Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2020.013782
Article

Proposing a High-Robust Approach for Detecting the Tampering Attacks on English Text Transmitted via Internet

Fahd N. Al-Wesabi[1,*], Huda G. Iskandar[2], Mohammad Alamgeer[3] and Mokhtar M. Ghilan[2]

[1]Department of Computer Science, King Khalid University, KSA & Faculty of Computer and IT, Sana'a University, Sana'a, Yemen
[2]Faculty of Computer and IT, Sana'a University, Sana'a, Yemen
[3]Department of Information Systems, King Khalid University, Mayahel Aseer, Saudi Arabia
*Corresponding Author: Fahd N. Al-Wesabi. Email: Falwesabi@kku.edu.sa
Received: 21 August 2020; Accepted: 14 September 2020

Abstract: In this paper, a robust approach INLPETWA (an Intelligent Natural Language Processing and English Text Watermarking Approach) is proposed to tampering detection of English text by integrating zero text watermarking and hidden Markov model as a soft computing and natural language processing techniques. In the INLPETWA approach, embedding and detecting the watermark key logically conducted without altering the plain text. Second-gram and word mechanism of hidden Markov model is used as a natural text analysis technique to extracts English text features and use them as a watermark key and embed them logically and validates them during detection process to detect any tampering. INLPETWA approach has been implemented by self-developed program using PHP with VS code IDE. INLPETWA approach has been proved with various experiments and simulation scenarios. Comparison results with baseline approaches also show that the proposed approach is appropriate to detect all types of tampering attacks. The paper includes implications for integrating natural language processing and text-watermarking to propose an intelligent solution. This paper fulfils an identified need to study how we can use a robust text information via various Internet applications.

Keywords: NLP; Markov model; text-watermarking; text feature; content authentication; tampering detection

1 Introduction

For the research community, the security and reliability of text information exchanged through the Internet is the greatest promising and challenging field. In communication technologies, content authentication and honesty of automated text verification in different Languages and formats are of great significance. Numerous applications such as electronic banking, electronic commerce etc. impose most challenges during contents transfer via internet. In terms of content, structure, grammar, and semantics, much of the multimedia exchanged via Internet is in textual form and 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 can be used to hide the various data, such as text, binary images, video, and audio and embed them in digital content as a watermark information [2,3].

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

Several conventional methods and solutions for text watermarking were proposed [5] and categorized into different classes, such as linguistic, structure, zero watermark and format based methods [6]. To insert the watermark information into the document, most of these solutions require certain modifications or improvements 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. Moreover, in this technique the contents of given digital context can be utilized to generate the watermark key [1,6–8].

Restricted research has centred on the appropriate solutions to verify the credibility of critical digital media online [9–11]. In the research community, digital text authentication and tampering detection have received great attention. In addition, 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 techniques and solutions 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 digital Holy Qur'an in Arabic, eChecks, online marks and exams. Different Arabic alphabet characteristics such as diacritics, extended letters, and other Arabic symbols make it easy to alter the key meaning of text material by making basic changes such as modifying diacritic arrangements [11,15]. The most popular soft computing and natural language processing (NLP) technique that is used for text analysis is HMM.

In this paper, authors present a robust approach INLPETWA (an Intelligent NLP and English Text Watermarking Approach) which makes use of English text zero watermarking and second gram of word method of Markov model. Soft computing tool and zero watermarking technique have been integrated in INLPETWA approach in order to analyzing the given English text and extract the watermark information. Embedding process will be conducted logically in the plain English text without effecting on contents and size of the plain text. After the transmission of the text, aim of the hidden DWM is used in next phase to detect and obtain tampered text on received English text and ensures the authenticity of the transmitted text.

The core objective of the INLPETWA approach is to achieve better performance with high detection level of any illegal tampering occurred in English text exchanged electronically via Internet.

This paper is organized in addition to the Section 1 as follows. Section 2 presents the previous related works. Section 3 presents the proposed INLPETWA. Section 4 explain the implementation, simulation, and

experimental details. Section 5 describes the comparison and results discussion, and Section 6 offers conclusions.

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-Based Methods

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

To enhance the capability and imperceptibility of Arabic text, a text watermarking algorithm based on open-word spaces [16] have been suggested. In this method, every word-space is used to embed binary data 1 or 0 to obtain the physical altering occurred on plain text.

A technique of text steganography [17] has been proposed to conceal details in the Arabic language. The process of this algorithm considers the presence in Arabic of Harakat (diacritics, i.e., Fat-ha, Kasra 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 a bit 1 and omitted for a bit 0.

The method of text steganography [19] has been proposed to use Kashida extensions based on the characters ‘moon’ and ‘sun’ 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 cases are used 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 should be hidden in a time frame. Used ASCII code for embedding 00. However, for embedding 01,10, and 11, Unicode used multilingual ones.

2.2 Structure-Based Methods

Structural text-watermark-based methods are based on a framework dependent on material in which altering on structure of the original text are performed to hide a watermark data [21–24].

