Computers, Materials & Continua
DOI:10.32604/cmc.2021.015785
Article

Tech Science Press

Text Analysis-Based Watermarking Approach for Tampering Detection of English Text

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

[1]Department of Computer Science, King Khalid University, Muhayel Aseer, Kingdom of Saudi Arabia
[2]Faculty of Computer and IT, Sana'a University, Sana'a, Yemen
[*]Corresponding Author: Fahd N. Al-Wesabi. Email: falwesabi@kku.edu.sa

Received: 06 December 2020; Accepted: 11 January 2021

Abstract: Due to the rapid increase in the exchange of text information via internet networks, the security and the reliability of digital content have become a major research issue. The main challenges faced by researchers are authentication, integrity verification, and tampering detection of the digital contents. In this paper, text zero-watermarking and text feature-based approach is proposed to improve the tampering detection accuracy of English text contents. The proposed approach embeds and detects the watermark logically without altering the original English text document. Based on hidden Markov model (HMM), the fourth level order of the word mechanism is used to analyze the contents of the given English text to find the interrelationship between the contexts. The extracted features are used as watermark information and integrated with digital zero-watermarking techniques. To detect eventual tampering, the proposed approach has been implemented and validated with attacked English text. Experiments were performed using four standard datasets of varying lengths under multiple random locations of insertion, reorder, and deletion attacks. The experimental and simulation results prove the tampering detection accuracy of our method against all kinds of tampering attacks. Comparison results show that our proposed approach outperforms all the other baseline approaches in terms of tampering detection accuracy.

Keywords: Text analysis; English language processing; hidden Markov model; zero-watermarking; content authentication; tampering detection

1 Introduction

For the research community, the reliability and security of exchanged text through the internet is the most promising and challenging field. In communication technologies, authentication of content and automated text verification of honesty in different languages and formats are of great significance. Numerous applications for instance; e-Banking and e-commerce render information transferred via the internet the most difficult. In terms of content, structure, grammar, and semantics, much of the digital media transferred over the internet is in text form and is very susceptible to online transmission. During the transfer process, malicious attackers can temper such digital content [1].

For information security, many algorithms and techniques are available such as the authentication of content, verification of integrity, detection of tampering, access control, and copyright protection. To overcome these issues, steganography and automated methods of watermarking are commonly used. A technique of digital watermarking (DWM) which can be inserted into digital material through various details such as text, binary pictures, audio, and video [2,3]. A fine-grained text watermarking procedure is proposed based on replacing the white spaces and Latin symbols with homoglyph characters [4].

Several conventional methods and solutions for text watermarking were proposed [4] and categorized into different classifications such as linguistic, structure, and image-based and format-based binary images [5]. To insert the watermark information into the document, most of these solutions require certain upgrades or improvements to the original text in digital format material. Zero-watermarking without any alteration to the original digital material to embed the watermark information is a new technique with smart algorithms that can be used [3]. Several techniques have been proposed to improve the capacity of coverless information hidden in digital media based on various techniques such as anime characters [6], to represent and convey confidential information using unmodified natural stego-carriers [7], and to map the relationships constructed between the inherent features of the images and the secret information [8].

Restricted research has centered on the appropriate solutions to verify the credibility of critical digital media online [9–11]. The verification of digital text and the identification of fraud in research earned great attention. Besides, text watermarking study has several researches concentrated on copyright protection in the last decade, but less interest and attention has been paid to integrity verification, identification of tampering, and authentication of content due to the existence of text content based on the natural language [12].

Proposing the most appropriate approaches and strategies for dissimilar formats and materials, especially in Arabic and English languages, is the most common challenge in this area [13,14]. Therefore, authentication of content, verification of honesty, and detection of sensitive text tampering are major issues in different systems that need critical solutions.

Some instances of such sensitive digital text content are Arabic interactive Holy Qur'an, online, eChecks, tests, and marks. Different Arabic alphabet characteristics such as diacritics lengthened letters and extra symbols of Arabic make it simple to modify the key meaning of the text material by making basic changes such as modifying diacritic arrangements [11,15]. The most popular soft computation and natural language processing (NLP) technique that supported the analysis of the text is HMM.

This paper presents a hybrid text analysis and zero-watermarking approach (HTAZWA) for content authentication and tampering detection of English text. The proposed technique combines the Markov model and zero-watermarking. The fourth-order of the word mechanism of the Markov model has been used for text analysis to extract the interrelationships between contents of the given English text which consequently generates the watermark key. The generated watermark is logically embedded in the original English context without modifying the original text. After text transmission, the embedded watermark is used to detect any tamper that occurred on the received English text and ensures its authenticity.

The primary objective of the HTAZWA strategy is to meet the high accuracy of content authentication and identification of sensitive tampering attacks of English text which is transmitted through the Internet. Therefore, the key contributions of the proposed HTAZWA can be presented as follows.

- For the identification of English text, a novel hybrid text watermarking with NLP technique has been suggested for tampering.
- Most modern information protection techniques combined NLP and tools of soft computing to enhance the robustness, sophistication, security issues of the watermark and to decrease the capability of the watermark and logically incorporate the watermark without making any modification to the original document.
- Any random tamper can be detected with varying amounts and characters.
- In terms of the scale, structure, and content documents of English-text, including character numbers, sets, and special symbols, the method was developed without prior assumptions.
- Watermark capability has been decreased and the existing watermark key is eliminated as a result of processing the text analysis.

The rest of the paper has five more sections. Section 2 provides a literature review of the related work. Section 3 presents HTAZWA. Section 4 describes the implementation, simulation, and experimental details. Section 5 describes the comparison and results discussion, and Section 6 offers conclusions.

2 Literature Review

Natural language is the foundation of linguistic text watermarking approaches. The mechanism of those methods embedding the watermark is based on changes applied to the semantic and syntactic essence of plain text [1].

To enhance the capability and imperceptibility of Arabic text, a method of text watermarking suggested room dependent on the accessible words [16]. In this method, any word-space is used to mask the Boolean bit 0 or 1 that physically modifies the original text. A text steganography technique was proposed to hide information in the Arabic language [17]. The step of this approach considers Harakat's existence in Arabic diacritics such as Kasra, and Damma as well as reverses Fatha to cover the message.

Kashida-marks invisible method of watermarking [18], based on the features of frequent recurrence of document security and authentication characters, was proposed. The method is based on a predetermined watermark key with a Kashida placed for a bit 1 and a bit omitted. The method of steganography of the text was proposed to use Kashida extensions depending on the characters 'moon' and 'sun' to write digital contents of the Arabic language [19]. Also, Kashida characters are seen alongside with characters from Arabic to decide which hidden secret bits are kept by specific characters. In this form, four instances are included in the kashida characters: Moon characters representing '00'; sun characters representing '01'; sun characters representing '10'; and moon characters representing '11'.

A text steganographic approach [20] based on multilingual Unicode characters has been suggested to cover details in English scripts for the use of the English Unicode alphabet in other languages. Thirteen letters of the English alphabet have been chosen for this approach. It is important to embed dual bits in a timeframe using ASCII code for embedding 00. However, multilingual ones were used by Unicode to embed between 01, and 10, as well as 11. The algorithm of text watermarking is used to secure textual contents from malicious attacks according to Unicode extended characters [21]. The algorithm requires three main steps, the development, incorporation, and extraction of watermarks. The addition of watermarks is focused on the development of predefined coding tables. While scrambling strategies are often used in generation and removal, the watermarking key is safe.

The substitution attack method focused on preserving the position of words in the text document has been proposed [22]. This method depends on manipulating word transitions in the text document. Authentication of Chinese text documents based on the combination of the properties of sentences and text-based watermarking approaches have been suggested [23,24]. The proposed method is presented as follows: A text of the Chinese language is split into a group of sentences, and for each word, the code of a semantic has been obtained. The distribution of semantic codes influences sentence entropy.

