Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2021.014322
Article

Tampering Detection Approach of Arabic-Text Based on Contents Interrelationship

Fahd N. Al-Wesabi[1], Abdelzahir Abdelmaboud[2,*], Adnan A. Zain[3], Mohammed M. Almazah[4] and Ammar Zahary[5]

[1]Department of Computer Science, King Khalid University, Muhayel Aseer, KSA & Faculty of Computer and IT, Sana'a University, Yemen
[2]Department of Information Systems, King Khalid University, Mayahel Aseer, KSA
[3]Department of Electronic & Communication Engineering, Faculty of Engineering, University of Aden, Aden, Yemen
[4]Department of Mathematics and Computer, IBB University, IBB, Yemen & Department of Mathematics, KKU, KSA
[5]Faculty of Computer and IT, Sana'a University, Sana'a, Yemen
[*]Corresponding Author: Abdelzahir Abdelmaboud. Email: aelnour@kku.edu.sa
Received: 13 September 2020; Accepted: 28 October 2020

Abstract: Text information depends primarily on natural languages processing. Improving the security and usability of text information shared through the public internet has therefore been the most demanding problem facing researchers. In contact and knowledge sharing through the Internet, the authentication of content and the identification of digital content are becoming a key problem. Therefore, in this paper, a fragile approach of zero-watermarking based on natural language processing has been developed for authentication of content and prevention of misuse of Arabic texts distributed over the Internet. According to the proposed approach, watermark embedding, and identification was technically carried out such that the initial text document did not need to be changed to embed a watermark. The automated zero-watermarking approaches have been combined with a second-tier word order framework based on the Markov model to boost the efficiency, precision, ability, and fragility of the existing researches. This second-tier of the Markov-model has been used as a natural language processing technique to analyze Arabic-text and extract the features of the interrelationship between textual contexts. Moreover, the extracted features have been utilized as information of watermark and then validated to identify any possible tampering with the attacked Arabic-text. The recommended solution has been applied with VS code IDE using PHP. The experimental results using four datasets of varying size show that the proposed approach can obtain better detection accuracy of tampering attacks, effectiveness and high fragility for common random insertion, reorder and deletion attacks for common attacks, e.g., Comparison results with baseline approaches also show the advantages of the proposed approach.

Keywords: Text analysis; NLP; HMM; Arabic text processing; watermark fragility; tampering detection accuracy

1 Introduction

The reliability and security of text information shared over the Internet is the most exciting and demanding area for the scientific community. In communication technologies, authentication of content, and honesty of automated text verification with different Languages formats are great significance. Numerous applications such as electronic banking, electronic commerce, etc. impose most challenges during contents transfer via the internet. In terms of content, structure, grammar, and semantics, much of the multimedia exchanged via the Internet is in form of textual which is very susceptible to online transmission. During the transfer process, malicious attackers can temper such digital content and thus the changed count [1].

For information security, many algorithms and techniques are available, such as content authentication, verification of integrity, detection of tampering, identification of owners, access control, and copyright protection.

To overcome these issues, digital watermarking (DWM) is a technique that can be used to hide the various data, for example; binary images, text, audio, and video with embed them in digital content as watermark information [2,3].

A fine-grain process for document watermarking is suggested based on the substitution of homoglyph characters of white spaces and Latin symbols [4].

Several conventional methods and solutions for text watermarking were proposed [5] and categorized into different classes, such as linguistic, structure, zero watermarks, and format-based methods [6]. To insert the watermark information into the document, most of these methods need improvement to plain text material. Zero-watermarking without any alteration to the original digital material to embed the watermark information is a recent technology used with intelligent methods and algorithms. In addition, the contents of a given digital background can be used in this process to produce the watermark key [1,6–8].

Restricted research has centered on the appropriate solutions to verify the credibility of critical digital media online [9–11]. In the research community, digital text tampering detection and authentication have received great attention. Also, research in the field of text watermarking has 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 for various formats and content, especially in English and Arabic languages, is the most common challenge in this area [13,14]. Therefore, authentication of content, verification of honesty, and detection of tampering of sensitive text is a major problem in different applications and needs required solutions.

Some instances of such sensitive digital text content are Arabic Holy-Qur'an, eChecks, and online marking of exams. Different Arabic alphabet characteristics such as diacritics, letters extraction, and symbols of Arabic supplementary that make it easy to alter the key text material meaning by creating basic changes such as modifying diacritic arrangements [11,15]. The most popular soft computing and Natural-Language Processing (NLP) technique which is involved for HMM text analysis.

This paper, present Fragile Arabic-Text Zero-Watermarking approach based on NLP (FATZWNLP). The FATZWNLP approach is focused on the word order mechanism of the second rank, focused on the Markov model for material authentication and identification of Arabic-text transmitted over the Internet. It includes of a model that functions as NLP and soft computing methods in cooperation between the zero-watermarking system and the Markov-model. The second-order term method was included in this approach to extract the interconnections between of the Arabic-text contents for text analysis and to produce a key watermark. Without any alteration or influence mostly on scale of a original text, the created watermark is logically

incorporated with in original Arabic sense. The embedded watermark would later be used to identify any tampering that happens on the Arabic-text obtained and to determine whether it is genuine or not.

The main goal of the FATZWNLP strategy is to meet high precision of authentication of content and detection of Arabic-text sensitive attack tampering that is distributed over the Internet.

As follows, the remainder of the paper is structured. The current works completed so far are clarified in Section 2. FATZWNLP is discussed in Section 3. Implementation, study of experimental setup, findings, and discussion are defined in Section 4. Finally, an inference is reached in Section 5.

