): A survey of digital watermarking techniques, applications and attacks. *International Journal of Engineering Innovation and Technologies*, vol. 2, no. 9, pp. 165-175.

Tayan, O.; Kabir, M.; Alginahi, Y. (2014): A hybrid digital-signature and zero-watermarking approach for authentication and protection of sensitive electronic documents. *Science World Journal*, vol. 2014, pp. 1-14.

Tayan, O.; Yasser, M.; Muhammed, N. (2014): An adaptive zero-watermarking approach for text documents protection. *International Journal of Image Processing Techniques*, vol. 1, no.1, pp. 33-36.

Zhu, P.; Xiang, G.; Song, W.; Li, A.; Zhang, Y. et al. (2016): A text zero watermarking algorithm based on Chinese phonetic alphabets. *Wuhan University Journal of Natural Sciences*, vol. 21, no. 4, pp. 277-282.

Zulfiqar, A.; Shamim, M.; Ghulam, M.; Muhammad, A. (2018): New zero-watermarking algorithm using Hurst exponent for protection of privacy in telemedicine. *IEEE Access*, vol. 6, pp. 7930-7940.