A zero-watermarking method has been proposed to preserve the privacy of a person who relies on the Hurst exponent and the nullity of the frames [25]. For watermark embedding, the two steps are determined to evaluate the unvoiced frames. The process of the proposed approach is based on integrating an individual's identity without notifying any distortion in the signals of medical expression.

A zero-watermarking method was proposed to resolve the security issues of English language text-documents, such as verification of content and copyright protection [26]. A zero-watermarking approach has been suggested based on Markov-model authentication of the content of English text [27,28]. In this approach, to extract the safe watermark information, the probability characteristics of the English text are involved and stored to confirm the validity of the attacked text-document. The approach provides security against popular text attacks with a watermark distortion rate if, for all known attacks, it is greater than one. For the defense of English text copyright, based on the present rate of ASCII non-vowel letters and terms, the conventional watermark approach [29] has been suggested.

3 The Proposed Approach

This paper proposes a hybrid text analysis and the zero-watermarking approach by integrating zero-watermark technique and the Markov model as NLP technique in which there is no need to embed extra information such as a watermark key, or even to perform any modification on the original text.

The fourth-order of word mechanism of the Markov model has been used as NLP techniques to analyze the contents of the given English text and extract the interrelationships features of these text contents.

The following subsections explain in detail two main processes that should be performed in HTAZWA. The first process is called the watermark generation and embedding process, whereas, the second one is called the watermark extraction and detection process.

3.1 Watermark Generation and Embedding Process

The core sub-processes consist of pre-processing, watermark embedding, and watermark generation algorithms as well as text analysis as illustrated in Fig. 1.

3.1.1 Algorithm of Pre-Processing

The pre-processing of the original English text is one of the key steps in both the watermark generation and extraction processes to convert letter case from capital to small letter and remove extra spaces and newlines, and it will directly influence the tampering detection accuracy. The original English text (OET) is required as input for the Pre-processing process.

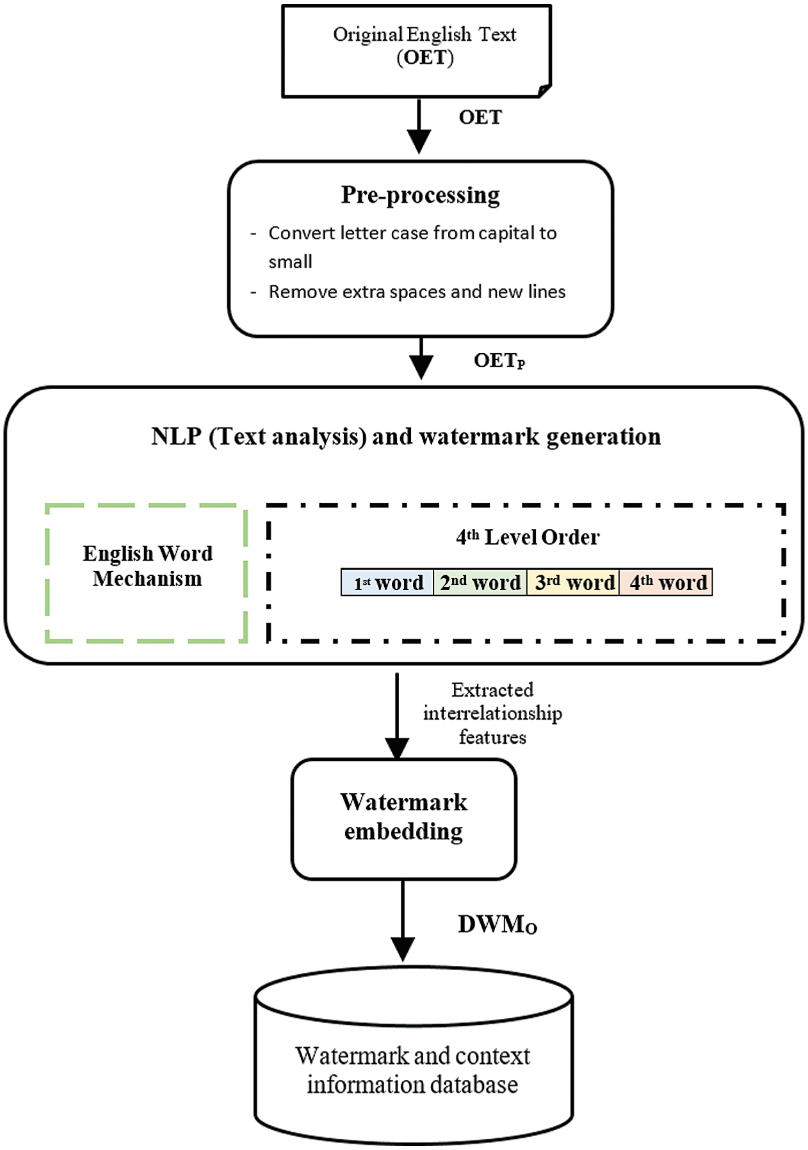

Original English Text
(OET)
OET
Pre-processing
- Convert letter case from capital to small
- Remove extra spaces and new lines
OETP
NLP (Text analysis) and watermark generation
English Word Mechanism
4th Level Order
1st word
2nd word
3rd word
4th word
Extracted interrelationship features
Watermark embedding
DWMO
Watermark and context information database

Figure 1: HTAZWA zero-watermark processes

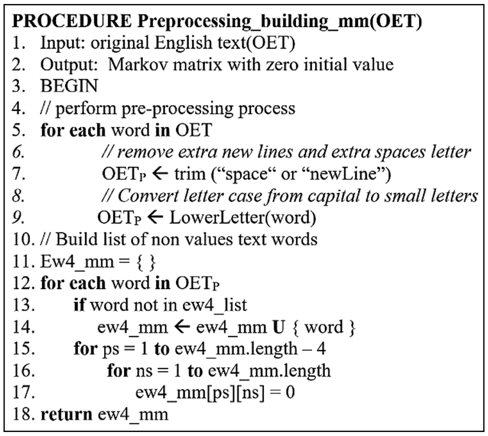
3.1.2 Algorithm of Watermark Generation

This algorithm involves the construction of the Markov matrix, the generation of watermarks, and the interpretation of the text.

- *Developing the Markov matrix* is a core step in developing the HTAZWA approach. A Markov chain matrix must be constructed in this process to set up the Markov model environment and represent all available states and transitions. In this approach, each unique sequence of four words within a provided English-text represents a current state, and every single word corresponds to a conversion in the matrix of the Markov chain. When constructing the Markov chain matrix, zero values will be initialized for all states and transition positions. Those positions will be used later holding a record of the number of occurrences. Then, the ith unique sequence of four words is backed up through the jth single word and provided by English-text.

Pre-processing and building the Markov matrix algorithm is executed as presented in Algorithm 1.

Algorithm 1. Algorithm of building Markov matrix using HTAZWA



```
PROCEDURE Preprocessing_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
        $OET_P$ ← trim (“space“ or “newLine”)
        // Convert letter case from capital to small letters
        $OET_P$ ← LowerLetter(word)
// Build list of non values text words
Ew4_mm = { }
for each word in $OET_P$
    if word not in ew4_list
        ew4_mm ← ew4_mm U { word }
    for ps = 1 to ew4_mm.length – 4
        for ns = 1 to ew4_mm.length
            ew4_mm[ps][ns] = 0
return ew4_mm
```

where,

OET: is the original English text, OET_P: is a pre-processed English text, ew4_mm: represents states and transitions matrix with zeros values for all cells, ps: refers to the current state, ns: refers to next state.

The length of ew4_mm$[i][j]$ matrix of HTAZWA is dynamic, in which the size of the rows is equal to the total number of unique sequences of four words within the given English text. However, the column size is equal to the total number of unique single words within the given English text.