Text watermarking method based on Unicode extended characters has been proposed in [21] to avoid the textual contents from illegal attacks. This method covers three main phases. Embedding the watermark key in this method based on building the predefined tables of encoding letters and using the scrambling algorithm to protect the watermark key.

The replacement attack method [22], which is focused on preserving the position of words in the text document, has been proposed. This approach depends on manipulating word transitions in the text document. For authentication the Chinese text, text based watermarking approaches [23,24] have been suggested based on combining the sentences properties. The mechanism of these approaches is as follows: first, a Chinese text is split into sentence sets, and then, for each word, a semantic code is obtained. Sentence entropy is determined by the frequency of semantic codes. Sentence significance

is determined by the finding the similarity of semantic of words in Tongyici Cilin via the tree structure of the words. Finally, by using sentence entropy, importance, length, and a weighting function of each sentence.

2.3 Zero-Watermark-Based Methods

Zero watermark-based methods rely on text characteristics. Several zero-text-watermark-based algorithms and techniques have been suggested, as in studies [12,15,25–29].

To measure the reliability of the electronic texts posted on social application ns, the ANiTH method [12] has been presented. Inside a digital text, this algorithm hides an invisible watermark and can be detected later to verify the reliability of the text content. A system based on zero watermarks [15] presented to validate the integrity internet data in which watermarks are embedded in plain text prior to transmission. The created watermark key is based on certain content characteristics, such as data size, frequency of data appearance, and data capture time.

Zero-watermarking algorithm has been presented in [25] to improve the privacy of a person by using Hurst exponent and zero crossing of the frames. For watermark embedding, these two steps are determined to evaluate the unvoiced frames. The process of this approach depends on integrating an individual's identity without notifying any distortion in the signals of medical expression.

A zero-watermarking approach [26] was proposed to resolve the issues related with the security of English context, such as content verification and copyright protection. A zero-watermarking approach [27,28] has been suggested, which is based on the Markov model for English text content authentication. In this method, to extract the safe watermark information, the probability features of the English text are used and stored to verify the validity of the attacked text. These methods provide security against popular text attacks with a watermark distortion rate if, for all known attacks, it is greater than one. For copyright protection of English text based on the presence frequency of non-vowel ASCII letters and terms, the conventional watermark method [29] has been suggested.

3 Proposed Approach

An intelligent approach is proposed in this paper by integrating text-watermark and hidden Markov model as NLP technique in which do not need additional details to be embedded as a watermark data and do not need to make any changes to the plain text to insert a watermark inside it. Second gram of word method of Markov model is used as NLP to analyze English content and extract the features of these text contents. Several assumptions of INLPETWA are addressed as follows:

- Watermark key will be extracted as a result of English text analysis without altering the original text.
- High watermark robustness in all cases whenever the tackers get watermark key in any way.
- All types of tampering attacks will be addressed to detect randomly such as insertion, deletion, and reorder attacks.
- All volumes of tampering will be addressed to detect whenever attack volume is very low.
- There are no limitations in size of English text.

The following subsections explain in detail two main processes that should perform in INLPETWA. The first process called watermark generation and embedding process, however, the second one called watermark extraction and detection process.

3.1 Watermark Generation and Embedding Processes

Three algorithms should be performed in this process are pre-processing, English text analysis and WM generation, and watermark embedding algorithm as illustrated in Fig. 1.

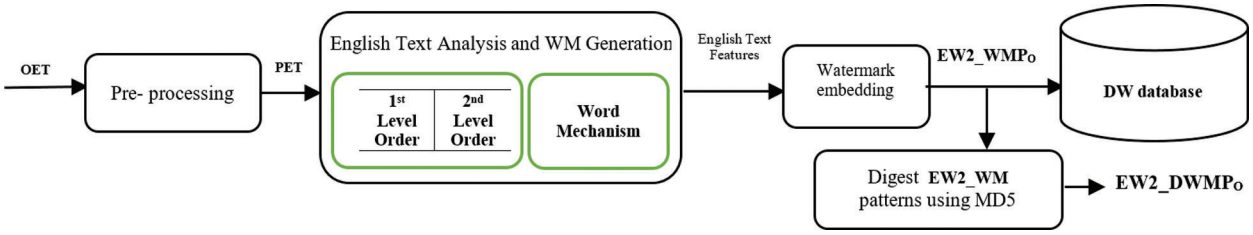

Figure 1: Text zero-watermarking generation and embedding processes of INLPETWA

3.1.1 Pre-processing Algorithm

Preprocessing of the plain English text is a core activity in both the WM generation and extraction phases to set all English letter in small case, delete blank spaces and extra new lines, and it will affect the accuracy of tampering detection and watermark robustness. The original English text (OET) is a necessary provided as input for this process.

3.1.2 Text Analysis and WM Generation Algorithm

This algorithm contains two sub procedures—building Markov chain matrix and text analysis, and WM generation processes.

–*Building Markov matrix* is the core phase to run INLPETWA approach. Markov chain matrix should be constructed in this phase to configure the Markov model environment and represents all possible states and transitions without reputations. In INLPETWA approach, each pair of English words of a given text refers to current state, and every word represent a transition in Markov chain matrix. During constructing the Markov chain matrix, zero values will be initialized for all states and transitions positions. Those positions will be used later to keep the status of presence times that the i^{th} pair of words is followed by the j^{th} unique word within the given English text as presented below in Alg. 1.