2 Related Works

According to the processing domain of NLP and text watermarking, these existing methods and solutions of text watermarking reviewed in this paper classified into linguistic, structural, and zero-watermark techniques [1,6,12].

2.1 Linguistic Techniques

The techniques to watermarking of linguistic text are based upon natural language to hide the watermark key by making some altering on the syntactic and semantic nature of the original text [1].

Watermarking text open-word spaces-based algorithm has been proposed to increase the capability and imperceptibility of Arabic text [16]. In this method, every word-space is used to embed binary data 1 or 0 to obtain the physical altering that occurred on plain text.

A technique of text steganography [17] has been suggested to conceal details in the Arabic-language. The process of this algorithm considers the presence in Arabic of Harakat (diacritics, i.e., Kasra, Fatha, and Damma) and reverses the Fatha for the hiding of the message.

A Kashida-watermark based method has been presented in [18], which is frequency recurrence is utilized to get the document features. This method used a predefined watermark data whereby a Kashida is positioned for bit 1 and excluded for bit 0.

The method of text steganography [19] has been proposed to use Kashida extensions based on the characters ‘sun’ and ‘moon’ to write digital contents of the Arabic language. In this process, Kashida characters are used beside Arabic letters to decide which hidden secret bits are kept by specific characters. In this form, four situations are involved for 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 scripts of English letters using the English Unicode alphabet in other languages. Thirteen letters of the English alphabet have been chosen for this approach. Two bits would be hidden in a time frame. Used ASCII code for embedding00. However, for embedding 01, 10, and 11, Unicode involved multilingual ones.

2.2 Zero-Watermark Techniques

Several techniques and ideas based on watermarking text have been suggested in literature focused on text functionality and features [21–24]. For instance, the authors in Ref. [21] presents a zero-based watermarking method to maintain the Internet’s data security for items where a watermark is inserted in plain-text before being sent to base stations. The created watermark key is focused on certain content characteristics, such as data size, frequency of data presence, and data capture time. Ref. [22] offers a zero-watermarking solution to resolving the issues involved with the preservation of English text records, such as authentication of content and copyright protection. In addition, Ref. [23] provides a Markov model-based zero-watermarking for material authentication of English text, where the possibility

characteristics of the English text have been used to extract and store safe watermark information to verify the validity of the attacked text message. Moreover, the authors in Ref. [24] suggest a conventional method of watermarking English text copyright rights, based on the frequency of occurrence of – anti-vowel ASCII words and letters.

2.3 Hybrid Techniques

Authors in Ref. [25] present a zero-watermarking method to protect the identification of persons with vocal fold defects. The method proposed is to create a picture of the identity of an individual and incorporate everything into watermark information to verify positions by measuring local binary patterns from the time-frequency continuum. In Ref. [26], the same strategy is suggested to use a separate method by which the identity of a patient is secured by producing the hidden shares through visual cryptography. Reference [27], presents an authentication scheme by using semi-fragile watermarking for content authentication of images from malicious attacks. The authors in Refs. [28,29] recommend zero-watermark image authentication methods by integrating scheme of zero-watermark with geometric invariants. In the proposed methods change in original images not necessary. Finally, a hybrid image and text watermarking approach for preserving an English text document is given in reference [30]. In order to validate the ownership of content, the process of this approach relies on inserting a watermark key literally throughout the text that is later extracted.

3 Proposed Approach

This paper introduces a new intelligent approach named FATZWNLP by the combination of zero-watermark and NLP strategies in which no external details like the watermark key needs to be embedded or otherwise changed in the original text. The second tier ordering of the word method of the Markov-model was used to examine the Arabic-text contents and derive the interrelationship characteristics of such NLP technique and text contents. In FATZWNLP, two major processes should be conducted, which are the process of watermark creation and embedding as well as the process of extraction and identification of watermarks shown in Fig. 1.

The following paragraphs describe the generation and extraction processes of the watermark in detail.

3.1 Watermark Generation and Embedding Algorithms

The pre-processing process can be carried out to eliminate any additional spaces and newlines in the provided Arabic-text before the generation of watermark and inserting process. Input for the generation of watermarks and embedding algorithms includes a pre-processed original document of Arabic-text. The initial watermark pattern (W2 AWM_O) is the performance of this algorithm. The created watermark would be recorded in the WM database alongside simple Arabic text document information, including the name of the author, the size and identity of the document and the last updated date.

In this step, the three major sub-algorithms included development and pre-processing of text analysis, Markov matrix, and generation and embedding of watermark.

3.1.1 Markov Chain Algorithm

For the pre-processing process to delete additional spaces and newlines, the original Arabic-text (OAT) is needed as an input. Building a Markov matrix is the starting point of Arabic text analysis and watermark generation process using the Markov model. Without reputation, a Markov chain is constructed that describes the potential states and transformations available in a given document. In this method, each specific set of words defines a present situation within a provided Arabic text, and each similar word expresses a transformation in the matrix of Markov. The presented algorithm identifies all transformation values by

zero during the building step of the Markov matrix using these cells late to record the number of times the ^{ith} pair of words is preceded by the ^{jth} word inside the specified Arabic text document. As presented below in Alg. 1, pre-processing and constructing the algorithm of Markov matrix is performed.