- *Watermark generation-based text analysis process*: The proposed algorithm is performed as the second step of this process to perform English text analysis and extract the features of the given text and produce watermark information. Also, in this algorithm, there are several appearances of potential conversions for every present state of unique sequences of four words that will be computed as transition probabilities by Eq. (1) below.

$$ew4\_mm[ps][ns] = \sum_{i,j=1}^{n-4} trans[i][j] \tag{1}$$

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

This example of the English version demonstrates how this methodology was used to introduce the phase of transformation from the current state to the next state.

“The quick brown fox jumps over the brown fox who is slow jumps over the brown fox Who is dead.”

When using the fourth level order of word mechanism of the hidden Markov model, every unique sequence of four words is a present state. However, each unique single word is a present transition. Text analysis is processed as the text is read to obtain the interrelationship between the present state and the next states. Fig. 2. illustrates the available transitions and analysis results of the above sample of English text.

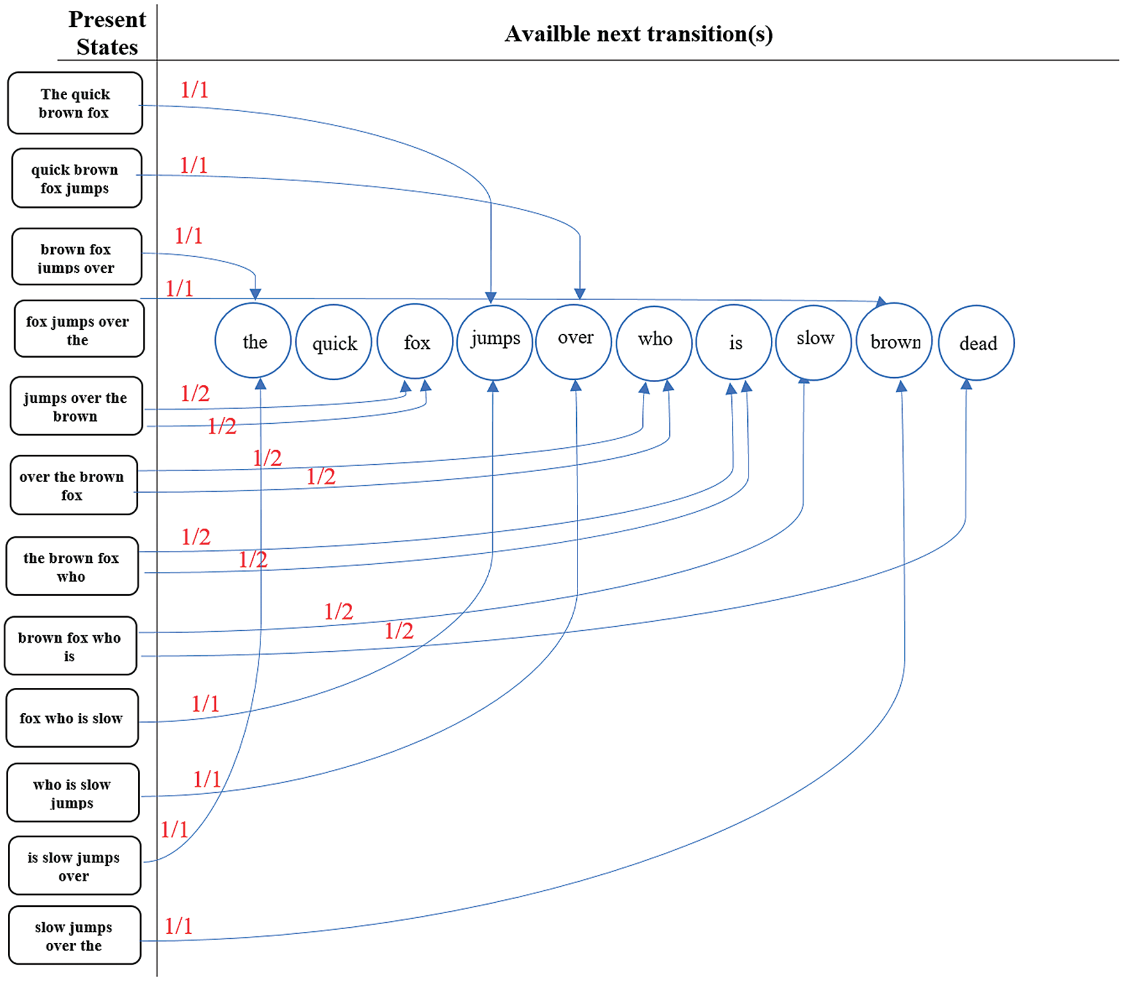
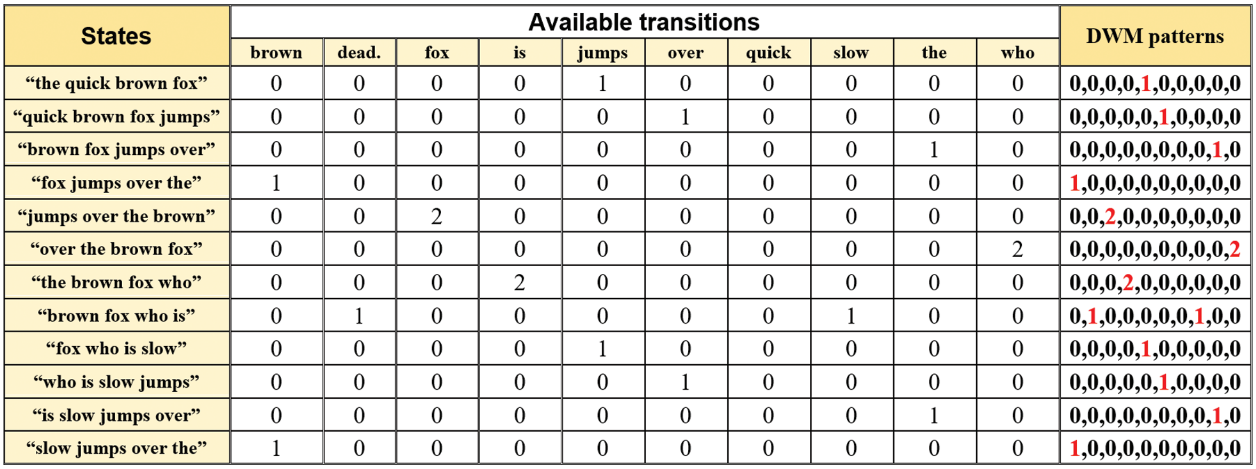


Figure 2: States and their transitions representation of sample given English text using HTAZWA

As shown in Fig. 2, and as a result of states and transitions representation of a given English text sample using HTAZWA, the author finds 12 unique states and 10 unique transitions. The author assumes "brown fox who is" is a present state, and the available next transition(s) is "slow" and "dead." The author also observed that "who" transition appears twice in the case of "over the brown fox" state in the given English text sample. The algorithm of text analysis and watermark generation based on the fourth-level order of the word mechanism of the Markov model proceeds as illustrated in Fig. 3.