Algorithm 1: Barkov Algorithm of INLPETWA

```
PROCEDURE PBMM (OET)

1.  Input: original English text (OET)
2.  Output: Markov matrix with initial zero values
3.  BEGIN
4.  // perform pre-processing process
5.  for each word in OET
6.        // remove new lines and spaces letters
7.        PET ← trim ("space" or "newLine")
8.        // convert letter case from capital to small letters
9.        PET ← LowerCharacter(word)
10. // Build list of non values text words
11. EW2_MM = { }
12. for each pair-of-words in PET
13.     if pair-of-words not in w2list
14.         EW2_MM ← EW2_MM U {pair-of-words}
15.     for ps = 1 to EW2_MM.length – 2
16.         for ns = 1 to EW2_MM.length
17.             EW2_MM[ps][ns] = 0
18. return EW2_MM
```

where,

–OET: is plain English text, PET: is a pre-processed English text.

This algorithm is performed as second step of this process in which English text should be analyzed to extract the features of the given text and utilize them to generate watermark information. In this algorithm, occurrence time of all transitions for each present state of pair words will computed by Eq. (1).

$$EW2_MM[ps][ns] = \sum_{i,j=1}^{n-2} transitions\ (i,j) \tag{1}$$

The following example of the provided English text illustrates the work mechanism of this algorithm.

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

In second gram of word method of Markov model, each pair unique of English words represent a unique state. Fig. 2 explain the representation of states and transitions available of the above sample of English text.

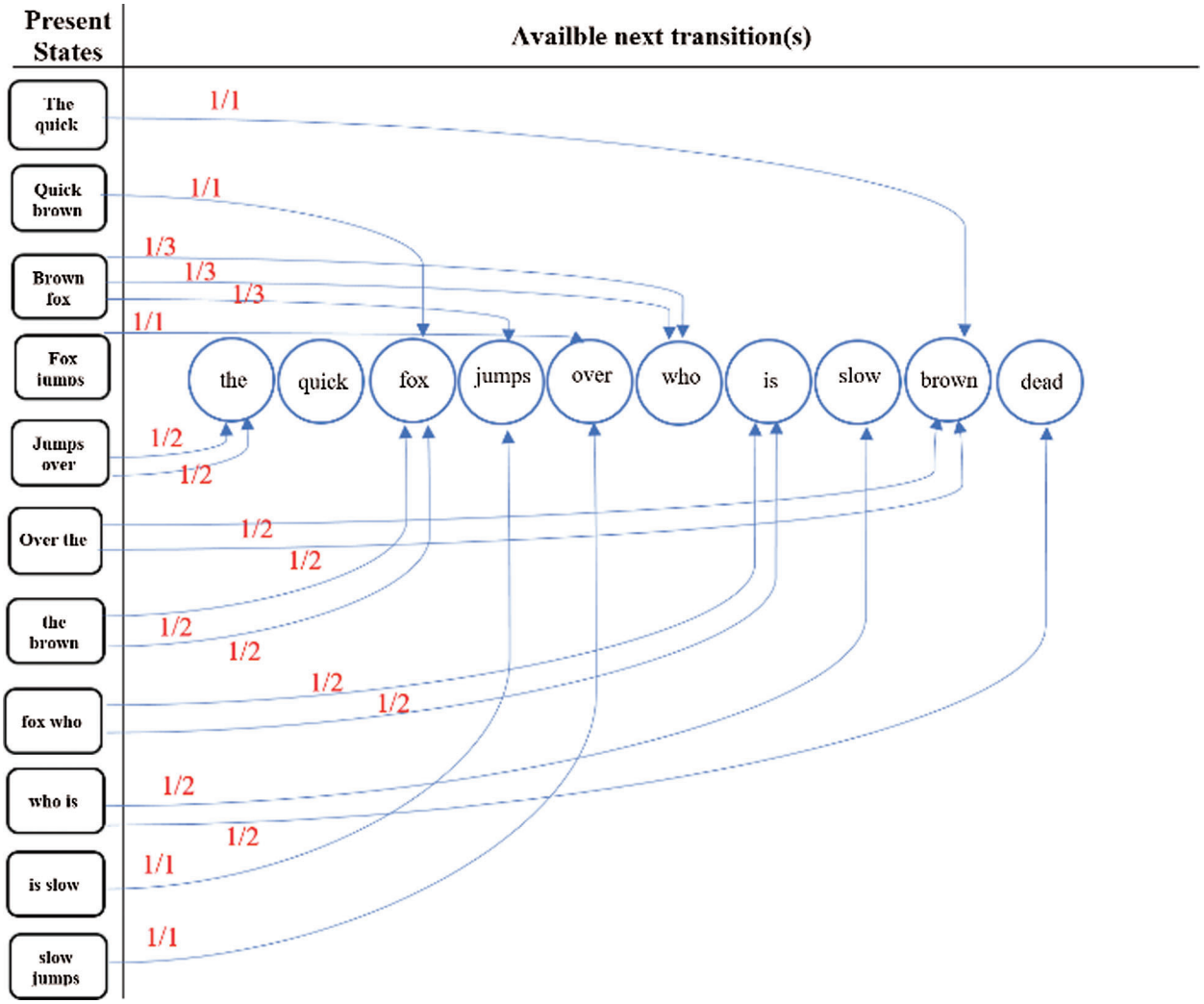

Figure 2: States and transitions representation of the given text sample using INLPETWA

