International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 7– No. 1, April 2014 – www.ijais.org

Content Authentication of English Text via Internet using Zero Watermarking Technique and Markov Model

Fadl M. Ba-Alwi
Associate professor, Faculty of Computer and Information Technology, Sana'a University, Yemen, P.O.Box 1247

Mokhtar M. Ghilan
Assistant professor, Information Systems Department, Faculty of Computer and Information Technology, Sana'a University, Yemen

Fahd N. Al-Wesabi
PhD Candidate, Faculty of Engineering, SRTM University, Nanded, INDIA, Assistant teacher, Department of IT, Faculty of Computing and IT, UST, Sana'a, Yemen

ABSTRACT
In the study of content authentication and tamper detection of digital text documents, there are very limited techniques available for content authentication of text documents using digital watermarking techniques. A novel intelligent text zero watermarking approach based on probabilistic patterns has been proposed in this paper for content authentication and tamper detection of English text documents. In the proposed approach, Markov model of order THREE and letter-based was constructed and abbreviated as LNMZW3 for text analysis and utilizes the interrelationship between contents of given text to generate the watermark. However, we can extract this watermark later using extraction and detection algorithms to identify the status of text document such as authentic, or tampered. The proposed approach was implemented using PHP Programming language with Net Beans IDE 7.0. Furthermore, the effectiveness and feasibility of our LNMZW3 approach has proved and compared with other recent approaches with experiments using five datasets of varying lengths and different volumes of attacks. Results show that the proposed approach is always detects tampering attacks occurred randomly on text even when the tampering volume is low, mid or high. Comparative results with the recent approaches shows that the our LNMZW3 approach provides added value under random insertion and deletion attacks in terms of performance, watermark robustness and watermark security. However, it is provide worst enhancement under reorder attacks.

General Terms
Digital Watermarking, Watermark Robustness, Information Hiding, Probabilistic Patterns.

Keywords
Markov Model, Content Authentication, Tampering Detection.

1. INTRODUCTION
With the increasing use of internet, e-commerce, and other efficient communication technologies, the copyright protection and authentication of digital contents, have gained great importance. Most of these digital contents are in text form such as email, websites, chats, e-commerce, eBooks, news, and short messaging systems/services (SMS) [1].

These text documents may be tempered by malicious attackers, and the modified data can lead to fatal wrong decision and transaction disputes [2].

Content authentication and tamper detection of digital image, audio, and video has been of great interest to the researchers. Recently, copyright protection, content authentication, and tamper detection of text document attracted the interest of researchers. Moreover, during the last decade, the research on text watermarking schemes is mainly focused on issues of copyright protection, but gave less attention on content authentication, integrity verification, and tamper detection [4].

Various techniques have been proposed for copyright protection, authentication, and tamper detection for digital text documents. Digital Watermarking (DWM) techniques are considered as the most powerful solutions to most of these problems.

The digital watermarking techniques have emerged as a solution to breaches of copyright protection, tampering detection, and content authentication of digital media. Researchers have proposed various watermarking approaches for images, audio, and video. However, watermarking approaches for text is very inadequate. The reason behind is the difficulty to watermark text.

Traditional text watermarking techniques for tampering detection and text authentication such as format-based, content-based, and image-based have a number of limitations. In other words, they are not applicable under random tampering attacks on all types of texts, and they need to use some transformations or modifications on contents of the text document to embed watermark information within the original text document itself which results in text capacity, quality, meaning, and value degradation.

Text watermarking algorithms such as Word Length Zero-Watermarking, Non-Vowel ASCII Characters Zero-Watermarking, and content based Zero-Watermarking are not applicable under random tampering attack to all types of text documents. These algorithms are restricted to only alphabetical watermarks, which means they are not applicable to specialized texts such as legal documents, web contents, and documents containing financial, accounting, and mathematical notations. All makes these algorithms impractical. Furthermore, text watermarking algorithms using binary text image are not robust against retyping attacks. The text watermarking methods based on semantics are language dependent and do not provide a complete practical solution for all types of texts.

The English text zero watermarking algorithm based on the probabilistic weight of Markov model presented in [24] has a number of limitations in that it is not applicable under all tampering attacks to all type of documents. However, we can

make the approach more robust, secure, and more efficient by extending the character set and including other mechanisms and various orders of Markov model for English text analysis.

In this paper, the authors present a new zero-watermarking technique for content authentication and tampering detection of English text documents. This technique utilizes the natural language processing to extract the probabilistic patterns based on third order 3-gram of Markov model. This approach is abbreviated here as LNMZW3.

This approach provides interesting extensions to the most important limitations of some of the existing approaches. It can be used to solve the natural language rules, redundancy in texts to hide information, and detect of random tampering attack problems.

The paper is organized as follows. Section 2 provides an overview of the previous work done on text watermarking. The proposed approach is described in detail in section 3. Section 4 presents the experimental and comparative results for the various tampering attacks such as insertion, deletion and reordering. Performance of the proposed approach is evaluated by multiple text datasets. The last section concludes the paper along with directions for future work.

2. PREVIOUS WORK

Text watermarking techniques have been proposed and classified by many literatures based on several features and embedding modes of text watermarking. We have examined briefly some traditional classifications of digital watermarking as in literatures. These techniques involve text images, content based, format based, features based, synonym substitution based, and syntactic structure based, acronym based, noun-verb based, and many others of text watermarking algorithms that depend on various viewpoints [1][3][4].

2.1 Format-based Techniques

Text watermarking techniques based on format are layout dependent. In [5], proposed three different embedding methods for text documents which are, line shift coding, word shift coding, and feature coding. In line-shift coding technique, each even line is shifted up or down depending on the bit value in the watermark bits. Mostly, the line is shifted up if the bit is one, otherwise, the line is shifted down. The odd lines are considered as control lines and used at decoding. Similarly, in word-shift coding technique, words are shifted and modify the inter-word spaces to embed the watermark bits. Finally, in the feature coding technique, certain text features such as the pixel of characters, the length of the end lines in characters are altered in a specific way to encode the zeros and ones of watermark bits. Watermark detection process is performed by comparing the original and watermarked document.