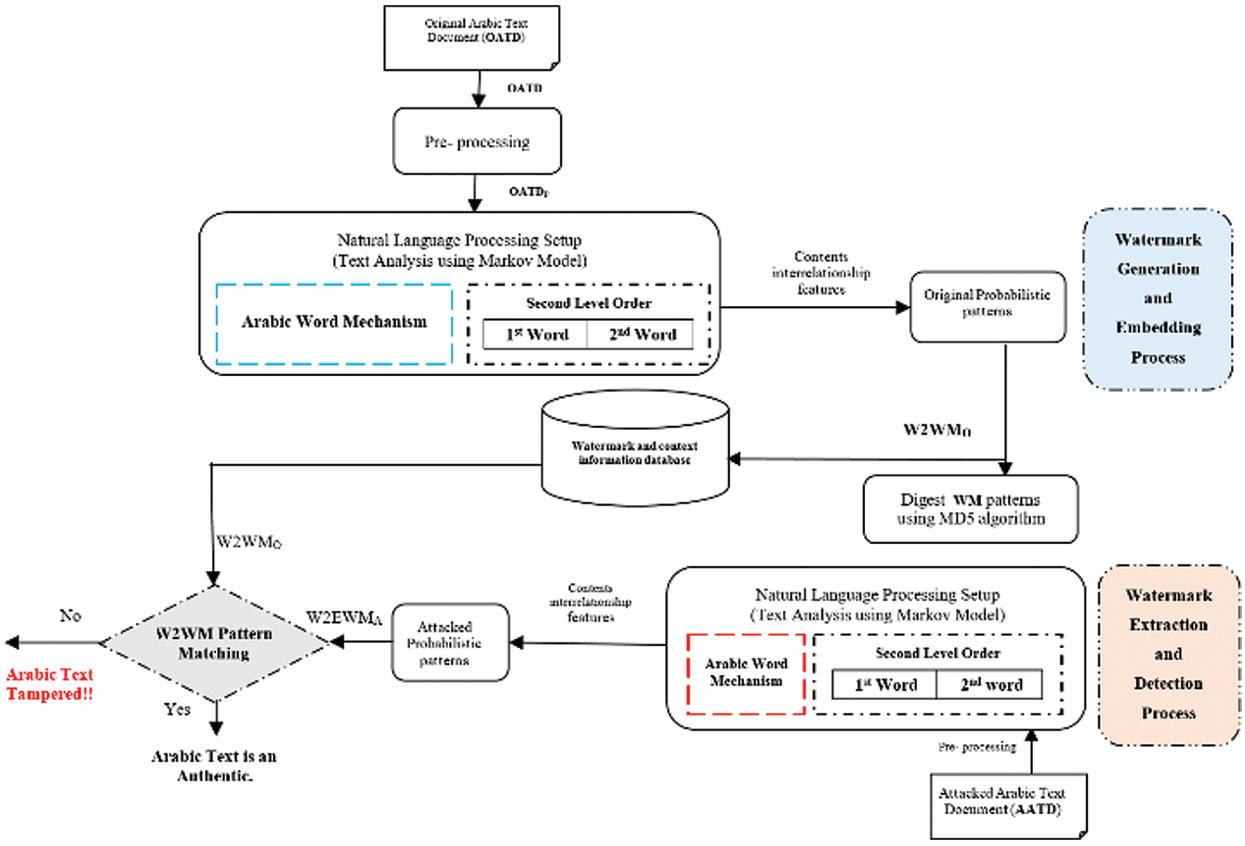

Figure 1: FATZWNLP approach of Arabic-text

PROCEDURE **Prep_Building_MM** (OAT)

- Input: original Arabic text (OAT)
- Output: Markov matrix with zeros initial value
- BEGIN
- // perform pre-processing process
- **for each** word **in** OAT
  - PAT ← trim ("space" or "newLine")
- // Build list of non values text words
- w2_mm = { }
- **for each** word **in** PAT
  - **if** word not in w2_list
    - w2_mm ← w2_mm **U** {word}
  - **for** ps = 1 **to** w2_mm.length – 2
    - **for** ns = 1 **to** w2_mm.length
      - w2_mm[ps][ns] = 0
- **return** w2_mm

Algorithm 1: Pre-processing FATZWNLP algorithm

Hence, OAT: represent Arabic-text originality, PAT: represent Arabic-text pre-processed, w2_mm: represent transitions and states matrix based zeros value for defining cells, ps: represent the current state, ns: represent the next state.

A procedure for constructing a matrix of two-dimensional of Markov-states and transformations called W2 mm[i][j], which describes the backbone of the Markov-model for analyzing Arabic-text, and providing conjunction with the above.

The W2 mm[i][j] matrix duration of the FATZWNLP is a complex matrix in which the majority of states differs according to the meaning of the defined Arabic-text, which is proportional to the number of non-reputable terms in the pair.

3.1.2 Algorithm of Watermark Generation

Just after Markov matrix has been created to identify the interrelationships between the meaning of the given Arabic text and generate watermark patterns, the NLP of the text analytical technique must be carried out.

The following example of an Arabic sample of texts explains the method of the phase of transformation from the current state to a new state.

"يقفز الثعلب البني السريع فوق الثعلب البني البطيء للوصول إلى الثعلب البني الميت"

When the hidden Markov-model uses the second-tier order of the word technique, every special pair of words is a current state technique. As the document is read to gain the interconnections between the current state and the next ones, document processing is processed. The transitions available from the above sample of the Arabic text are shown in Fig. 2 from below.