Authors assume “brown fox” is a current state, and its transition(s) are “jumps”, “who”, and “who”. We observe that “who” transition appears twice in the given English text sample.

Based on second gram of word method of hidden Markov, algorithm of text analysis and WM generation performed as presented in Fig. 3.

States	Transitions										DWM Patterns
	brown	dead.	fox	is	jumps	over	quick	slow	the	who	
the quick	1	0	0	0	0	0	0	0	0	0	1,0,0,0,0,0,0,0,0,0
quick brown	0	0	1	0	0	0	0	0	0	0	0,0,1,0,0,0,0,0,0,0
brown fox	0	0	0	0	1	0	0	0	0	2	0,0,0,0,1,0,0,0,0.2
fox jumps	0	0	0	0	0	1	0	0	0	0	0,0,0,0,0,1,0,0,0,0
jumps over	0	0	0	0	0	0	0	0	2	0	0,0,0,0,0,0,0,0,2,0
over the	2	0	0	0	0	0	0	0	0	0	2,0,0,0,0,0,0,0,0,0
the brown	0	0	2	0	0	0	0	0	0	0	0,0,2,0,0,0,0,0,0,0
fox who	0	0	0	2	0	0	0	0	0	0	0,0,0,2,0,0,0,0,0,0
who is	0	1	0	0	0	0	0	1	0	0	0,1,0,0,0,0,0,1,0,0
is slow	0	0	0	0	1	0	0	0	0	0	0,0,0,0,1,0,0,0,0,0
slow jumps	0	0	0	0	0	1	0	0	0	0	0,0,0,0,0,1,0,0,0,0

Figure 3: Feature extraction and WM generation of given English text using INLPETWA

Feature extraction of English text and WM generation algorithm is proceeds formally as presented in Alg. 2.

Algorithm 2: Watermark generation algorithm of INLPETWA

```
PROCEDURE ETA_WMG (PAT)

1.  Input: PET, IMM
2.  Output: FM
3.  BEGIN
4.  PBMM (PET)
5.  ppw = first-pair-of-words(PET)
6.  pd2 = PET – [ppw] // begin with 2nd pair of words
7.  fm = EW2_MM
8.  for each word in pd2
9.      fm[ppw][cpw] = fm[ppw][word] + 1
10.     ppw = cpw
11. return fm
```

where, pw: previous pair of words, cpw: current pair of words.

3.1.3 Watermark Embedding Algorithm

In this approach, watermark embedding process will be done logically without necessity to make altering on the original plain text. As a result of feature extraction of the given English text, WM data is embedded logically by obtaining non-zeros values in Markov chain matrix. Those values will be concatenated and used extract the WM key pattern $EW2\_WMP_O$, as given in in Eq. (2) and showed in Fig. 4.

1 – 1 – 1.2 – 1 – 2 – 2 – 2- 2 – 1.1 – 1 - 1

Figure 4: Original WM EW2_WMP$_O$ using INLPETWA

$$EW2\_WMP_O \;\&= EW2\_mm[ps]\,[ns], \tag{2}$$

Algorithm of watermark embedding process using INLPETWA approach is executed as showed below in Alg. 3.

Algorithm 3: Watermark embedding algorithm of INLPETWA

```
PROCEDURE WM_embedding (PET)

1.  Input: pre-processed text (PET)
2.  Output: original watermark patterns
3.  BEGIN
4.  ETA_WMG (PET)
5.  for ps = 1 to EW2_arrList.Length - 2,
6.      for ns = 1 to EW2_arrList.Length,
7.          if EW2_MM [ps][ns] != 0
8.              EW2_WMPO &= EW_MM [ps] [ns]
9.  EW2_DWMPO = MD5(EW2_WMPO)
10. return EW2_DWMPO, wEW2_WMPO
```

where, EW2_WMP$_O$: WM patterns, EW2_DWMP$_O$: digested WM.

3.2 Watermark Extraction and Detection Process

Pre-processing process is required for attacked English text (PET$_A$). Then, attacked watermark key (EW2_EWM$_A$) should be produced, and detection process should be calculated by INLPETWA approach to detect any illegal tampering occurred in the given English text.

This process includes two core algorithms are watermark extraction and detection. Though, EW2_EWM$_A$ will be produced from (PET$_A$) and compared with EW2_WMP$_O$ by detection process.

Fig. 5 illustrate the work mechanism of this process.