2.2 Content-based Techniques

Text watermarking techniques based on content are structure-based natural language dependent [4]. In [6][14], a syntactic approach has been proposed which use syntactic structure of cover text for embedding watermark bits by performed syntactic transformations to syntactic tree diagram taking into account conserving of natural properties of text during watermark embedding process. In [18], a synonym substitution has been proposed to embed watermark by replacing certain words with their synonyms without changing the sense and context of text.

2.3 Binary Image-based Techniques

Text Watermarking techniques of binary image documents depends on traditional image watermarking techniques that based on space domain and transform domain, such as Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), and Least Significant Bit (LSB) [5]. Several formal text watermarking methods have been proposed based on embedding watermark in text image by shifting the words and sentences right or left, or shifting the lines up or down to embed watermark bits as it is mentioned above in section format-based watermarking [5][7].

2.4 Zero-based Techniques

Text watermarking techniques based on Zero-based watermarking are content features dependent. There are several approaches that designed for text documents have been proposed in the literatures which are reviewed in this paper [1][19] [20] and [21].

The first algorithm has been proposed by [19] for tamper detection in plain text documents based on length of words and using digital watermarking and certifying authority techniques. The second algorithm has been proposed by [20] for improvement of text authenticity in which utilizes the contents of text to generate a watermark and this watermark is later extracted to prove the authenticity of text document. The third algorithm has been proposed by [1] for copyright protection of text contents based on occurrence frequency of non-vowel ASCII characters and words. The last algorithm has been proposed by [21] to protect all open textual digital contents from counterfeit in which is insert the watermark image logically in text and extracted it later to prove ownership. In [22], Chinese text zero-watermark approach has been proposed based on space model by using the two-dimensional model coordinate of word level and the sentence weights of sentence level.

2.5 Combined-based Techniques

One can say the text is dissimilar image. Thus, language has a distinct and syntactical nature that makes such techniques more difficult to apply. Thus, text should be treated as text instead of an image, and the watermarking process should be performed differently. In [23] A combined method has been proposed for copyright protection that combines the best of both image based text watermarking and language based watermarking techniques.

The above mentioned text watermarking approaches are not appropriate to all types of text documents under document size, types and random tampering attacks, and its mechanisms are very essential to embed and extract the watermark in which maybe discovered easily by attackers . On the other hands, these approaches are not designed specifically to solve problem of authentication and tamper detection of text documents, and are based on making some modifications on original text document to embed added external information in text document and this information can be used later for various purposes such as content authentication, integrity verification, tamper detection, or copyright protection. This paper proposes a novel intelligent approach for content authentication and tamper detection of English text documents in which the watermark embedding and extraction process are

performed logically based on text analysis and extract the features of contents by using hidden Markov model in which the original text document is not altered to embed watermark.

3. THE PROPOSED APPROACH

This paper proposes a novel intelligent approach based on third order 3-gram of Markov model and is named as LNMZW3. In our LNMZW3 approach, the original text document is not altered to embed watermark, that means the watermark embedding process is performed logically. The proposed approach uses the Markov model of the natural languages that is Markov chains which are used to analyze the contents of English text documents and extract the probability features of interrelationships between these contents as probabilistic patterns based on letter mechanism and third order of Markov model which are utilized to generate the watermark key that is embedded logically within the original text document or stored in the watermark database. This watermark key can be used later and matched with watermark generated from attacked document for identifying any tampering that may happen to the document and authenticating its content. This process illustrated in figure 1 below.

Before we explain the watermark generation and extraction processes in the next subsection, we present a preliminary mathematical description of Markov models of natural language text analysis.

3.1 LNMZW3 Approach for Text Analysis

This subsection explains how to model English text using our LNMZW3 approach with 3-gram order of Markov model. For example, we have an English sentence as shown in Figure 2 below.

When we are using 3-gram, each sequence of three unique consecutive letters in the given text is a state by itself. The process of Markov transitions from state to state as the text is read.

For instance, When we are using 3-gram order of Markov model named here as LNMZW3, and as the above sample text is processed, the system generates a list of all unique letter-level of 3-gram order as a possible states as the following:

"the", “he ", "e q", " qu", "qui ", "uic", "ick ", "ck ", "k b ", "kbr", "bro ", "row", "own ", " wnf", " nfo", …, “dea”

As this sample text is processed, and if the Markov chain is currently at “the q” state, the possible transitions for each a state that could come next are:

“the→**,** the→**q,** the→**u,** the→**i,** the→**c,** the→**k,** the→**,**the→**b,** the→**r,** the→**o,** the→**w,** the→**n,**…, the→**d”**

The previous scenario of our LNMZW3 approach show that the algorithm used to build possible states and transitions of the Markov chain matrix M[i][j]. For instance, the sample English text shown above contains 94 character with spaces.

After being processed by the system, and represented in the Markov chains by applying the equations (1), (2), (3) and (4), we get for 49 unique states U_S, and we can get for 21600 possible states P_S as sets of three consecutive letters, 61 possible transitions for each state T_S, and 2989 possible transitions for all states P_T.

$$P_S = (n - 3)^3 \quad (1)$$

$$U_S = (n - R_S - 3)^3 \quad (2)$$

$$T_S = \sum_{i=1}^{i=n} U_S(i) + 3 + (3 - 1) \quad (3)$$

$$P_T = P_S * T_S \quad (4)$$

Where,

- n: is the total count of all character sets in the given text document.
- P_S: refers to the total of all possible states.
- U_S: is the total of all unique states (letter sets) within the given text document.
- R_S: is the count of all repeated states within Markov chain matrix.
- T_S: refers to all possible transitions for each state.
- P_T: refers to the total of possible transitions for all states.

As a result of 3-gram order of Markov model based on letter level to analyze the above sentence, we obtain Figure 3 below.