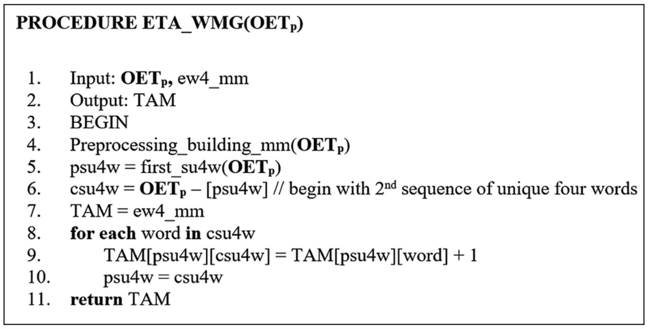


States	Available transitions										DWM patterns
	brown	dead.	fox	is	jumps	over	quick	slow	the	who	
"the quick brown fox"	0	0	0	0	1	0	0	0	0	0	0,0,0,0,1,0,0,0,0,0
"quick brown fox jumps"	0	0	0	0	0	1	0	0	0	0	0,0,0,0,0,1,0,0,0,0
"brown fox jumps over"	0	0	0	0	0	0	0	0	1	0	0,0,0,0,0,0,0,0,1,0
"fox jumps over the"	1	0	0	0	0	0	0	0	0	0	1,0,0,0,0,0,0,0,0,0
"jumps over the brown"	0	0	2	0	0	0	0	0	0	0	0,0,2,0,0,0,0,0,0,0
"over the brown fox"	0	0	0	0	0	0	0	0	0	2	0,0,0,0,0,0,0,0,0,2
"the brown fox who"	0	0	0	2	0	0	0	0	0	0	0,0,0,2,0,0,0,0,0,0
"brown fox who is"	0	1	0	0	0	0	0	1	0	0	0,1,0,0,0,0,0,1,0,0
"fox who is slow"	0	0	0	0	1	0	0	0	0	0	0,0,0,0,1,0,0,0,0,0
"who is slow jumps"	0	0	0	0	0	1	0	0	0	0	0,0,0,0,0,1,0,0,0,0
"is slow jumps over"	0	0	0	0	0	0	0	0	1	0	0,0,0,0,0,0,0,0,1,0
"slow jumps over the"	1	0	0	0	0	0	0	0	0	0	1,0,0,0,0,0,0,0,0,0

Figure 3: Text analysis and watermark generation of given English text sample using HTAZWA

Text analysis and watermark generation algorithm is presented formally as illustrated in Algorithm 2.

Algorithm 2. Watermark generation algorithm of HTAZWA



```
PROCEDURE ETA_WMG(OET_p)

Input: OET_p, ew4_mm
Output: TAM
BEGIN
Preprocessing_building_mm(OET_p)
psu4w = first_su4w(OET_p)
csu4w = OET_p – [psu4w] // begin with 2nd sequence of unique four words
TAM = ew4_mm
for each word in csu4w
    TAM[psu4w][csu4w] = TAM[psu4w][word] + 1
    psu4w = csu4w
return TAM
```

where, psu4w: is a previous sequence of unique four words, csu4w: is a current sequence of unique four words, and TAM: is text analysis matrix.

3.1.3 Algorithm of Watermark Embedding

Watermark embedding has taken place logically in this method with no need to change the original text. The feature extraction of the given English-text is utilized as a watermark key and embedded logically by identifying all non-zero values in the Markov chain matrix. All these non-zero values are sequentially concatenated to form the original pattern of watermark DWM_O, as defined in Eq. (2) and shown in Fig. 4.

$$ew4\_DWM_O \&= ew4\_mm[ps][ns], \quad for\ i,\ j = \text{non-zeros values resulted in ew4\_mm} \tag{2}$$

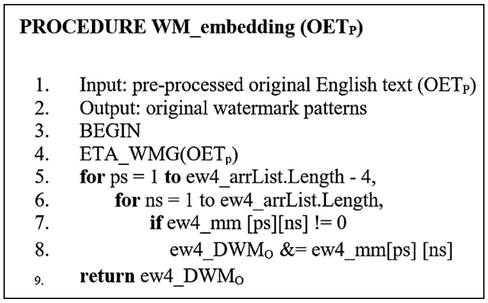
1 – 1– 1 – 1 – 2 – 2 – 2 – 1.1 – 1 – 1 – 1 - 1

Figure 4: The generated original pattern of watermark key $ew4\_DWM_O$ using HTAZWA

The generated ew4_DWM_O is stored in WM database beside information of the given English text.

The algorithm of watermark embedding based on the fourth-level order of the word mechanism of the Markov model is presented formally and executed as illustrated below in Algorithm 3.

Algorithm 3. Watermark embedding algorithm of HTAZWA



```
PROCEDURE WM_embedding ($OET_P$)

Input: pre-processed original English text ($OET_P$)
Output: original watermark patterns
BEGIN
ETA_WMG($OET_p$)
for ps = 1 to ew4_arrList.Length - 4,
    for ns = 1 to ew4_arrList.Length,
        if ew4_mm [ps][ns] != 0
            ew4_$DWM_O$ &= ew4_mm[ps] [ns]
return ew4_$DWM_O$
```

3.2 Algorithms of Watermark Extraction and Detection

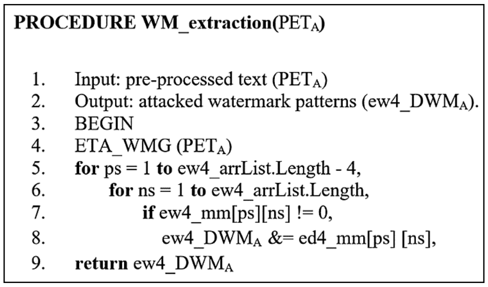
Before the detection of the pre-proceed attacked English text (PET_A), attacked watermark patterns (ew4_DWM_A) should be generated, and the matching rate of patterns and watermark distortion should be calculated by HTAZWA for detecting any tamper with the authentication of the given contents.

Two core algorithms are involved in this process, which are watermark extraction and watermark detection. However, ew4_DWM_A will be extracted from the received (PET_A) and matched with ew4_DWM_O by the detection algorithm. PET_A should be provided as the input for the proposed watermark extraction algorithm. The same process of watermark generation algorithm should be performed to obtain the watermark pattern for (PET_A) as illustrated in Fig. 5.

3.2.1 Algorithm of Watermark Extraction

PET_A is the main input required to run this algorithm. However, the output of this algorithm is ew4_DWM_A. The watermark extraction algorithm is presented and executed as illustrated in Algorithm 4.

Algorithm 4. Algorithm of watermark extraction based HTAZWA



```
PROCEDURE WM_extraction($PET_A$)

Input: pre-processed text ($PET_A$)
Output: attacked watermark patterns (ew4_$DWM_A$).
BEGIN
ETA_WMG ($PET_A$)
for ps = 1 to ew4_arrList.Length - 4,
    for ns = 1 to ew4_arrList.Length,
        if ew4_mm[ps][ns] != 0,
            ew4_$DWM_A$ &= ed4_mm[ps] [ns],
return ew4_$DWM_A$
```

where, PET_A: pre-processed attacked English text and ew4_DWM_A: Attacked digital watermark.