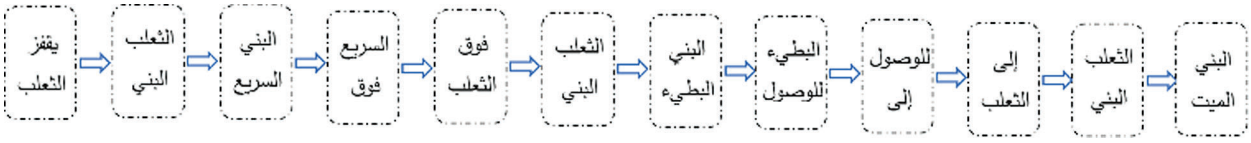

Figure 2: Sample of Arabic-text States based FATZWNLP

As a result of the study of the defined Arabic text using the Markov model's second tier order of word mechanism, authors describe all current states and their possible transformations, as shown in Fig. 3.

```
                                                                }
("يقفز الثعلب")        : [ ' البني'] ،
("الثعلب البني")       : [ ' السريع ، ' البطيء' ، ' الميت ' ]،
("البني السريع")       : [ ' فوق' ]،
("السريع فوق")         : [' الثعلب ' ]،
("فوق الثعلب")         : [ ' البني ' ] ،
("البني البطيء")       : [' للوصول ' ] ،
(" البطيء للوصول") : [' إلى ' ] ،
("للوصول إلى")         : [' الثعلب ' ] ،
("إلى الثعلب")         : [' البني ' ] ،
                                                                {
```

Figure 3: Arabic-text states sample based FATZWNLP

Authors assume that "الثعلب البني" is a present state, and the available next transitions are "البطيء" ,"السريع", and " الميت". We observe that there are three transitions available in the defined sample of Arabic-text.

In this algorithm, to every present state of a pair of terms, the number of occurrences of potential next system variables with greater than zero values will be computed and built as transformation probabilities by Eq. (1).

$$w2\_mm[ps][ns] = \sum_{i,j=1}^{n-2} Total\ Number\ of\ Transitions\ [i][j] \quad (1)$$

where,

- n: refer to total-number of states.
- i: refer to the i^{th} of the current-state of the pair-words.
- j: refer to the j^{th} of the next state-transition.

The watermark generation algorithm is related to the second-level order of the process of the word of the Markov model, as seen in Fig. 4

Present States (ps)	Avalible next states(ns) in the given Arabic text										
	يقفز	الثعلب	البني	السريع	فوق	البطيء	للوصول	إلى	الميت	Watermark patterns	Interrelation wheight
يقفز الثعلب	0	0	**1**	0	0	0	0	0	0	1	(0,0,**1/1**,0,0,0,0,0,0)
الثعلب البني	0	0	0	**1**	0	**1**	0	0	**1**	1.1.1	(0,0,0,**1/3**,0,**1/3**,0,0,**1/3**)
البني السريع	0	0	0	0	**1**	0	0	0	0	1	(0,0,0,0,**1/1**,0,0,0,0)
السريع فوق	0	1	0	0	0	0	0	0	0	1	(0,**1/1**,0,0,0,0,0,0,0)
فوق الثعلب	0	0	**1**	0	0	0	0	0	0	1	(0,0,**1/1**,0,0,0,0,0,0)
البني البطيء	0	0	0	0	0	0	**1**	0	0	1	(0,0,0,0,0,0,**1/1**,0,0)
البطيء للوصول	0	0	0	0	0	0	0	**1**	0	1	(0,0,0,0,0,0,0,**1/1**,0)
للوصول إلى	0	**1**	0	0	0	0	0	0	0	1	(0,**1/1**,0,0,0,0,0,0,0)
إلى الثعلب	0	0	**1**	0	0	0	0	0	0	1	(0,0,**1/1**,0,0,0,0,0,0)

Figure 4: of Sample Arabic-text processes based FATZWNLP

Let PATP represent text of pre-processed, w2 mm[ps][ns] defines the Markov-matrix to record values as many times as the i^{th} of words-pair (present state) is preceded by the j^{th} term in the defined Arabic text (next state transitions). As shown in Alg. 2 the algorithm of watermark generation is introduced formally and executed.

PROCEDURE **ATA_WM_generation**(PAT)

- Input: PAT, IMM
- Output: FM
- BEGIN
- Prep_Building_MM (PAT)
- pw = first_word(PAT)
- pd2 = PAT – [pw] // begin with 2nd word
- fm = w2_mm
- **for each** w **in** pd2
  - fm[pw][cw] = fm[pw][w] + 1
  - pw = cw
- **return** fm

Algorithm 2: Watermark generation algorithm of FATZWNLP

where, cw: refers to the current-word and pw: refer to the previous-word.

3.1.3 Algorithm of Watermark Embedding

The embedding process of FATZWNLP is performed logically without any improvement in the original Arabic text by finding the summation of nonzeros weight values of the content interrelationship of each state and store it in a watermark database as given in Eq. (2) and illustrated in Fig. 5 below.

$$wv = \sum_{i,\,j=1}^{n-2} wheightvalues[i][j] \tag{2}$$

Present States (ps)	Avalible next states(ns) in the given Arabic text										
	يقفز	الثعلب	البني	السريع	فوق	البطيء	للوصول	إلى	الميت	Interrelation wheight	Wheight value
يقفز الثعلب	0	0	**1**	0	0	0	0	0	0	1/1	1
الثعلب البني	0	0	0	**1**	0	**1**	0	0	**1**	1/3, 1/3, 1/3	1
البني السريع	0	0	0	0	**1**	0	0	0	0	1/1	1
السريع فوق	0	1	0	0	0	0	0	0	0	1/1	1
فوق الثعلب	0	0	**1**	0	0	0	0	0	0	1/1	1
البني البطيء	0	0	0	0	0	0	**1**	0	0	1/1	1
البطيء للوصول	0	0	0	0	0	0	0	**1**	0	1/1	1
للوصول إلى	0	**1**	0	0	0	0	0	0	0	1/1	1
إلى الثعلب	0	0	**1**	0	0	0	0	0	0	1/1	1
Total whieght value											**9**

Figure 5: Logical embedding of the watermark using FATZWNLP

3.2 Algorithms of Watermark-Extraction and Detection

Patterns of attacked watermark (W2-EWM$_A$) must be created before the identification of the attacked documents of Arabic-Text (PAT) and the corresponding rate of patterns and watermark distortion must be determined by FATZWNLP to detect any tampering with the authentication of the provided material.