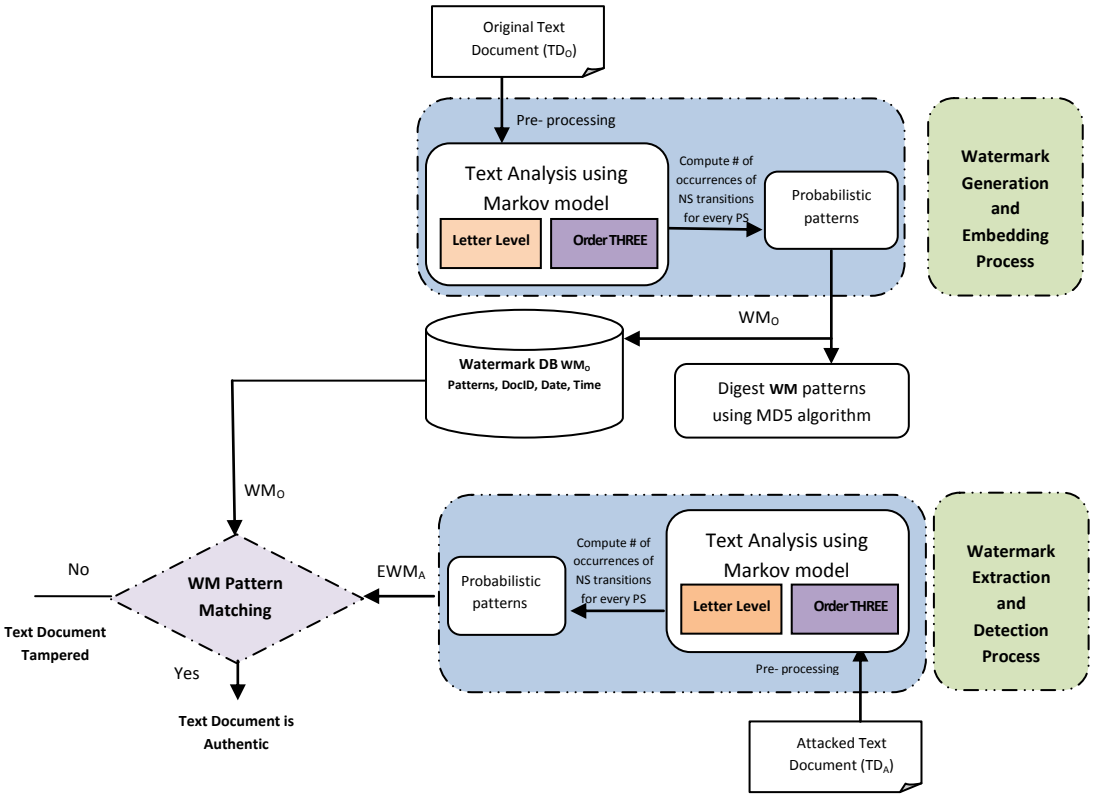

Fig 1: Watermark generation and detection processes

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

Fig 2: Sample of English text

State ID	State	Transitions
1	(" br"):	(['o', 'o', 'o'],
2	("bro"):	['w', 'w', 'w'],
3	("dea"):	['d'],
4	("ick"):	[' '],
5	("jum"):	['p', 'p'],
…	…	…
…	…	…
…	…	…
48	("x j"):	['u'],
49	("x w"):	['h', 'h'])

Fig 3: Sample text states and transitions of LNMZW3

We can see from Figure 3 above that there are 49 unique states, and 91 actual transitions for a 3-gram order of Markov model after processed to analyzing the above given sentence.

Now if we consider state "jum" from the Figure 3 above, the next state transitions are "p", "p", which indicates that the transition "p" occurs twice.

Next we present a simple method to build the states and the Markov transition matrix M[i, j]which is the most basic part of text analysis using Markov model.

In the proposed approach LNMZW3, the text considered is not limited to alphabetic characters, but includes spaces, numbers, and special characters such as [, . ; : - ? !].The total number of states depends on size of given text document (n) which equal n-2, and the total number of states is 61.

These are [English letters = 26, space letter= 1, Integer numbers from 0 to 9= 10, specific symbols such as . ' " , ; : ? ! / \ @ $ & % * + - = >< ( ) [ ] = 24].The entry M[i, j] will be used to keep track of the number of times that the i^{th}characterset of the text is followed by the j^{th}character of the text.Fori = 1 to L − 1, where Lis the length of the text document - 1", let x be the ith character set in the text and y be the (i+1)st character in the text. Then increment M[x,y]. Now the matrix M contains the counts of all transitions. Next we turn these counts into probabilities as follows, for each i from 1 to 61, sum the entries on the ith row, i.e., let counter[i] = M[i,1] + M[i,2] + M[i,3] + ... + M[i,61] .

Now define P[i,j] = M[i,j] / counter[i] for all pairs i,j. This just gives a matrix of probabilities. In other words, now P[i,j] is the probability of making a transition from character set i to character j. Hence a matrix of probabilities that describes a Markov model of order three for the given text is obtained.

3.2 Watermark Generation and Embedding Algorithm

The watermark generation and embedding algorithm requires the original text document as input, then as a pre-processing step it is required to perform conversion of capital letters to small letters and to remove all spaces within the text document. A watermark pattern is generated as the output of this algorithm. This watermark is then stored in watermark database along with the original text document, document identity, author name, current date and time.

This stage includes two main processes which are watermark generation and watermark embedding. Watermark generation from the original text document and embed it logically within the original watermark will be done by the embedding algorithm.

In this proposed watermark generation algorithm, the original text document (T) is to be provided by the author. Then text analysis process should be done using Markov model to compute the number of occurrences of the next state transitions (ns) for every present state (ps). Markov model of order three. A Matrix of transition probabilities that represents the number of occurrences of transition from a state to another is constructed according to the procedure explained in previous section (3.1) and can be computed by equation (5).

$$M[ps][ns] = P[i][j], for\ i,.j=1,2, .,n \qquad (5)$$

Where,

- **n:** is the total number of states
- **i**: refers to PS "the present state".
- **j**: refers to NS "the next state".
- **P[i,j]**: is the probability of making a transition from character set i to character j.

After performing the text analysis and extracting the probability features, the watermark is obtained by identifying all the nonzero values in the above matrix. These nonzero values are sequentially concatenated to generate a watermark patterns, denoted by WMP_O as given by equation (6) and presented in figure 4.