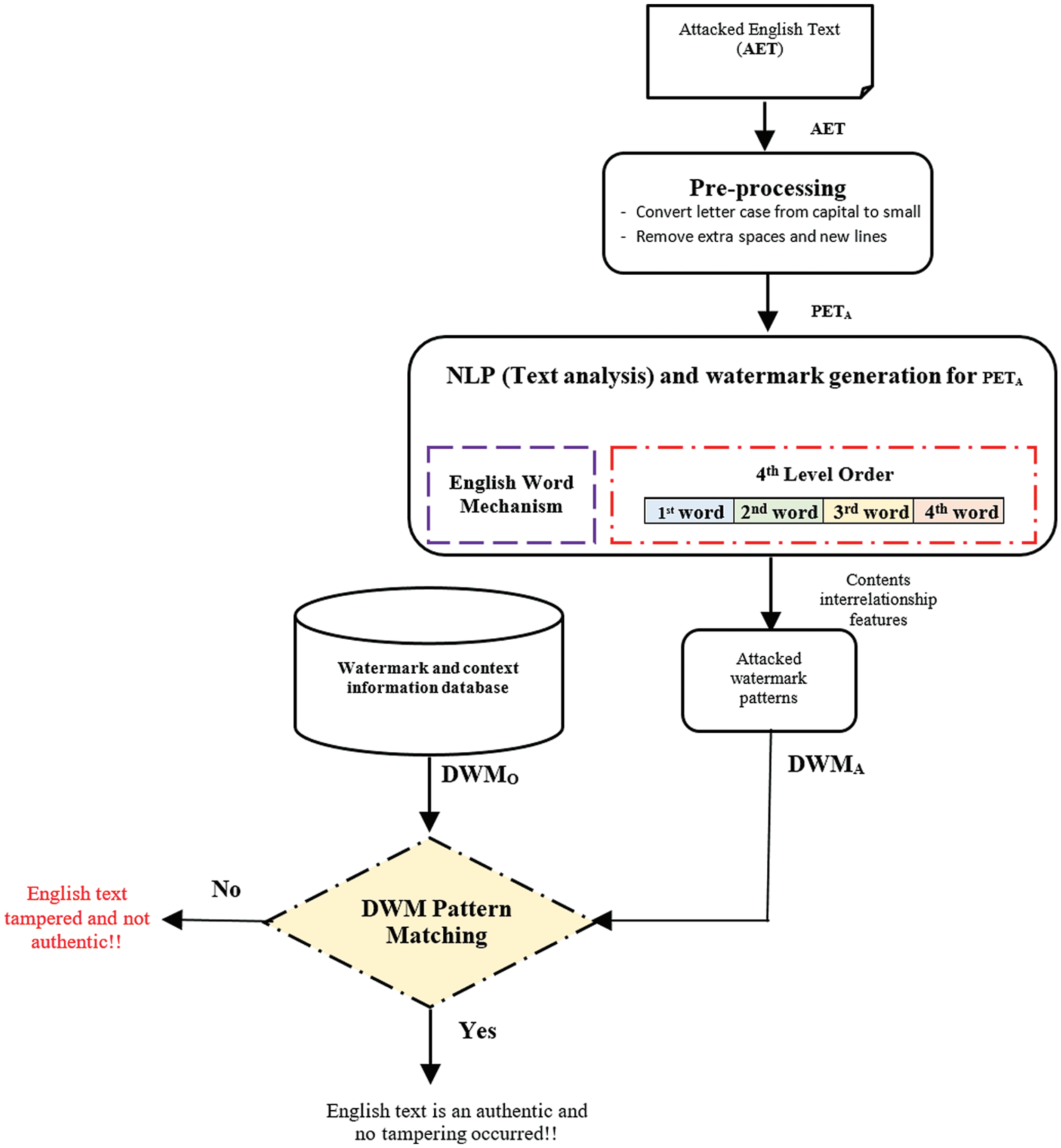

Attacked English Text
(AET)
AET
Pre-processing
- Convert letter case from capital to small
- Remove extra spaces and new lines
PETA
NLP (Text analysis) and watermark generation for PETA
English Word Mechanism
4th Level Order
1st word
2nd word
3rd word
4th word
Contents interrelationship features
Watermark and context information database
Attacked watermark patterns
DWMO
DWMA
DWM Pattern Matching
No
English text tampered and not authentic!!
Yes
English text is an authentic and no tampering occurred!!

Figure 5: Zero-watermark HTAZWA processes of extraction and detection

3.2.2 Algorithm of Watermark Detection

ew4_DWM$_A$ and ew4_DWM$_O$ are the main inputs required to run this algorithm, while the output of this algorithm is the notification English text status, which can be authentic or tampered. The detection process of an extracted watermark is achieved in two main phases:

- *Primary matching* is achieved for ew4_DWM$_O$ and ew4_DWM$_A$. If these two patterns appear identical, then an alert will appear as "English text is authentic, and no tampering occurred." Otherwise, the notification will be "English text is tampered and not authentic," and then it continues to the next step.

- *Secondary matching* is achieved by matching the transition of each state in the whole generated pattern. This means ew4_DWM$_A$ of each state will be compared to the equivalent transition of ew4_DWM$_O$ as given by Eqs. (3) and (4).

$$ew4\_TDA_T\,(i,j) = \left|\frac{ew4\_DWM_O\,[i]\,[j] - (ew4\_DWM_O\,[i]\,[j] - ew4\_DWM_A\,[i]\,[j])}{ew4\_DWM_O\,[i]\,[j]}\right|, \quad \textit{for all } i,\ j \textit{ states and transitions} \tag{3}$$

where,

—ew4_TDA$_T$: represents tampering detection accuracy rate value in transition level, (0 < ew4_TDA$_T$ <= 1)
—ew4_DWM$_O$: refers to original watermark value in transition level.
—ew4_DWM$_A$: refers to attacked watermark value in transition level.

$$ew4\_TDA_S\,(i) = \left|\frac{\sum_{j=1}^{n-4}(ew4\_TDA_T\,(i,j))}{TotalStatePatternCount\,(i)}\right| \quad \textit{for all } i \tag{4}$$

where, ew4_TDA$_S$: is value of tampering detection accuracy rate in state level, (0 < ew4_TDA$_S$ ≤ 100).

After the tampering detection accuracy rate of every state has been produced; the weight of every state stored in the Markov matrix should be found as presented in Eq. (5).

$$ew4\_Sw = \left|\frac{ew4\_TDA_S(i) * Transitions\ frequency(i)}{total\ number\ of\ transitions}\right| \tag{5}$$

where,

—ew4_TDA$_S$: is the total tampering detection accuracy rate in state level for each unique sequence of four words.
—i: is the count of all states in the given English text.

The final ew4_TDA of PET_A and OET_P are calculated by Eq. (6).

$$ew4\_TDA = \left|\frac{\sum_{i=1}^{n-4} ew4\_TDA_S(i)}{N}\right| \tag{6}$$

where,

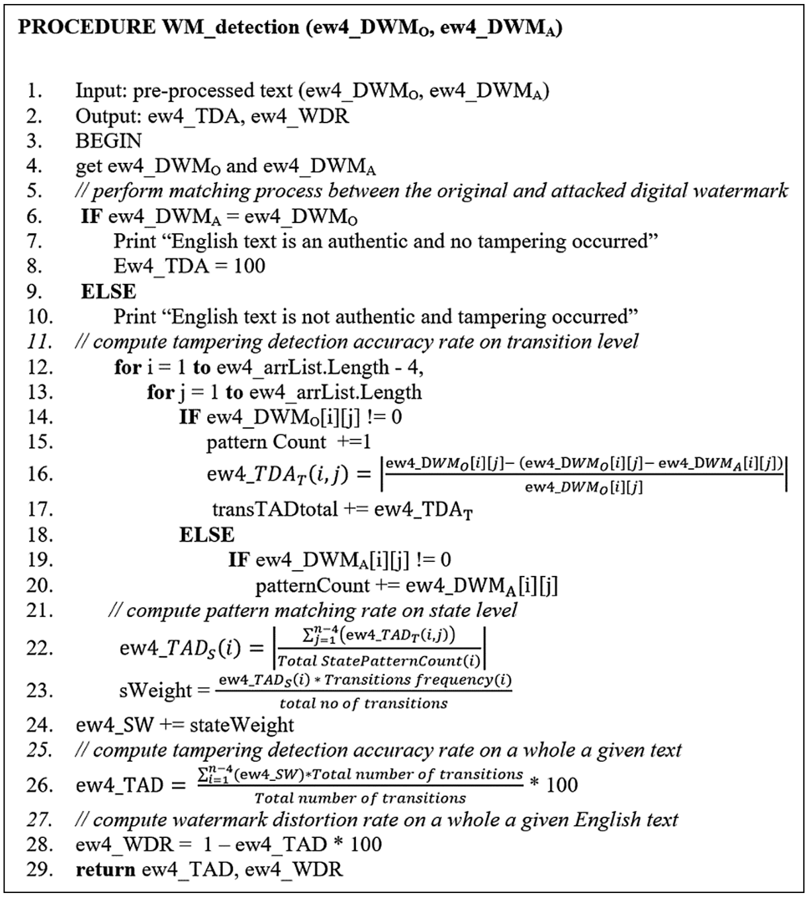
—ew4_TDA: is the total tampering detection accuracy rate of a whole given text.
—N: represents the total number of non-zeros values in ew4_mm.