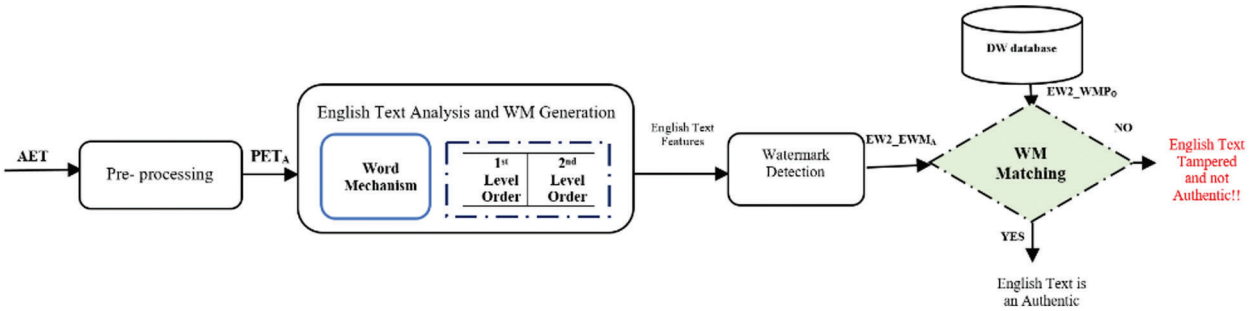

Figure 5: Text zero-watermark extraction and detection processes of INLPETWA

3.2.1 Watermark Extraction Algorithm

PET_A should be provided as input to initial setup of this algorithm. Though, $EW2\_WMP_A$ is a core output of this algorithm as illustrated formally in Alg. 4.

Algorithm 4: Algorithm of water mark extraction using INLPETWA

PROCEDURE WME (PET_A)

```
1.  Input: pre-processed text (PETA)
2.  Output: attacked watermark patterns (EW2_WMPA).
3.  BEGIN
4.  ETA_WMG(PETA)
5.  for ps = 1 to EW2_arrList'.Length - 2,
6.      for ns = 1 to EW2_arrList'.Length,
7.          if EW2_MM'[ps][ns] != 0,
8.              EW2_WMPA &= EW2_MM'[ps] [ns],
9.  return EW2_WMPA
```

where, PET_A: pre-processed attacked English text document, $EW2\_WMP_A$: attacked DWM.

3.2.2 Algorithm of Watermark Detection

$EW2\_WMP_A$ and $EW2\_WMP_O$ should be provided as inputs to run this algorithm. However, the status of the given English 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 $EW2\_WMP_O$ and $EW2\_WMP_A$ is achieved. If those two WM patterns are similar in appearance, then there will be a warning "Given English text is a reliable". Otherwise, the note will be rendered "Given English text is not reliable", and then it going through next phase.
- Secondary matching is performed by matching each state's transition status in the entire produced pattern of watermarks. This means $EW2\_WMP_A$ of each state is contrasted with an analogous transition of $EW2\_WMP_O$ as given by Eq. (3) and (4) below.

$$EW2\_PMR_T(i,j) = \left| \frac{EW2\_WMP_O[i][j] - \left(EW2\_WMP_O[i][j] - EW2\_WMP_A[i][j] \right)}{EW2\_WMP_O[i][j]} \right| \tag{3}$$

where,

- $EW2\_PMR_T$: represents matching rate at the transition of change, $(0 < EW2\_PMR_T <= 1)_T$
- $EW2\_WMP_O$: refers to the initial transfer stage watermark value.
- $EW2\_WMP_A$: refers to attacked transfer stage watermark value.

$$EW2\_PMR_S(i) = \left| \frac{\sum_{j=1}^{n-2} (EW2\_PMR_T(i,j))}{Total\ State\ Pattern\ Count(i)} \right| for\ all\ i \tag{4}$$

where,

- n: is a summation value of non zeros transitions.
- i: is the cumulative pattern matching rate of the word state.
- $EW2\_PMR_S$: represents matching rate at the state of change, (0 < EW2_PMRS <=100).

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

$$EW2_Sw = \left| \frac{EW2\_PMR_S(i) \ * \ Transitions\ frequency(i)}{total\ number\ of\ transitions} \right| \tag{5}$$

where,

- $EW2\_PMR_S$: is the matching value of ith state for each pair of words.

The final EW2_PMR of PETA and OETP are computed by Eq. (6).

$$EW2\_PMR = \left| \frac{\sum_{i=1}^{n-2} EW2\_PMRS(i)}{N} \right| \tag{6}$$

where, N: is summation of non-zeros in EW2_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 EW2_WDR and computed by Eq. (7).

$$EW2\_WDR = 1 - \ EW2\_PMR * 100 \tag{7}$$

Algorithm of WM detection process is implemented as showed in Alg. 5.