States	Transitions																																																												DWM patterns
	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	0	1	2	3	4	5	6	7	8	9		'	"	,	;	:	?	!	/	\	@	$	&	%	*	+	-	=	>	<	(	)	[	]	
' br'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
'bro'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
'dea'	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
'ick'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
'jum'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
… … …	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
'x j'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
'x w'	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2

Fig 4: Watermark generation processes using LNMZW3

$$WMP_O \&= M\ [ps]\ [ns],\ fori,.\ j= nonzero\ values\ in\ the\ Markov\ matrix \qquad (6)$$

This watermark is then generated as sequential patterns as presented in figure 5, and stored in a watermark database along with the original text document, document identity, author name, current date and time.

DWM Patterns of LNMZW3:
3 – 3 – 1 – 1 – 2 – … – 1 – 2

Fig 5: The original watermark patterns of the given text sample based on LNMZW3

After watermark generation as sequential patterns, an MD5 message digest is generated for obtaining a secure and compact form of the watermark, notational as given by equation (7) and presented in figure 6.

$$DWM = MD5(WMP_O) \qquad (7)$$

DWM Patterns of LNMZW3:
1fd9140249a0d9d0ae1d27a7c5e7ab6e

Fig 6: The original watermark patterns after MD5 digesting

The proposed watermark generation and embedding algorithm of our LNMZW3 approach is presented formally in figure 7.

DWM Generation Algorithm (A Zero Text DWM based on LNMZW3 approach)

- **Input**: Original Text Document ($\mathbf{TD_O}$)
- **Output**: WMP_o value, and States and Transitions Matrix [n-3][61], // **n**: is the length of given text

1. Read the Original Text document (TD_O) and performs Pre-processing for it.
2. Loop ps = 1 to n-2, //Build the states of Markov matrix based on text size
   - Loop ns = 1 to 61, //Build the fixed 61 possible transitions for all states of Markov matrix
     - M[ps][ns] = Total Number of Transition[ps][ns]//compute the total frequencies of transitions for every state.
3. Loop i = 1 to n-3, // Generate original watermark
   - Loop j = 1 to 61,
     - IF M[i][j] != 0 // states that have transitions
     - WMP_O &= M[i] [j]
4. DMM = MD5(WMP_O) // Digest original watermark using MD5 algorithm
5. Output WMP_O, DWM

WMP_o: Original watermark patterns, **n**: is the total number of states, **TD_o**: Original text document array,

M: States and Transitions matrix, **DWM**: Digested watermark, **MD5**: Hash algorithm,

ps: The present state, **ns:** The next state.

Fig 7: Watermark generation and embedding algorithm using our LNMZW3 approach

3.3 Watermark Extraction and Detection Algorithm

The watermark detection algorithm is on the base of zero-watermark, so before detection for attacked text document TDA, the proposed algorithm still need to generate the attacked watermark patterns′. When received the watermark patterns′, the matching rate of patterns′ and watermark distortion are calculated in order to determine tampering detection and content authentication.

This stage includes two main processes which are watermark extraction and detection. Extracting the watermark from the received attacked text document and matching it with the original watermark will be done by the detection algorithm.

The proposed watermark extraction algorithm takes the attacked text document, and performs the same watermark generation algorithm to obtain the watermark pattern for the attacked text document.

After extracting the attacked watermark pattern, the watermark detection is performed in two steps,

- Primary matching is performed on the whole watermark pattern of the original document WMP_O, and the attacked document WMP_A. If these two patterns are found the same, then the text document will be called authentic text without tampering. If the primary matching is unsuccessful, the text document will be called not authentic and tampering occurred, then we proceed to the next step.
- Secondary matching is performed by comparing the components associated with each state of the overall pattern. Which compares the extracted watermark pattern for each state with equivalent transition of original watermark pattern. This process can be described by the following mathematical equations (8),and (9).

$$PMR_T(i,j) = \left| \frac{WMP_O[i][j] - (WMP_O[i][j] - WMP_A[i][j])}{WMP_O[i][j]} \right|$$
$$for\ all\ i.j, \quad 1 \geq PMR_T(i,j) \geq 0 \qquad (8)$$

$$PMR_S(i) = \left| \frac{\sum_{j=1}^{N} (PMR_T(i,j))}{Total\ StatePatternCount(i)} \right|$$
$$for\ all\ i, \quad 1 \geq PMR_S(i) \geq 0 \qquad (9)$$

Where,

- **N:** is the number of non-zero elements in the Markov chain matrix.

Finally, the PMR is calculated by equation (10), which represents the pattern matching rate between the original and attacked text document.

$$PMR = \left| \frac{\sum_{i=1}^{n} PMRS(i)}{n} \right| \qquad (10)$$

Where,

- **n:**is the total number of states.

This process is illustrated in figure 8.

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 state level $PMR_S(i,j)$
					TP1	TP2	
' br'	3	2	2	-	0.6667	-	0.6667
'bro'	3	2	2	-	0.6667	-	0.6667
'dea'	1	1	1	1	-	-	1
'ick'	1	1	1	1	-	-	1
'jum'	2	2	2	1	-	-	1
…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…
'x j'	1	1	1	1	-	-	1
'x w'	2	1	1	-	0.5	-	0.5
Robustness (PMR) =							58.0006 / 72 = 0.8056

Fig 8: Watermark detection using our LNMZW3 approach

The watermark distortion rate refers to tampering amount occurred by attacks on contents of attacked text document, this value represent in WDR which we can get for it by equation (11):

$$WDR = 1 - PMR \quad (11)$$

The detection algorithm is illustrated in figure 9.

DWM Extraction and Detection Algorithm (A Zero Text DWM based on LNMZW3 approach)

- **Input**: Original Text Document, Attacked Text Document
- **Output**: **WMP_O**, **WM_A**, **WMP_A. PMR, WDR,** Attacked States and Transitions Matrix [n-3][61].