The rate of watermark distortion represents the amount of tampering attacks occurring on the contents of the attacked English context, which is denoted by ew4_WDR and calculated by Eq. (7).

$$ew4\_WDR = 1 - ew4\_TDA * 100 \tag{7}$$

The watermark detection algorithm is presented formally and executed as illustrated in Algorithm 5.

Algorithm 5. Algorithm of watermark detection using HTAZWA



```
PROCEDURE WM_detection (ew4_DWM_O, ew4_DWM_A)

Input: pre-processed text (ew4_DWM_O, ew4_DWM_A)
Output: ew4_TDA, ew4_WDR
BEGIN
get ew4_DWM_O and ew4_DWM_A
// perform matching process between the original and attacked digital watermark
 IF ew4_DWM_A = ew4_DWM_O
     Print "English text is an authentic and no tampering occurred"
     Ew4_TDA = 100
 ELSE
     Print "English text is not authentic and tampering occurred"
// compute tampering detection accuracy rate on transition level
     for i = 1 to ew4_arrList.Length - 4,
         for j = 1 to ew4_arrList.Length
             IF ew4_DWM_O[i][j] != 0
                 pattern Count  +=1
                 $ew4\_TDA_T(i,j) = \left|\frac{ew4\_DWM_O[i][j] - (ew4\_DWM_O[i][j] - ew4\_DWM_A[i][j])}{ew4\_DWM_O[i][j]}\right|$
                 transTADtotal += ew4_TDA_T
             ELSE
                 IF ew4_DWM_A[i][j] != 0
                     patternCount += ew4_DWM_A[i][j]
         // compute pattern matching rate on state level
     $ew4\_TAD_S(i) = \left|\frac{\sum_{j=1}^{n-4}(ew4\_TAD_T(i,j))}{Total\ StatePatternCount(i)}\right|$
     $sWeight = \frac{ew4\_TAD_S(i) * Transitions\ frequency(i)}{total\ no\ of\ transitions}$
ew4_SW += stateWeight
// compute tampering detection accuracy rate on a whole a given text
$ew4\_TAD = \frac{\sum_{i=1}^{n-4}(ew4\_SW) * Total\ number\ of\ transitions}{Total\ number\ of\ transitions} * 100$
// compute watermark distortion rate on a whole a given English text
ew4_WDR =  1 – ew4_TAD * 100
return ew4_TAD, ew4_WDR
```

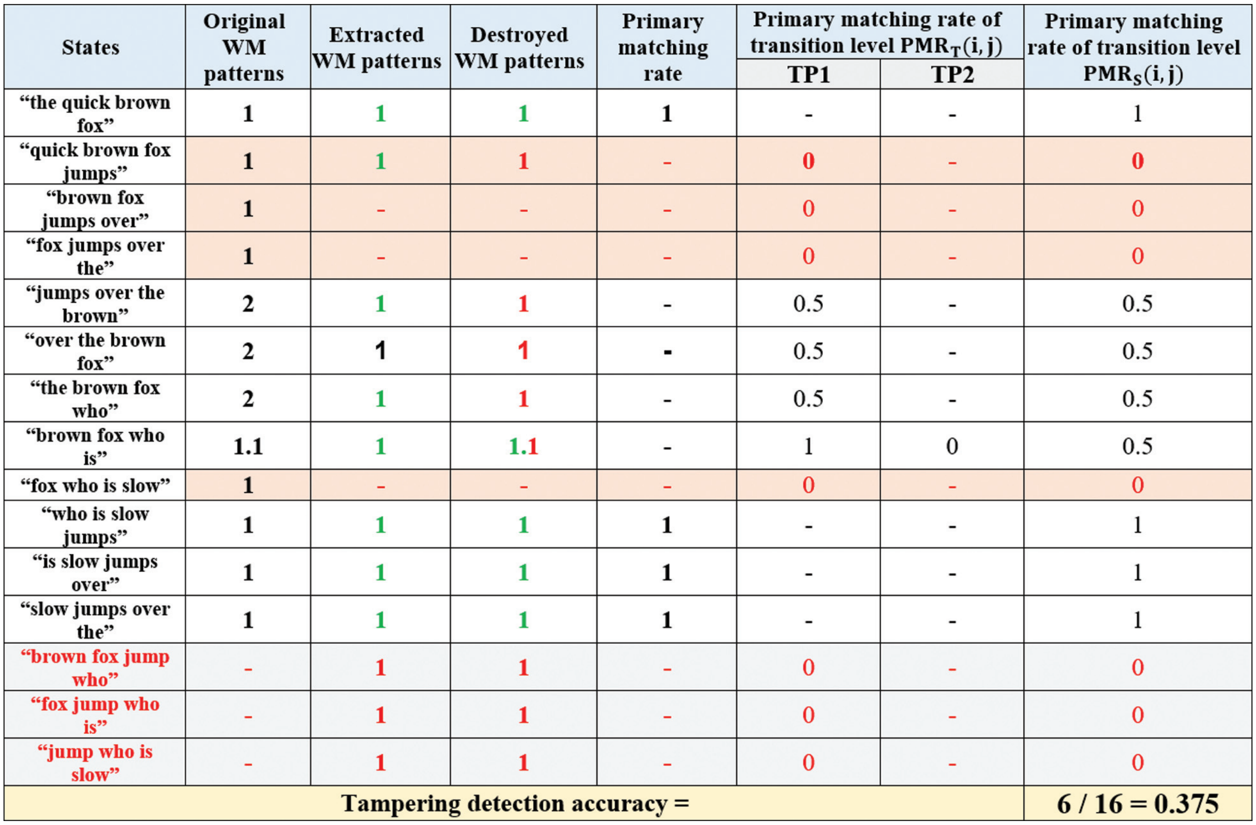
where, ew4SW: refers to the weight value of states matched correctly and ew4_WDR: refers to the value of watermark distortion rate ($0 < ew4\_WDR_S \leq 100$).

The results of the watermark extraction and detection process are illustrated in Fig. 6.

As shown in Fig. 6, TP1 represents 1st transition of non-zero in the given English text, TP2 represents 2nd transition, and so on. Some states have only one transition, which is shown in TP1. However, some states have more than one transition, which are represented in TP1, TP2, … etc. such as "brown fox who is."

4 Implementation and Simulation

A variety of implementation and simulation are conducted to test the accuracy of HTAZWA output and tampering detection. This section outlines the settings for implementation and experimentation, conditions for experiments, typical dataset experimental scenarios, and a discussion of outcomes.



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 fox"	1	1	1	1	-	-	1
"quick brown fox jumps"	1	1	1	-	0	-	0
"brown fox jumps over"	1	-	-	-	0	-	0
"fox jumps over the"	1	-	-	-	0	-	0
"jumps over the brown"	2	1	1	-	0.5	-	0.5
"over the brown fox"	2	1	1	-	0.5	-	0.5
"the brown fox who"	2	1	1	-	0.5	-	0.5
"brown fox who is"	1.1	1	1.1	-	1	0	0.5
"fox who is slow"	1	-	-	-	0	-	0
"who is slow jumps"	1	1	1	1	-	-	1
"is slow jumps over"	1	1	1	1	-	-	1
"slow jumps over the"	1	1	1	1	-	-	1
"brown fox jump who"	-	1	1	-	0	-	0
"fox jump who is"	-	1	1	-	0	-	0
"jump who is slow"	-	1	1	-	0	-	0
Tampering detection accuracy =							**6 / 16 = 0.375**

Figure 6: Results of watermark extraction and detection process using HTAZWA

4.1 Simulation and Implementation Environment

The self-developed software was developed to evaluate and assess the efficiency of HTAZWA. The HTAZWA implementing environment is: CPU: Intel Core i7-4650U/2.3 GHz, RAM: 8.0 GB, Windows 10–64 bit, PHP VS Code IDE programming language.