This method includes two main algorithms, which are watermark extraction and watermark identification. Nevertheless, the detection algorithm will remove W2-EWM$_A$ from the obtained (PAT) and balance W2-WMP$_O$.

For the suggested algorithm of watermark extraction, PAT must be given as the input. In order to extract the watermark pattern for (PAT), this very of algorithm of watermark generation procedure should have been carried out.

3.2.1 Algorithm of Watermark Extraction

PAT should be provided as input to initial setup of this algorithm. Though, W2_WMP$_A$ is a core output of this algorithm as illustrated formally in Alg. 3.

```
PROCEDURE WM_extraction(PAT_A)

- Input: pre-processed text (PAT_A)
- Output: attacked watermark patterns (WMP_A).
- BEGIN
- ATA_WM_generation (PAT_A)
- for ps = 1 to W2_arrList'.Length - 2,
        o   for ns = 1 to W2_arrList'.Length,
        o   if W2_MM'[ps][ns] != 0,
        o   W2_WMP_A &= W2_MM'[ps] [ns],
- return W2_WMP_A
```

Algorithm 3: Watermark extraction algorithm of FATZWNLP

where PAT_P refers to the pre-processed attacked Arabic text, $W2\_WMP_A$: attacked watermark.

3.2.2 Algorithm of Watermark Detection

$W2\_WVM_A$ and $W2\_WVM_O$ should be provided as inputs. However, the status of the given Arabic text is a core output of this algorithm which can be reliable or not. This process can perform in two steps as follows:

– Main matching for $W2\_WVM_O$ and $W2\_WVM_A$ is achieved. If those two WM patterns are similar in appearance, then there will be a warning "Given Arabic text is a reliable". Otherwise, the note will be rendered "Given Arabic text is not reliable", and then it will be going through next phase.

– Secondary matching is performed by matching of each state's transition status in the entire produced pattern of watermarks. This means $W2\_WVM_A$ of each state is contrasted with an analogous transition of $W2\_WVM_O$ as given by Eqs. (3) and (4) below.

$$W2_{WVMT}(i,j) = \left| \frac{W2_{WVMO}[i][j] - \left(W2_{WVMO}[i][j] - W2_{WVM_A[i][j]} \right)}{W2_{WVM_O[i][j]}} \right| \tag{3}$$

where,

– $W2\_WVM_T$: represents matching of weight rate in transition of change, $(0 < W2\_PMR_T <= 1)_T$

– $W2\_WVM_O$: refers to the initial transfer stage of watermark value.

– $W2\_WVM_A$: refers to attacked transfer stage watermark value.

$$W2\_WVM_S(i) = \left| \frac{\sum_{j=1}^{n\ 2} (W2\_WVM_T(i,j))}{Total\ StateCount(i)} \right| for\ all\ i \tag{4}$$

where,

– i: is the cumulative pattern matching rate of the word state.

– $W2\_WVM_S$: represents matching rate at the state of change, $(0 < W2\_WVM_S <= 100)$.

The following step is obtaining the weight of each state stored in the Markov chain matrix as presented in Eq. (5).

$$sWeight\_W2\_Sw = \left| \frac{W2\_WVM_S(i)\ *\ Transitions\ frequency(i)}{total\ number\ of\ transitions} \right| \tag{5}$$

where,

– $W2\_WVM_S$: is the summation of matching weight of i^{th} state for each pair of Arabic words.

W2_WVM of $AATD_P$ and $OATD_P$ are computed by Eq. (6).

$$W2\_WVM = \left| \frac{\sum_{i=1}^{n-2} W2\_WVMS(i)}{m} \right| \tag{6}$$

where m: is summation of non-zeros in W2_MM.

The distortion rate of the watermark reflects the volume of tampering attacks that take place on the attacked contents of Arabic background, denoted by W2_WDR and calculated Eq. (7).

$$W2\_WDR = 1 - \ W2\_WVM * 100 \tag{7}$$

Algorithm of WM detection is implemented as illustrated in Alg. 4.

PROCEDURE **WM_detection** ($W2\_WVM_O$, $W2\_WVM_A$)

- Input: pre-processed text ($W2\_WVM_O$, $W2\_WVM_A$)
- Output: W2_WVM, W2_WDR
- BEGIN
- ATA_WM_generation ($W2\_WVM_O$)
- *WM_extraction* (WVM_A)
  - **IF** $W2\_WVM_A = W2\_WVM_O$
    - Print "Arabic document is authentic and no tampering occurred"
    - W2_WVM = 100
  - Else
    - Print "Arabic document is not authentic and tampering occurred"
  - **for** i = 1 **to** W2_arrList'.Length - 2,
    - **for** j = 1 **to** W2_arrList'.Length
      - IF $W2\_WVM_O[i][j]$!= 0
        - pattern Count +=1
        - $W2\_WVM_T(i,j) = \left|\frac{W2\_WVM_O[i][j] - (W2\_WVM_O[i][j] - W2\_WVM_A[i][j])}{W2\_WVM_O[i][j]}\right|$
        - transWVMTotal += $W2\_WVM_T$
      - Else
        - IF $W2\_WVM_A[i][j]$!= 0
        - patternCount += $W2\_WVM_A[i][j]$
  - $W2\_WVM_S(i) = \left|\frac{\sum_{j=1}^{n-2}(W2\_WVM_T(i,j))}{Total\ StateCount(i)}\right|$
  - $sWeight = \frac{W2\_WVM_S(i) * Transitions\ frequency(i)}{total\ no\ of\ transitions}$
- W2_SW += stateWeight
- $W2\_WVM = \frac{\sum_{i=1}^{n-2}(W2\_SW) * Total\ number\ of\ transitions}{Total\ number\ of\ transitions} * 100$
- $W2\_WDR = 1 - W2\_WVM * 100$