1. Read WMo or Original Text(TD_O) and Attacked Text(TD_A) documents and performs Pre-processing for them.
2. Loop ps = 1 to n-3, // Build the states of Markov chain Matrix
- Loop ns = 1 to 61, // Build the transitions for each state in Markov chain Matrix
  - M'[ps][ns] = Total Number of Transition[ps][ns] // compute the total frequencies of transitions for every state

3. Loop i = 1 to n-3, // Extract the embedded watermark
- Loop j = 1 to 61,
  - IF M'[i][j] != 0 // states that have transitions
  - WMP_A &= M'[i] [j]

4. Output WMP_O, WMP_A
5. IF WMP_A = WMP_O
- Print "Document is authentic and no tampering occurred"
  - PMR = 1
- Else
  - Print "Document is not authentic and tampering occurred"
  - For i = 1 to n-3// Extract transition patterns and match each of them with original transition patterns
    - For j = 1 to 61
      - IF WMP_O[i][j] != 0
        - patternCount +=1

$$\mathbf{PMR_T}[i][j] = \left|\frac{\mathbf{WMP_O}[i][j] - (\mathbf{WMP_O}[i][j] - \mathbf{WMP_A}[i][j])}{\mathbf{WMP_O}[i][j]}\right|$$

        - transPMRTotal += PMR_T
      - Else
        - IF WMP_A[i][j] != 0
          - patternCount += WMP_A[i][j]
  - statePMR[i] = $\frac{transPMRTotal[i][j]}{patternCount[i]}$
  - PMR_S += statePMR[i]
- Totalpattern += patternCount

6. PMR = $\frac{PMR_S}{Totalpatterns}$
7. WDR = 1 – PMR

WMP_O: Original watermark, **WMP_A**: Attacked watermark, **M'**: Attacked states and transitions matrix, **TD_A**: Attacked text document array, **ps:** the present state, **ns:** the next state, **PMR_T:** Transition patterns matching rate**, PMR_S:** State patterns matching rate, **PMR**: Watermark patterns matching rate**, WDR**: Watermark distortion rate.

Fig 9: Watermark extraction and detection algorithm using our LNMZW3 approach

4. EXPERIMENTAL SETUP, RESULTS AND DISCUSSION

4.1 Introduction

After we experimented and implemented algorithms of our LNMZW3 approach, we evaluated the accuracy of tampering detection, and compared them with the existing two previous orders of letter level of Markov model based on the criteria of the authentication features, and under the most common nature and volume of possible tampering attacks, which are insertion, deletion, and reorder attacks in multiple random locations of the experimental datasets.

This subsection introduces results discussion and evaluation of our LNMZW3 approach as a contribution of this paper. A methodological description of the process is introduced, as well.

- Evaluation the performance of our LNMZW3 approach, different scenarios of tampering attacks on different dataset sizes, was conducted for different scenarios of the attack volumes and attack types. Then, the performance average was found and compared to current existing approaches which are LNMZW1 and LNMZW2 approaches presented in [25] and [26]. After that, the best performance of which approach was found.
- Results study of attack volume effect on watermark robustness against our LNMZW3 approach and previous approaches (LNMZW1 and LNMZW2) for different scenarios of dataset sizes, and different attack types was conducted. Then, we found the maximum of maximum robustness and compared it in all of these approaches. We also found the attack volume that led to the best watermark robustness of which approach.

Each step mentioned above is presented in a separate subsection in this paper.

4.2 Experimental Parameters and Setup

In order to test the proposed approach and compare it with other approaches, we conducted a series of simulation experiments. The experimental environment (CPU: Intel Core™i5 M480/2.67 GHz, RAM: 8.0 GB, Windows 7, and PHP Programming language with Net Beans IDE 7.0.) is explained below:

First, we present the results study and analysis the performance, watermark robustness, complexity, and watermark capacity of our LNMZW3 approach against different sizes of Standard English text datasets.

Then, the effect of tampering volumes has been studied on our LNMZW3 approach with two current developed approaches and compared their results.

The experiment datasets namely SST4, SST2, MST5, MST2, and LST4. The datasets are a large collection of writings of a specific kind called "Corpora", and a statistically representative sample of natural language texts referred to as "Corpus" and found in [11]. The datasets were categorized, according to their size, into three classes: Small Size Texts (SST), Medium Size Texts (MST), and Large Size Texts (LST). The datasets were used for all attack scenarios of the different common volumes and types. In other words, the datasets were used with 5%, 10%, 20%, and 50% of the attack volumes used for the insertion, detection, and reorder attack types determined in [24].The details of our datasets and attacks used are shown in Table 1.

Table 1. Dataset names and sizes, attacked types and volumes

Dataset name	Dataset size	Attacks Types and Volumes		
		Insertion	Deletion	Reorder
[SST4]	179	5%, 10%, 20%, 50%	5%, 10%, 20%, 50%	5%, 10%, 20%, 50%
[SST2]	421			
[MST5]	469			
[MST2]	559			
[LST4]	2018			

To measure the performance of our LNMZW3 approach and compare it with two current developed approaches which have been developed under the similar environment and criteria, we used the PMR as a measure of the watermark robustness. The robustness value gives us the accuracy of tampering detection of the given text document targeted.

The WDR is also measured and compared between our LNMZW3 approach with two current developed approaches. In addition, the values of both PMR and WDR are arranged from 0 to 1. It appears that the lowest the WDR value is, the largest is the robustness value and vice versa. The desirable value of the PMR is when it is close to 0, and for the WDR is when it is close to 1.

We categorize tamper detection states into three classes based on PMR threshold values which are: High when the PMR values greater than 0.70, Mid when the PMR values are between 0.40 and 0.70, and Low when the PMR values are less than 0.40.

We evaluate the accuracy of our proposed approach LNMZW3 by conducting a series of experiments based on order level of our and existing approaches with all attacks types such as random insertion, deletion, and reorder of words and sentences on each sample of the datasets. These various kinds of attacks were applied at multiple locations in the datasets.

4.3 Performance Evaluation of LNMZW Approach

In this subsection, we present the evaluation for performance of our LNMZW3 approach against dataset size of different scenarios of the attack volumes and attack types. First, we present the results study and analysis of performance for our LNMZW3 approach and previous approaches (LNMZW1 and LNMZW2) against different dataset sizes as Table 1 shows. Then, we find the best performance average of which approach that has the best performance.

4.3.1 Insertion Attack

Insertion attack was examined with the four scenarios of the attack volume, i.e., 5%, 10%, 20%, and 50%.

Table 2 shows the performance averages of our proposed LNMZW3 approach with LNMZW1 and LNMZW2 approaches against all dataset sizes and all scenarios of insertion attack with all volumes.