4.2 Simulation and Experimental Parameters

Tab. 1 shows the experimental and simulation parameters and their associated values that were used to perform the experiments of the proposed HTAZWA approach.

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
ew4_TDA	(H if ew4_TDA > 70, M if 40 < ew4_TDA < 70, and L if ew4_TDA < 40)

4.3 Performance Metrics

The tampering detection accuracy refers to the performance of HTAZWA which is evaluated by the metrics below:

- Precision for tampering identification (ew4_TDA) in all addressed assaults in all scenarios in English data size with many normal attack volumes of very low volume (5 percent), low volume (10 percent), medium volume (20 percent), and high volume (50 percent)
- Accuracy of tampering detection refers to the performance and watermark fragility which its desired value is close to 100.
- Diagnosis and outcome assessment of the influence of dataset size, the impact of attack types, and attack volume with the suggested HTAZWA method.

4.4 HTAZWA Simulation and Experiment Findings

In this portion, the author tests the accuracy of HTAZWA tampering detection. This dataset includes all English characters, numbers, spaces, and symbols. Experiments have been conducted with distinct dataset size and various kinds of frequency attacks as seen above in Tab. 1.

4.5 Evaluation of Tampering Detection Accuracy (All Volumes)

To evaluate the accuracy of tampering detection of HTAZWA, scenarios of many studies are performed as shown in Tab. 2 for all forms of attacks and their volumes.

Table 2: Accuracy of tampering detection under all volumes

Attack volume (%)	Insertion	Deletion	Reorder
5	93.75	90.53	78.13
10	89.50	83.33	64.45
20	81.47	72.01	45.28
s50	64.49	40.65	20.52

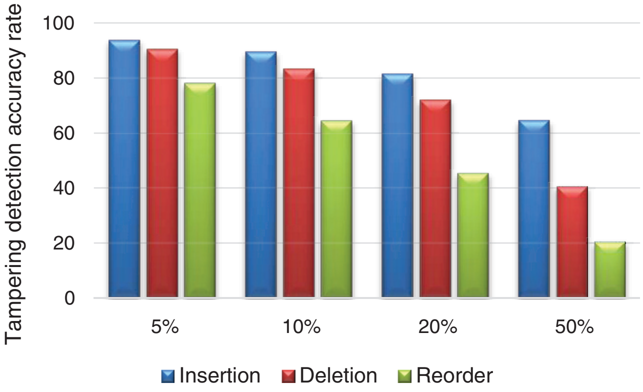


Figure 7: Tampering impact under all attacks of many volumes

The results shown in Tab. 2 and Fig. 7 show the insertion attack outperforms deletion and reorder attacks in all cases of attack volumes 5%, 10%, 20%, and 50%. Results show also deletion attack outperforms reorder attack in all cases of attack volumes. In other meaning, the low effect is detected under insertion and deletion attacks and the high effect is detected under reorder attack in all experimental scenarios under all attack volumes. This means that HTAZWA gives the best detection accuracy under insertion attack in all scenarios of attack volumes.

5 Comparison and Result Discussion

The tampering detection accuracy results were critically analysed. The study impact is investigated and comparisons between HTAZWA and baseline approaches ZWAFWMMM and HNLPZWA showed their effect under the major factors (i.e., dataset size, attack types, and volumes).

5.1 Baseline Approaches

The efficiency and accuracy of HTAZWA were contrasted with HNLPZWA (an intelligent hybrid of natural language processing and zero-watermarking approach) [5] and ZWAFWMM (zero watermark approach based on the fourth Markov model order and Arabic word mechanism) [30]. Comparisons are made under all success indicators to assess which method has the best accuracy in the identification of tampering. The following sub-sections describe the basic methods and their operating parameters.

5.2 Comparison Results

This subdivision provides a comparison of HTAZWA, ZWAFWMMM and HNLPZWA approaches and analyses their results in conjunction with key affected variables i.e., scale of datasets, kinds of attacks, and volumes.

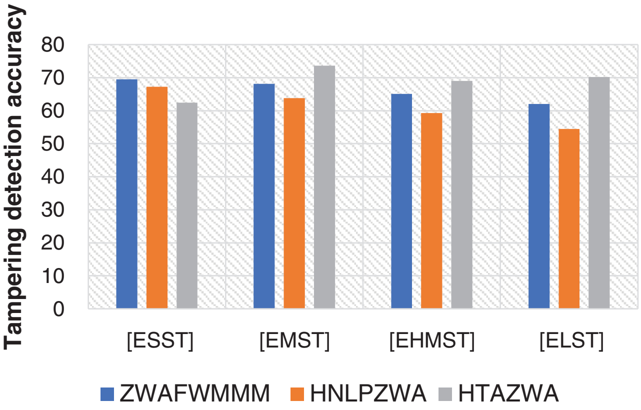


Figure 8: Dataset impact on tampering detection accuracy based on HTAZWA and other approaches

5.2.1 Comparison Results of Dataset Size Impact

The comparative results shown in Fig. 8 reflect the tampering detection accuracy of HTAZWA approach. The results show that in the proposed HTAZWA 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 the case of reordering attacks. This means that the tampering detection accuracy increased with decreasing document size and decreased with increasing document size. On the other hand, results show that HTAZWA approach outperforms both ZWAFWMMM and HNLPZWA approaches in terms of tampering detection accuracy under all scenarios of mid and large dataset size (EMST, EHMST, and ELST) except in the case of small dataset size (ESST).

5.2.2 Comparison Results of Attack Type Impact

Fig. 9 shows how the tampering detection accuracy of HTAZWA, ZWAFWMMM, and HNLPZWA approaches are influenced by the type of tampering attacks. In all cases of insertion, deletion, and reorder attacks, HTAZWA outperforms ZWAFWMMM and HNLPZWA approaches with a high rate of tampering detection accuracy. This means that the proposed HTAZWA approach is strongly recommended and applicable for content authentication and tampering detection of English text under all attack types.

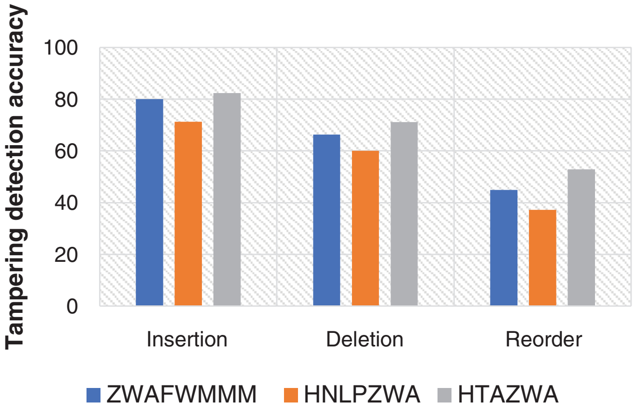


Figure 9: Attack impact on tampering detection accuracy based on HTAZWA and baseline approaches

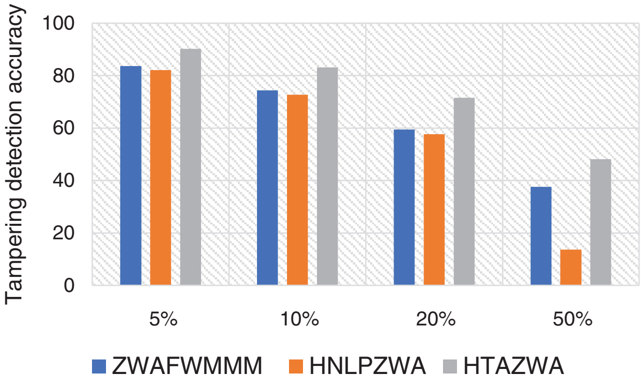


Figure 10: Attack rates impact on tampering detection accuracy of HTAZWA and baseline approaches

5.2.3 *Comparison Results of Attack Volume Impact*

Fig. 10 shows how the tampering detection accuracy is influenced by low, mid, and high attack volumes. In all cases of HTAZWA, ZWAFWMMM, and HNLPZWA approaches, it has been 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 has been seen HTAZWA outperforms ZWAFWMMM and HNLPZWA approaches in terms of tampering detection accuracy in all scenarios of low, mid, and high volumes of all attacks. This means that HTAZWA approach is strongly recommended and applicable for content authentication and tampering detection of English text documents under all volumes of all attacks.