return w2_WVM, w2_WDR

Algorithm 4: Algorithm of WM detection using FATZWNLP

where,

– W2_SW: is value of weight of matched states.

– W2_WDR: refer to the importance of WM distortion rate ($0 < W2\_WDR_S <= 100$).

Fig. 6 shows the effects results of WM extraction and detection.

State	Embedded wm patterns	Extracted wm patterns	Primary matching on state level	Secondary matching on transitions level			**WVM**
			ps	ns1	ns2	ns3	
يقفز الثعلب	1	1.1	-	$1/1$	$(0-(0-1)/1$	-	0.5
الثعلب البني	1.1.1	1.2.1	-	$1/1$	$(1-(1-2)/1$	$1/1$	0.66
البني السريع	1	1	1	-	-	-	1
...	…	…	…	…	…	…	…
إلى الثعلب	1	0.1	-	$0/1$	$1/1$	-	0.5
Total WVM							2.66

Figure 6: Results of WM extraction and detection using FATZWNLP

4 Implementation, Simulation, and Comparison

To validate the accuracy of FATZWNLP, a self-developed program has been implemented, several scenarios of experiments and simulation are performed as explained in detail in the following subsections.

4.1 Implementation and Setup Environment

FATZWNLP approach, is executed by self-developed program in object oriented and PHP using VS Code IDE on the environment having modern features.

4.2 Simulation and Experimental Parameters

The following an experimental, simulation metrics and their related values that used to perform the experiments are given in Tab. 1.

Table 1: Simulation and experimental metrics

Metric	Value
Arabic dataset size	[ASST, 179], [AMST, 421], [AHMST, 559] and [ALST, 2018]
Attack type	Insertion, deletion, and reorder
Attack volumes	5%, 10%, 20% and 50%
Watermark fragility weight	H when close to 100 L when close to 0
W2_WVM	(H if W2_WVM > 70, M if 40 < W2_WVM < 70, and L if W2_WVM < 40)
W2_WDR	(H if W2_WDR > 70, M if 40 < W2_WDR < 70, and L if W2_WDR < 40)

4.3 Watermark Fragility and Tampering Detection Accuracy Metrics

Watermark fragility and tampering detection accuracy of FATZWNLP is evaluated using the following metrics.

– Accuracy of watermark fragility (W2_WVM and W2_WDR) is evaluated under main four attack volumes which are: very low (5%), low (10%), mid (20%) and high (50%).

– Desired accuracy of watermark fragility values near to 100%.

– Comparison of text size, attack types, and attack volumes effects against accuracy of watermark fragility using the proposed FATZWNLP, MACATDW and UZWAMW baseline approaches.

4.4 Baseline Approaches

Detection accuracy of FATZWNLP are compared to MACATDW and UZWAMW approaches. A comparison is performed by using performance and accuracy parameters. Baseline approaches and their objectives are stated in Tab. 2.

Table 2: The baseline approaches

Approach	Attacks types	Attacks volumes	Dataset size	Objective
MACATDW UZWAMW	Insertion, deletion and reorder	5%, 10%, 20% and 50%	Small, medium, and large	Content authentication and tampering detection.

4.5 Simulation and Experiment Discussion with FATZWNLP

In this subsection, various scenarios of simulation and experiments of FATZWNLP have been performed to evaluate its performance in terms of watermark fragility. The character set covers all Arabic characters, spaces, special symbols, and numbers. Simulations are performed on various datasets sizes and various kind of attacks and volumes as showed above in Tab. 1.

4.5.1 Accuracy Evaluation of Tampering Detection

Various simulation scenarios have been conducted to text and evaluate the watermark fragility accuracy of FATZWNLP using all types of attacks and their rates as show in Tab. 3. Results are illustrated in Fig. 7.

Table 3: Watermark fragility accuracy evaluation of FATZWNLP

Attack volume	Attacks		
	Insertion	Deletion	Reorder
5%	93.32	88.14	77.49
10%	88.03	82.50	64.10
20%	80.29	66.30	44.51
50%	62.12	38.73	25.49

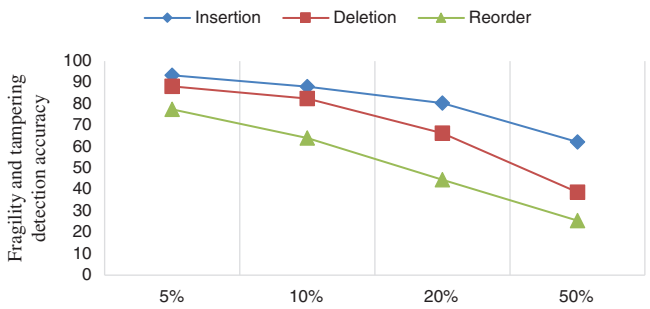

Figure 7: Watermark fragility accuracy evaluation of FATZWNLP

From Tab. 3 above and Fig. 7 below, it shows that FATZWNLP gives accurate results of WM fragility accuracy. In case of reorder attack, results show high effect of watermark fragility because insertion and deletion attacks are occurred automatically when reorder attack occurred.