Algorithm 5: Algorithm of water mark detection using INLPETWA

PROCEDURE WMD ($EW2\_WMP_O$, $EW2\_WMP_A$)

1. Input: pre-processed text ($EW2\_WMP_O$, $EW2\_WMP_A$)
2. Output: EW2_PMR, EW2_WDR
3. BEGIN
4. ETA_WMG ($EW2\_WMP_O$)
5. *WME* (WMP_A)
6. **IF** $EW2\_WMP_A = EW2\_WMP_O$
7. Print "English document is authentic and no tampering occurred"
8. EW2_PMR = 100
9. **Else**
10. Print "English document is not authentic and tampering occurred"
11. **for** i = 1 **to** EW2_arrList'.Length - 2,
12. **for** j = 1 **to** EW2_arrList'.Length
13. **IF** $EW2\_WMP_O[i][j]$!= 0
14. pattern Count +=1
15. $EW2\_PMR_T(i,j) = \left| \frac{EW2\_WMP_O[i][j] - (EW2\_WMP_O[i][j] - EW2\_WMP_A[i][j])}{EW2\_WMP_O[i][j]} \right|$
16. transPMRTotal += $EW2\_PMR_T$
17. **Else IF** $EW2\_WMP_A[i][j]$!= 0
18. patternCount += $w2\_WMP_A[i][j]$
19. $EW2\_PMR_S(i) = \left| \frac{\sum_{j=1}^{n-2}(EW2\_PMR_T(i,j))}{Total\ StatePatternCount(i)} \right|$
20. sWeight = $\frac{EW2\_PMR_S(i) * Transitions\ frequency(i)}{total\ no\ of\ transitions}$
21. EW2_SW += stateWeight
22. EW2_PMR = $\frac{\sum_{i=1}^{n-2}(EW2\_SW) * Total\ number\ of\ transitions}{Total\ number\ of\ transitions}$ * 100
23. EW 2_WDR = 1 – EW2_PMR * 100
24. **return** EW2_PMR, EW2_WDR

where,

- EW2_SW: is value of properly weight of matched states.
- EW2_WDR: refer to the importance of WM distortion rate ($0 < EW2\_WDR_S <= 100$).

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

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	1	1	1	1	-	-	1
quick brown	1	1	1	-	0	-	0
brown fox	1.2	2	1.2	-	0	1	0.5
fox jumps	1	-	-	-	0	-	0
jumps over	2	1	2	-	0.5	-	0.5
over the	2	1	2	-	0.5	-	0.5
the brown	2	2	2	1	-	-	1
fox who	2	1.1	1.1	-	0.5	-	0.25
who is	1.1	1	1.1	-	1	0	0.5
is slow	1	-	-	-	0	-	0
slow jumps	1	1	1	-	0		0
brown jumps	-	-	1	-	-	-	0
who are	-	-	1	-	-	-	0
are very	-	-	1	-	-	-	0
very slow	-	-	1	-	-	-	0
jumps more	-	-	1	-	-	-	0
more the	-	-	1	-	-	-	0
PMR =							4.25 / 17 = 0.25

Figure 6: WM extraction and detection process of the given sample text

4 Implementation, Simulation and Experimental

To validate the accuracy of INLPETWA, 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 Environment and Setup

INLPETWA 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 Metrics

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

4.3 Performance Parameters

The performance of INLPETWA refers to accuracy of robustness and tampering detection which is evaluated by using the following parameters.

- Accuracy of tampering detection (EW2_PMR and EW2_WDR) is evaluated under main four attack volumes which are: very low (5%), low (10%), mid (20%) and high (50%).
- Desired accuracy of tampering detection values near to 100%.
- Comparison of text size, attack types, and attack volumes effects against detection accuracy using the proposed INLPETWA approach, ZWAFWMMM and HNLPZWA baseline approaches.

Table 1: Simulation and experimental metrics

Metric	Value
English dataset size	[ESST, 179], [EMST, 421], [EHMST, 559] and [ELST, 2018]
Attack type	Insertion, deletion and reorder
Attack volumes	5%, 10%, 20% and 50%
Robustness and tampering detection accuracy	H when close to 100
	L when close to 0
EW2_PMR	(H if EW2_PMR > 70,
	M if 40 < EW2_PMR < 70, and
	L if EW2_PMR < 40)
EW2_WDR	(H if EW2_WDR > 70,
	M if 40 < EW2_WDR < 70, and
	L if EW2_WDR < 40)

4.4 Baseline Approaches

The performance and accuracy of INLPETWA is compared with HNLPZWA (an intelligent hybrid of natural language processing and zero-watermarking approach) and ZWAFWMMM (Zero-Watermarking Approach based on Fourth level order of Word Mechanism of Markov Model) [30]. Comparison is performed by using performance and accuracy parameters. Baseline approaches and their working parameters are stated in Tab. 2.

Table 2: Compared baseline approach

Approach	Attacks types	Attacks volumes	Dataset size
ZWAFWMMM HNLPZWA	Insertion, deletion and reorder	5%, 10%, 20% and 50%	Small, medium, and large

4.5 Simulation and Experiment Results of INLPETWA

In this sub section, performance evaluation of INLPETWA have been performed. The character set covers all English 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 tampering detection accuracy of INLPETWA using all types of attacks and their rates as show in Tab. 3. Results are illustrated in Fig. 8.