6 Conclusion

HTAZWA approach is implemented in PHP programming language using VS code IDE. The experiments are performed on various standard datasets under different volumes of insertion, deletion, and reorder attacks. HTAZWA has been compared with ZWAFWMMM and HNLPZWA approaches. The comparison results show that HTAZWA outperforms ZWAFWMMM and HNLPZWA approaches in terms of the accuracy of tampering detection. The results also show that HTAZWA applies to all English alphabetic letters, special characters, numbers, and spaces. For future work, the improvement of detection accuracy should be considered for all kinds of attacks.

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

[1] F. N. Al-Wesabi, “A smart English text zero-watermarking approach based on third-level order and word mechanism of Markov model,” *Computers, Materials & Continua*, vol. 65, no. 2, pp. 1137–1156, 2020.

[2] M. Abd-Eldayem, “A proposed security technique based on watermarking and encryption for digital imaging and communications in medicine,” *Egyptian Informatics Journal*, vol. 14, no. 1, pp. 1–13, 2013.

[3] F. N. Al-Wesabi, “A hybrid intelligent approach for content authentication and tampering detection of Arabic text transmitted via Internet,” *Computers, Materials & Continua*, vol. 66, no. 1, pp. 195–2011, 2021.

[4] S. G. Rizzo, F. Bertini and D. Montesi, “Fine-grain watermarking for intellectual property protection,” *EURASIP Journal on Information Security*, vol. 10, no. 1, pp. 804, 2019.

[5] F. N. Al-Wesabi, K. Mahmood and N. Nemri, “A zero watermarking approach for content authentication and tampering detection of Arabic text based on fourth level order and word mechanism of Markov model,” *Journal of Information Security and Applications*, vol. 52, no. 1, pp. 1–15, 2020.

[6] Y. Cao, Z. Zhou, Q. Wu, C. Yuan and X. Sun, “Coverless information hiding based on the generation of anime characters,” *EURASIP Journal on Image and Video Processing*, vol. 36, 2020.

[7] Y. Cao, Z. Zhou, X. Sun and C. Gao, “Coverless information hiding based on the molecular structure images of material,” *Computers, Materials and Continua*, vol. 54, no. 2, pp. 197–207, 2018.

[8] Y. Cao, Z. Zhou, C. Yang and X. Sun, “Dynamic content selection framework applied to coverless information hiding,” *Journal of Internet Technology*, vol. 19, no. 4, pp. 1179–1186, 2018.

[9] C. Qin, C. Chang and T. Hsu, "Fragile watermarking for image authentication with high-quality recovery capability," *KSII Transactions on Internet and Information Systems*, vol. 7, no. 11, pp. 2941–2956, 2013.

[10] S. Parah, J. Sheikh and G. Bhat, *StegNmark: A Joint Stego-watermark Approach for Early Tamper Detection*. Vol. 660. Switzerland: Springer International Publishing, pp. 427–452, 2017.

[11] S. Hakak, A. Kamsin, O. Tayan, M. Yamani and G. Gilkar, "Approaches for preserving content integrity of sensitive online Arabic content," *Information Processing and Management*, vol. 56, no. 2, pp. 367–380, 2019.

[12] M. Taleby, Q. Li, X. Zhu, M. Alazab and J. Zhang, " A Novel intelligent text watermarking technique for forensic identification of information on social media," *Computers and Security*, vol. 90, pp. 1–14, 2020.

[13] S. Parah, J. Sheikh, J. Akhoon and N. Loan, "Electronic health record hiding in images for smart city applications: A computationally efficient and reversible information hiding technique for secure communication," *Future Generation Computer Systems*, vol. 108, no. 6, pp. 935–949, 2020.

[14] R. Ahmed and L. Elrefaei, "Arabic text watermarking: A review," *International Journal of Artificial Intelligence & Applications*, vol. 6, no. 4, pp. 1–16, 2015.

[15] K. Hameed, A. Khan, M. Ahmed and A. G. Reddy, "Towards a formally verified zero watermarking scheme for data integrity in the internet of things based-wireless sensor networks," *Future Generation Computer Systems*, vol. 167, pp. 1–16, 2018.

[16] R. Alotaibi and L. Elrefaei, "Improved capacity text watermarking methods based on open word space," *Journal of King Saud University—Computer and Information Sciences*, vol. 30, no. 2, pp. 236–248, 2018.

[17] M. Memon and A. Shah, "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, 2015.

[18] Y. Alginahi, M. Kabir and O. Tayan, "An enhanced Kashida-based watermarking approach for increased protection in Arabic text-documents based on frequency recurrence of characters," *International Journal of Computer and Electrical Engineering*, vol. 6, no. 5, pp. 381–392, 2014.

[19] A. Shaker, F. Ridzuan and S. Pitchay, "Text steganography using extensions Kashida based on moon and sun letters," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 8, pp. 286–290, 2017.

[20] A. Rahma, W. Bhaya and D. Al-Nasrawi, "Text steganography based on unicode of characters in multilingual," *Journal of Engineering Research and Applications*, vol. 3, no. 4, pp. 1153–1165, 2013.

[21] N. Al-maweri, W. Adnan, A. Rahman, S. Khair and S. Syed, "Robust digital text watermarking algorithm based on unicode characters," *Indian Journal of Science and Technology*, vol. 9, no. 48, pp. 1–14, 2016.

[22] M. Bashardoost, M. Rahim, T. Saba and A. Rehman, "Replacement attack: A new zero text watermarking attack," *3D Research*, vol. 8, no. 1, 2017. https://doi.org/10.1007/s13319-017-0118-y.

[23] Y. Liu, Y. Zhu and G. Xin, "A zero-watermarking algorithm based on merging features of sentences for Chinese text," *Journal of the Chinese Institute of Engineers*, vol. 38, no. 3, pp. 391–398, 2015.

[24] P. Zhu, W. Song, A. Li, Y. Zhang and R. Tao, "A text zero watermarking algorithm based on Chinese phonetic alphabets," *Wuhan University Journal of Natural Sciences*, vol. 21, no. 4, pp. 277–282, 2016.

[25] Z. Ali, M. Shamim, G. Muhammad and M. Aslam, "New zero-watermarking algorithm using hurst exponent for protection of privacy in telemedicine," *IEEE Access*, vol. 6, pp. 7930–7940, 2018.

[26] O. Tayan, Y. Alginahi and M. Kabir, "An adaptive zero-watermarking approach for text documents protection," *International Journal of Image Processing Techniques*, vol. 1, no. 1, pp. 33–36, 2014.

[27] M. Ghilan, F. Ba-Alwi and F. N. Al-Wesabi, "Combined Markov model and zero watermarking to enhance authentication of Arabic text," *Journal of Computational Linguistics Research*, vol. 5, no. 1, pp. 26–42, 2014.

[28] F. N. Al-Wesabi, A. Alsakaf and K. U. Vasantrao, "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, 2013.

[29] H. Ahmed and M. Khodher, “Comparison of eight proposed security methods using linguistic steganography text,” *Journal of Computing & Information Sciences*, vol. 12, no. 2, pp. 243–251, 2016.

[30] F. N. Al-Wesabi, “Proposing high-smart approach for content authentication and tampering detection of Arabic text transmitted via Internet,” *IEICE transactions in Information Systems*, vol. E103, no. 10, 2020.