4.6 Comparative Results and Discussion

This subsection presents a performance comparison in terms of watermark fragility accuracy of FATZWNLP with MACATDW and UZWAMW approaches, and study their effect under main comparison metrics, which are dataset size, attack types, and volumes.

4.6.1 Results Analysis of Dataset Size Metric

A comparison of dataset size effect on watermark fragility accuracy and a performance evaluation between FATZWNLP, MACATDW, and UZWAMW approaches have been presented in this subsection and graphically illustrated in Fig. 8.

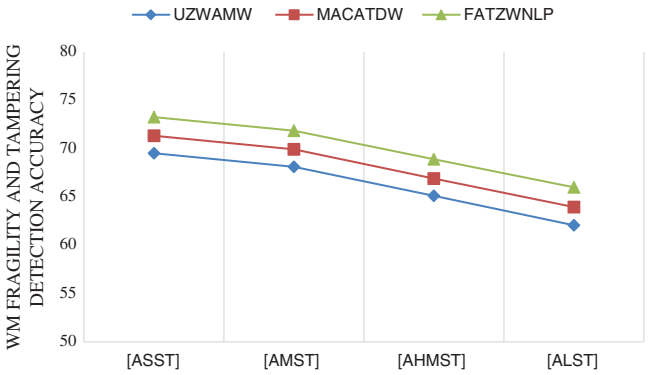

Figure 8: A compression of dataset size effect on the performance of FATZWNLP MACATDW and UZWAMW approaches

As resulted in the summary of the comparative results shown by Fig. 8, in the case of FATZWNLP approach, the highest effects of dataset size that lead to the best accuracy of watermark fragility are ordered as ASST, AMST, AHMST and ALST, respectively. This means that tampering detection accuracy increases with decreasing Arabic document size and decreases with increasing document size. Furthermore, results show that FATZWNLP approach outperforms both MACATDW and UZWAMW approaches in terms of watermark fragility detection under all scenarios of dataset sizes.

4.6.2 Results Analysis of Attack Type Metric

Fig. 9 illustrates a comparison of the different attack types affecting tampering detection accuracy against all dataset sizes and all scenarios of attack volumes. The comparison was conducted using FATZWNLP with MACATDW and UZWAMW approaches.

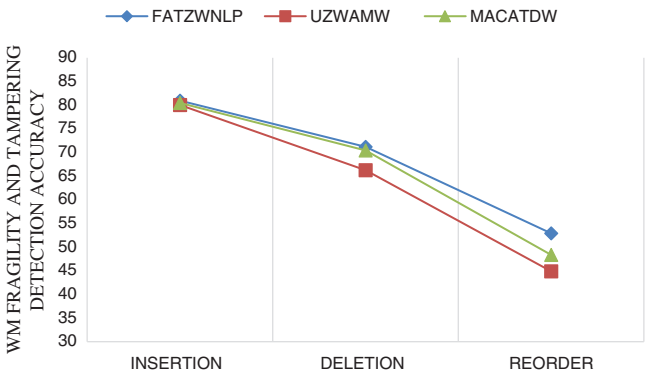

Figure 9: A compression of attack type effect on performance

As shown by Fig. 9, FATZWNLP outperforms MACATDW and UZWAMW in terms of watermark fragility detection in all scenarios of attack types. This means that FATZWNLP approach is strongly recommended for tampering detection of Arabic text exchanged via the Internet.

4.6.3 Results Analysis of Attack Volume Metric

Fig. 10 illustrates a comparison of the different attack volumes affecting watermark fragility accuracy against all scenarios of attack types and all dataset sizes. The comparison was performed using RDZWANLP with MACATDW and UZWAMW approaches.

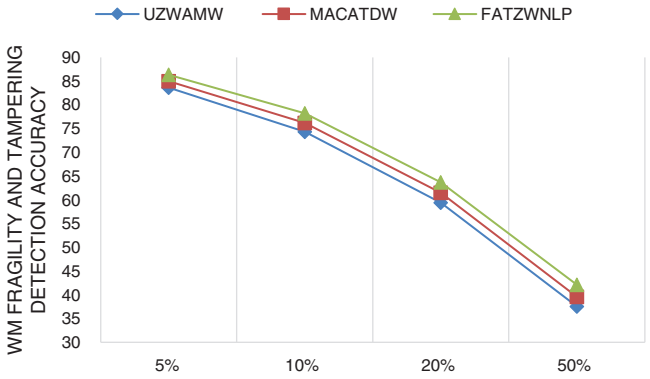

Figure 10: A compression of attack volume effect on performance

As shown by Fig. 10, FATZWNLP outperforms MACATDW and UZWAMW in terms of watermark fragility accuracy in all scenarios of low, mid, and high volumes of all attack types. This means that FATZWNLP approach is highly recommended and applicable for tampering detection of text exchanged via the Internet under all volumes of all attacks.

5 Conclusions

Based on the second tire and word technique of HMM, a fragile digital zero-watermark approach abbreviated as FATZWNLP has been developed in this paper for tampering detection of Arabic text exchanged via the Internet. In the paper, zero-watermarking has been integrated with HMM, which is used as NLP technique for text analysis and finding interrelationships weight between given Arabic contents to extract a watermark data. FATZWNLP approach has been implemented using PHP self-developed program in VS code IDE, likewise, simulation and experiments were conducted using different Arabic datasets under various attack types and volumes. Simulation and experiment results show that RDZWANLP achieves a high accuracy of tampering detection under all scenarios of attack types with high accuracy of watermark fragility. For the future work, authors will intend to improve the performance using other techniques of Markov model.