The results in Fig. 7 show that, high effect is noticed under reorder attack in all simulation scenarios. This result is logic because reorder attack makes changes as insertion and deletion attacks. However, simulation using both insertion and deletion attacks, high effect is noticed under deletion attack in all scenarios of attack volumes because deletion attack also makes changes as insertion and deletion attacks. This represent that INLPETWA gives best detection accuracy under all attack's scenarios with their volumes even attack volumes are very low.

Table 3: Accuracy evaluation of INLPETWA

Attack volume (%)	Attacks		
	Insertion	Deletion	Reorder
5	94.47	92.14	84.76
10	90.13	85.37	74.21
20	82.07	74.06	58.85
50	65.30	42.56	34.34

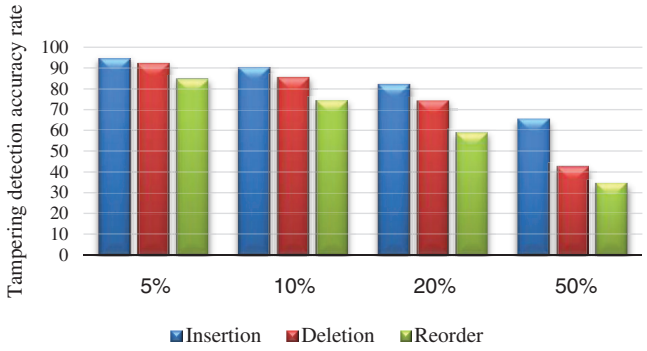

Figure 7: Accuracy evaluation of INLPETWA under all volumes of all attacks

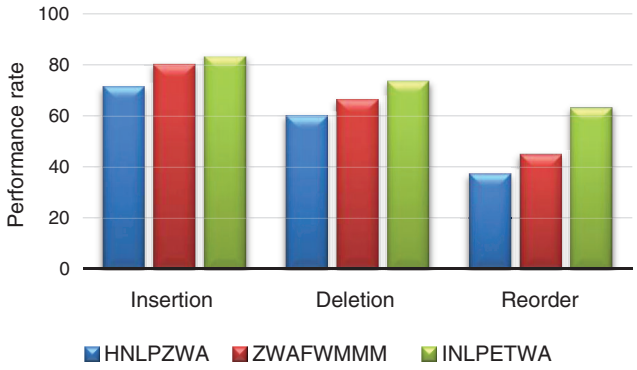

Figure 8: Performance compression under attack type effect

5 Comparison and Discussion

The performance and tampering detection accuracy results are critically analyzed, effect study and compared between INLPETWA and baseline approaches ZWAFWMMM and HNLPZWA and shows discussion of their effect under the major factors, i.e., attack volumes and types, and dataset size.

5.1 Comparison of Attack Type Effect

Tab. 4 shows a comparison of the different attack types effect on performance of INLPETWA, ZWAFWMMM and HNLPZWA approaches.

Tab. 4 and Fig. 8 show how the performance of INLPETWA, ZWAFWMMM and HNLPZWA methods are affected by tampering attacks type. Comparison result show that, the proposed INLPETWA approach

outperforms ZWAFWMMM and HNLPZWA in term of performance rate and watermark robustness in all scenarios of all attack types. This means that the proposed INLPETWA method is a strongly recommended to detection any illegal tampering of English text documents under all attack types.

Table 4: Attack type effect on performance of INLPETWA, ZWAFWMMM and HNLPZWA approaches

Attack type	Approach		
	ZWAFWMMM	HNLPZWA	INLPETWA
Insertion	80.02	71.28	82.99
Deletion	66.25	59.99	73.53
Reorder	44.88	37.23	63.04

5.2 Comparison of Attack Volume Effect

Tab. 5 shows a comparison of the different attack volume effect on performance of INLPETWA, ZWAFWMMM and HNLPZWA approaches.

Table 5: Attack volume effect on performance of INLPETWA with ZWAFWMMM and HNLPZWA

Attack volume (%)	Approach		
	ZWAFWMMM	HNLPZWA	INLPETWA
5	83.60	82.09	90.45
10	74.33	72.74	83.24
20	59.39	57.71	71.66
50	37.56	13.66	47.40

Tab. 5 and Fig. 9 show how the performance of INLPETWA, ZWAFWMMM and HNLPZWA methods are affected by all attack volumes. In case of HNLPZWA, the effect of tampering detection accuracy highly increased under high attack volume. However, it is approximately equal to effect of ZWAFWMMM under low and mid attack volumes. In Fig. 9, it can be seen that if the attack volume increases, the tampering detection accuracy also increases. In all cases of low, mid and high attack volumes, it seen also, the proposed INLPETWA approach outperforms ZWAFWMMM and HNLPZWA approaches in terms of performance and watermark robustness. This means that INLPETWA approach is a strongly recommended to detection any illegal tampering of English text under all volumes of attack types.

5.3 Comparison of Dataset Size Effect

In this subsection, authors present an evaluation of the different dataset size effects on performance of INLPETWA, ZWAFWMMM and HNLPZWA approaches against all attack types under their different volumes as shown in Tab. 6.