Table 2. Performance averages of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of insertion attack

Attack Volume	LNMZW1	LNMZW2	Our LNMZW3 approach
5%	92.56	93.25	94.02
10%	86.47	85.88	88.29
20%	78.54	79.13	80.89
50%	59.93	62.30	63.54
Average of performance averages	79.37	80.14	81.68

From Figure 10 below, we can see that our proposed approach LNMZW3 gives the maximum average of performance averages in all scenarios of insertion attack with value 81.68%. This means that the third order of letter based approach named as LNMZW3 is effective under all volumes of insertion attacks.

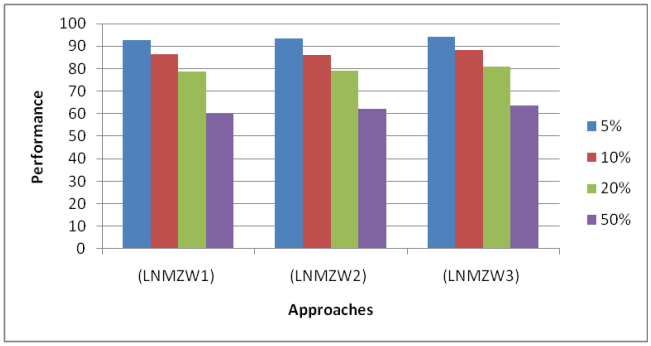

Fig 10: Performance averages of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of insertion attack

Also, we can see the enhancement of the performance with an increase of the order level of Markov model based on letter mechanism as shown in Figure 11.

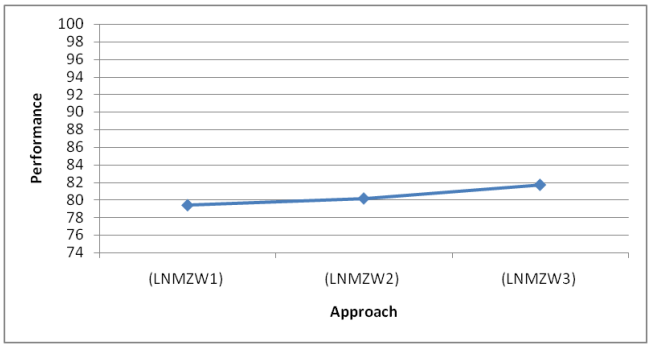

Fig 11: Performance enhancement of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of insertion attack

4.3.2 Deletion Attack

Deletion attack was examined with four scenarios of attack volume, namely 5%, 10%, 20%, and 50%.

Table 3 shows the performance averages of our proposed LNMZW3 approach with LNMZW1 and LNMZW2 approaches against all dataset sizes and all scenarios of deletion attack with all volumes.

Table 3. Performance averages of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of deletion attack.

Attack Volumes	LNMZW1	LNMZW2	Our LNMZW3 Approach
5%	94.25	94.79	96.41
10%	89.41	89.93	92.49
20%	80.06	81.07	85.86
50%	51.69	56.69	67.30
Average of performance averages	78.85	80.62	85.52

As shown by Figure 12, the maximum average of performance averages for all scenarios of deletion attack volumes is 85.52%. This value is produced by our LNMZW3 approach, which refers to the best performance against the deletion attack in terms of performance average.

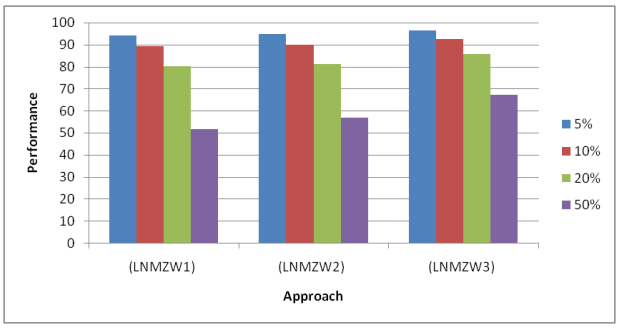

Fig 12: Performance averages of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of deletion attack

In addition, we can see that the enhancement of the performance is due to the increase of the order level of Markov model with letter mechanism as shown in Figure 13.

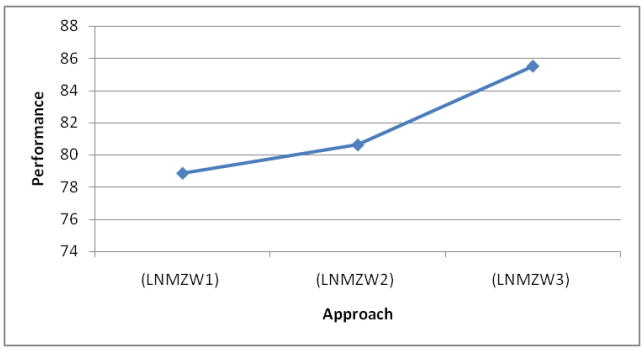

Fig 13: Performance enhancement of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of deletion attack

4.3.3 Reorder Attack

Reorder attack was examined with four scenarios of attack volumes: 5%, 10%, 20%, and 50%.

Table 4 shows the performance averages of our proposed LNMZW3 approach with LNMZW1 and LNMZW2 approaches against all dataset sizes and all scenarios of reorder attack with all volumes.

Table 4. Performance averages of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of reorder attack

Attack Volumes	LNMZW1	LNMZW2	Our LNMZW3 approach
5%	99.27	96.86	95.36
10%	98.75	95.32	91.62
20%	97.52	91.87	86.90
50%	96.12	88.72	80.19
Average of performance averages	97.92	93.19	88.52

As shown by Table 4 and Figure 14, the maximum average of performance averages for all scenarios of reorder attacks with all attack volumes is 97.92%. This value refers to the best performance average against reorder attack shown by LNMZW1 approach.