Funding Statement: The authors express their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through Research Groups under grant number (R. G. P. 2/55/40 /2019)

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Nurul, K. Amirrudin, Y. Lip and R. Hameedur, "A review of text watermarking: Theory, methods, and applications," *IEEE Access*, vol. 6, pp. 8011–8028, 2018.

[2] M. Mohamed, "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] D. Tong, C. Zhu, N. Ren and W. Shi, "High-capacity and robust watermarking scheme for small scale vector data," *KSII Transactions on Internet and Information Systems*, vol. 13, no. 12, pp. 6190–6213, 2019.

[4] S. Giovanni, F. Bertini and D. Montesi, "Fine-grain watermarking for intellectual property protection," *EURASIP Journal on Information Security*, vol. 10, pp. 1–20, 2019

[5] A. Alwan, M. Shahidan, N. Nur, S. Mohammed and S. Mohd, "A review and open issues of diverse text watermarking techniques in spatial domain," *Journal of Theoretical and Applied Information Technology*, vol. 96, pp. 5819–5840, 2018.

[6] P. Selvama, S. Balachandran, S. Pitchai and R. Jayabal, "Hybrid transform based reversible watermarking technique for medical images in telemedicine applications," *Optik*, vol. 145, pp. 655–671, 2017.

[7] N. N. Hurrah, S. A. Parah, N. A. Loan, J. A. Sheikh, M. Elhoseny *et al.,* "Dual watermarking framework for privacy protection and content authentication of multimedia," *Future Generation Computer Systems*, vol. 94, pp. 654–673, 2019.

[8] A. Soltani, S. Ron, T. Sellis and E. Bertino, "On the properties of non-media digital watermarking: A review of state-of-the-art techniques," *IEEE Access*, vol. 4, pp. 2670–2704, 2016.

[9] C. Qin, C. Chang and T. Hsu, "Effective 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, "A ioint stego-watermark approach for early tamper detection," *Springer International Publishing Switzerland*, vol. 660, pp. 427–452, 2017.

[11] S. Hakak, A. Kamsin, O. Tayan, M. Yamani and G. Amin, "Approaches for preserving content integrity of sensitive online Arabic content: A survey and research challenges," *Information Processing and Management*, vol. 56, no. 2, pp. 367–380, 2017.

[12] T. Milad, "ANiTH: A novel intelligent text hiding technique," *IEEE Dataport*, vol. 10, 2018.

[13] A. Shabir, A. Javaid, A. Jahangir and A. Nazir, "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, pp. 935–949, 2020.

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

[15] H. Khizar, K. Abid, A. Mansoor and G. Alavalapati, "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] A. Reem and A. Lamiaa, "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, 2018.

[17] S. Mujtaba and S. Asadullah, "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] M. Yasser, N. Muhammad and T. Omar, "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. Anes, R. Farida and A. Sakinah, "Text steganography using extensions kashida-based on the moon and sunletters concept," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 8, pp. 286–290, 2017.

[20] M. Abdul, S. Wesam and A. Dhamyaa, "Text steganography based on Unicode of characters in multilingual," *International Journal of Engineering Research and Applications*, vol. 3, no. 4, pp. 1153–1165, 2013.

[21] T. Omar, M. Yasser and N. Muhammed, "An adaptive zero-watermarking approach for text documents protection," *International Journal of Image Processing Techniques*, vol. 1, no. 1, pp. 33–36, 2014.

[22] M. Ghilan, F. Ba-Alwi and F. Al-Wesabi, "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, 2014.

[23] M. Hanaa and A. Maisa'a, "Comparison of eight proposed security methods using linguistic steganography text," *International Journal of Computing and Information Sciences*, vol. 12, no. 2, pp. 243–251, 2016.

[24] A. Zulfiqar, I. Muhammad, A. Mansour, S. Muhammad and U. Sana, "Chaos-based robust method of zero-watermarking for medical signals," *Future Generation Computer Systems*, vol. 88, pp. 400–412, 2018.

[25] A. Zulfiqar, I. Muhammad, A. Mansour, Z. Tanveer and S. Muhammad, "A zero-watermarking algorithm for privacy protection in biomedical signals," *Future Generation Computer Systems*, vol. 82, pp. 290–303, 2018.

[26] P. Jayashree and P. Theagarajan, "Semi fragile watermarking for content-based image authentication and recovery in the DWT-DCT domains," *International Arab Journal of Information Technology*, vol. 15, no. 6, pp. 1076–1081, 2018.

[27] N. Addin, A. Wan, R. Abdul Rahman, S. Khairulmizam and M. Sharifah, "Robust digital text watermarking algorithm based on Unicode extended characters," *Indian Journal of Science and Technology*, vol. 9, no. 48, pp. 1–14, 2016.

[28] T. Hung-Hsu, L. Yen-Shou and L. Shih-Che, "A zero-watermark scheme with geometrical invariants using SVM and PSO against geometrical attacks for image protection," *Journal of Systems and Software*, vol. 86, no. 2, pp. 335–348, 2013.

[29] T. Omar, N. Muhammad and M. Yasser, "A hybrid digital-signature and zero-watermarking approach for authentication and protection of sensitive electronic documents," *Scientific World Journal*, vol. 2014, pp. 1–14, 2014.

[30] K. Manpreet and K. Vinod, "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, 2016.