The comparative results as shown in Fig. 10 reflects the performance of INLPETWA with baseline ZWAFWMMM and HNLPZWA approaches. The results show that in the proposed INLPETWA approach, the highest effects of dataset size that lead to the best performance are ordered as AMST, AHMST, ALST and ASST respectively. This means that the performance increased with increasing

document size and decreased with decreasing document size. On the other hands, results show that the proposed INLPETWA approach outperforms both ZWAFWMMM and HNLPZWA approaches in term of performance rate under all scenarios of dataset sizes.

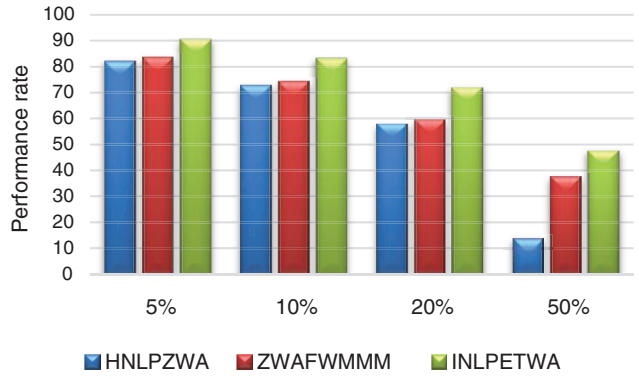

Figure 9: A compression of attack volume effect on performance

Table 6: Dataset size effect on general performance of INLPETWA with ZWAFWMMM and HNLPZWA

Dataset size	Approach		
	ZWAFWMMM	HNLPZWA	INLPETWA
[ESST]	69.53	67.27	68.83
[EMST]	68.13	63.80	78.72
[EHMST]	65.11	59.23	73.15
[ELST]	62.07	54.47	72.30

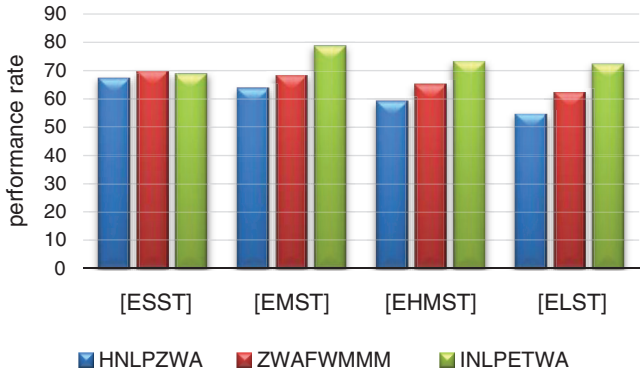

Figure 10: A compression of dataset size effect on performance

6 Conclusions

Centered on the hidden Markov model mechanism of second gram and word method, a novel hybrid approach of NLP and English text zero-watermarking has been developed which is abbreviated as INLPETWA has been proposed in this paper by integrating soft computing and digital watermarking techniques. soft computing and NLP used in INLPETWA to perform text analysis process to found

interrelationships between the content of the English-text provided and the main watermark created. Without modification or impact on plain text size, the created watermark should logically be embedded in the original English background. Hidden watermark will be used in the next phase to detect illegal tampering on received English-text after transmission of text through the Internet. INLPETWA approach has been developed and implemented in PHP using VS code IDE. The experiments are performed on different standard English datasets using various rates of insertion, reorder, and deletion attacks. The experiments results show that INLPETWA is applicable to detect tampering on English text. For future work, authors will intend to improve the performance using other mechanism 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.

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. 8011–8018, 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, 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," *ELSEVIER Optik*, vol. 145, pp. 655–671, 2017.

[7] N. Hurrah, A. Shabir, A. Nazir, A. Javaid, M. Elhoseny *et al.,* "Dual watermarking framework for privacy protection and content authentication of multimedia," *ELSEVIER 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, *StegNmark: A ioint stego-watermark approach for early tamper detection*, vol. 660. Switzerland: Springer International Publishing, 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," *ELSEVIER Information Processing and Management*, vol. 56, no. 2, pp. 367–380, 2017.

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

[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," *ELSEVIER Future Generation Computer Systems*, vol. 108, pp. 935–949, 2020.

[14] 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.

[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," *ELSEVIER 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 sun letters 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] A. Nasr, A. Wan, R. Abdul, S. Khairulmiz 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.

[22] M. Bashardoost, M. Rahim, T. Saba and A. Rehman, "Replacement attack: A new zero text watermarking attack," *3D Research*, vol. 8, no. 1, pp. 11, 2017.

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

[24] P. Zhu, G. Xiang, W. Song, A. Li, Y. Zhang *et al.,* "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] A. Zulfiqar, H. Shamim, M. Ghulam and A. Muhammad, "New zero-watermarking using Hurst exponent for protection of privacy in telemedicine," *IEEE Access*, vol. 6, pp. 1–11, 2018.

[26] 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.

[27] 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.

[28] N. Fahd, Z. Adnan and U. Kulkarni, "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, 2014.

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

[30] N. Fahd, M. Khalid and N. Nadhem, "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, pp. 1–15, 2020.