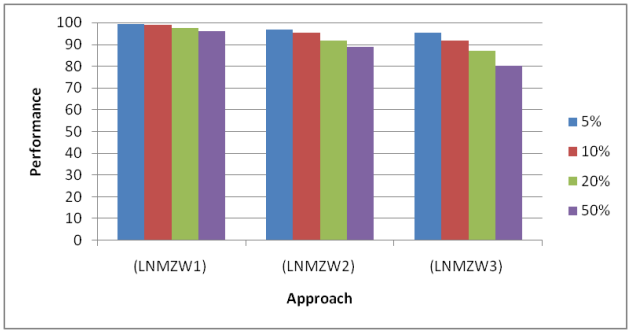

Fig 14: Performance averages of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of reorder attack

We can see that the enhancement of the performance is due to an decrease of the order level of Markov model with letter mechanism as depicted in Figure 15.

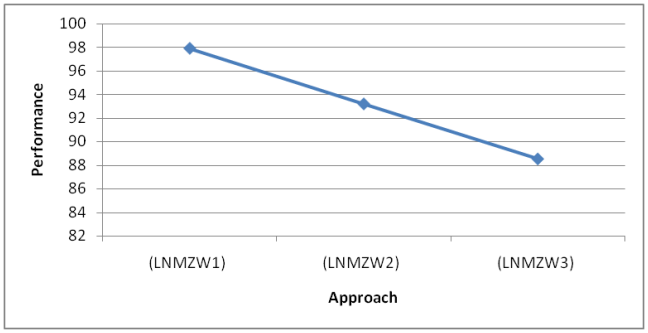

Fig 15: Performance enhancement of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all volumes of reorder attack

As Figure 16 depicts, all order levels of letter mechanism approaches of Markov model, i.e., LMNZW 1, 2, and 3 perform high performance under reorder attack, and LNMZW1 approach perform higher performance than LNMZW2 and LNMZW3 approaches. On the contrary, our LNMZW3 approach perform better performance than LNMZW1 and LNMZW2 approaches under insertion and deletion attacks.

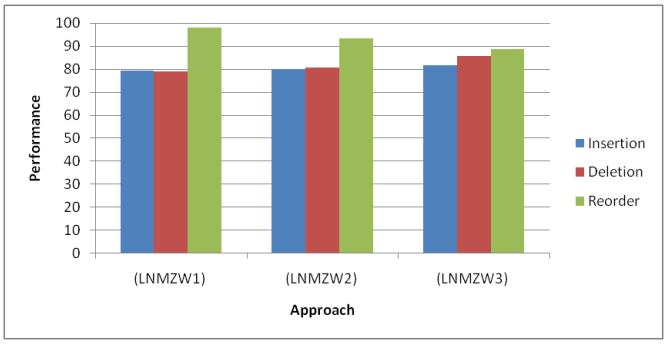

Fig 16: Average of performance averages of our LNMZW3 approach with LNMZW1 and LNMZW2 approaches under all attacks

4.4 Results Study of Volumes Effect of All Attacks

In this subsection, we present an evaluation of the attack volumes effects of all attack types on watermark robustness against our LNMZW3 approach and previous approaches (LNMZW1 and LNMZW2) for different under all scenarios of dataset sizes and attack types.

Table 5 shows the maximum of maximum values of watermark robustness of watermark robustness for our LNMZW3 approach and previous (LNMZW1 and LNMZW2) approaches against the effects of four scenarios of attack volumes categorized as [5%, 10%, 20%, and 50%] under all sizes of datasets and insertion, deletion and reorder attacks.

Table 5. Effect of attack volumes on watermark robustness using our LNMZW3 approach with LNMZW1 and LNMZW2) approaches

Attack Volumes	Insertion		Deletion		Reorder	
	Max Rob.	Best Approach	Max Rob.	Best Approach	Max Rob.	Best Approach
5%	94.62	LNMZW3	97.27	LNMZW3	99.89	LNMZW1
10%	89.25	LNMZW1	93.17	LNMZW3	99.94	LNMZW1
20%	82.10	LNMZW3	87.33	LNMZW3	98.42	LNMZW1
50%	81.03	LNMZW3	72.08	LNMZW3	99.19	LNMZW1

Inviting comparisons with all attack volumes effect in terms of the best watermark robustness under insertion, deletion and reorder attacks as Table 5 and Figure 17 shows, our LMNZW3 approach has the best watermark robustness value under all volumes of deletion attack, and under 5%, 20% and 50% volumes of insertion attack. This means that our LNMZW3 approach is applicable for medium and large volumes of insertion attack, and for all volumes of deletion attacks.

Results also show that the LNMZW1 approach outperforms all approaches in term of the best robustness values with all volumes of reorder attack.

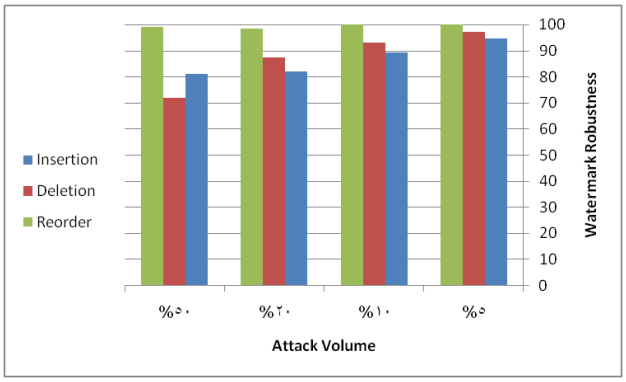

Fig 17: Volumes effect of all attacks on watermark robustness using our LNMZW3 approach, LNMZW1 and LNMZW2) approaches

5. CONCLUSION

Based on the Markov model for English Text analysis based on order three of letter mechanism, we designed a text zero watermark approach as a soft computing tool to analyze English text and find the inter-relationship of contents of the text documents according to the probabilistic patterns of states and transitions to finally generate and detect a watermark.

The experiment showed that the proposed approach had better performance and robustness, and it is more secure than other approaches especially in case of insertion and deletion attacks.

It is proved that the performance of our LNMZW3 approach was enhanced with high levels of Markov order in terms of insertion and deletion attacks, in which the third order LNMZW3 of Markov model had the best performance.

The attack volume effect against our LNMZW3 approach with the recently proposed watermark approaches named as (LNMZW1 and LNMZW2) was examined. The results showed that our LNMZW3 approach was recommended for tampering detection against small, medium and high volumes of insertion, and deletion attacks. However, the LNMZW1 was found to be applicable only under small volumes of insertion attack and it is recommended under all volumes of reorder attack.

6. REFERENCES

[1] . JaliI, A. Hamza, S. Shahidm M. Arif, A. Mirza, 2010. A Zero Text Watermarking Algorithm based on Non-Vowel ASCII Characters. International Conference on Educational and Information Technology (ICET 2010). IEEE.

[2] Suhail M. A., 2008. Digital Watermarking for Protection of Intellectual Property. A Book Published by University of Bradford. UK.

[3] L. Robert, C. Science, C. Government Arts, 2009. A Study on Digital Watermarking Techniques. International Journal of Recent Trends in Engineering. Vol. 1. No. 2. pp. 223-225.

[4] X. Zhou, S. Wang, S. Xiong, 2009. Security Theory and Attack Analysis for Text Watermarking. International Conference on E-Business and Information System Security. IEEE, pp. 1-6.

[5] T. Brassil, S Low, and N. F. Maxemchuk, 1999. Copyright Protection for the Electronic Distribution of Text Documents. Proceedings of the IEEE. vol. 87. no. 7. pp. 1181-1196.

[6] M. Atallah, V. Raskin, M. C. Crogan, C. F. Hempelmann, F. Kerschbaum, D. Mohamed, and S.Naik, 2001. Natural language watermarking: Design, analysis, and implementation. Proceedings of the a Fourth Hiding Workshop. vol. LNCS 2137. pp. 25-27.

[7] N. F. Maxemchuk and S Low, 1997. Marking Text Documents. Proceedings of the IEEE International Conference on Image Processing. Washington. pp. 13-16.

[8] D. Huang, and H. Yan, 2001. Interword distance changes represented by sine waves for watermarking text images. IEEE Trans. Circuits and Systems for Video Technology. Vol.11. No.12. pp. 1237 - 1245.

[9] N. Maxemchuk, and S. Low, 1998. Performance Comparison of Two Text Marking Methods. IEEE Journal of Selected Areas in Communications (JSAC). vol. 16. no. 4. pp. 561-572.

[10] S. Low, and N. Maxemchuk, 2000, Capacity of Text Marking Channel. IEEE Signal Processing Letters. vol. 7. no. 12. pp. 345 -347.

[11] M. Kim, 2008. Text Watermarking by Syntactic Analysis. 12th WSEAS International Conference on Computers. Heraklion. Greece.

[12] H. Meral, B. Sankur, A. Sumru, T. Güngör, and E. Sevinç, 2009. Natural language watermarking via morphosyntactic alterations. Computer Speech and Language. pp. 107-125.

[13] Z. Jalil, and A. Mirza, 2009. A Review of Digital Watermarking Techniques for Text Documents. International Conference on Information and Multimedia Technology. pp. 230-234 .IEEE.

[14] M. AtaIIah, C. McDonough, S. Nirenburg, and V. Raskin, 2000. Natural Language Processing for Information Assurance and Security: An Overview and Implementations. Proceedings 9th ACM/SIGSAC New Security Paradigms Workshop. pp. 5 1-65.

[15] H. Meral, E. Sevinc, E. Unkar, B. Sankur, A. Ozsoy, and T. Gungor, 2007. Syntactic tools for text watermarking. In Proc. of the SPIE International Conference on Security. Steganography. and Watermarking of Multimedia Contents. pp. 65050X-65050X-12.

[16] O. Vybornova, and B. Macq., 2007. Natural Language Watermarking and Robust Hashing Based on Presuppositional Analysis. IEEE International Conference on Information Reuse and Integration, IEEE,.

[17] M. tallah, V. Raskin, and C. Hempelmann, 2002. language watermarking and tamperproofing. Proc. of al.. Natural 5th International Information Hiding Workshop. Noordwijkerhout. Netherlands. pp.196-212.

[18] U. Topkara, M. Topkara, and M. J. Atallah, 2006. The Hiding Virtues of Ambiguity: Quantifiably Resilient Watermarking of Natural Language Text through

Synonym Substitutions. In Proceedings of ACM Multimedia and Security Conference. Geneva.

[19] Z Jalil, A. Mirza, and H. Jabeen, 2010, Word Length Based Zero-Watermarking Algorithm for Tamper Detection in Text Documents. 2nd International Conference on Computer Engineering and Technology. pp. 378-382. IEEE.

[20] Z Jalil, A. Mirza, and M. Sabir, 2010. Content based Zero-Watermarking Algorithm for Authentication of Text Documents. (IJCSIS) International Journal of Computer Science and Information Security. Vol. 7. No. 2.

[21] Z. Jalil , A. Mirza, and T. Iqbal, 2010. A Zero-Watermarking Algorithm for Text Documents based on Structural Components. pp. 1-5. IEEE.

[22] M.Yingjie, G. Liming, W.Xianlong, and G. Tao, 2011. Chinese Text Zero-Watermark Based on Space Model. In Proceedings of I3rd International Workshop on Intelligent Systems and Applications, pp. 1-5, IEEE.

[23] S. Ranganathan, A. Johnsha, K. Kathirvel, and M. Kumar, 2010. Combined Text Watermarking. International Journal of Computer Science and Information Technologies. Vol. 1 (5). pp. 414-416.

[24] Fahd N. Al-Wesabi, Adnan Alsakaf, and Kulkarni U. Vasantrao, 2012. A Zero Text Watermarking Algorithm based on the Probabilistic weights for Content Authentication of Text Documents. in Proc. On International Journal of Computer Applications(IJCA). U.S.A. pp. 388 – 393.

[25] Fahd N. Al-Wesabi, Adnan Alsakaf, and Kulkarni U. Vasantrao, 2013. A Zero Text Watermarking Algorithm Based on the Probabilistic Patterns for Content Authentication of Text Documents. International Journal of Computer Engineering & Technology (IJCET). India. Vol. 4. Issue 1. pp. 284 – 300.

[26] Fahd N. Al-Wesabi, Adnan Alsakaf, and Kulkarni U. Vasantrao, 2012. English Text Zero-Watermark Based on Markov Model of Letter Level Order Two. Journal of Intelligent Computing. UK. Vol. 3. Issue. 4. pp. 137-155.