Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Semantic-driven watermarking of relational textual databases

Maikel Lázaro Pérez Gort [a], Martina Olliaro [b,c], Agostino Cortesi [a], Claudia Feregrino Uribe [a,*]

[a] Instituto Nacional de Astrofísica, Óptica y Electrónica, Luis Enrique Erro 1, Sta María Tonanzintla, 72840 Puebla, Mexico
[b] Ca' Foscari University of Venice, Scientific Campus, Via Torino 155, 30172 Mestre, Venice, Italy
[c] Masaryk University, Faculty of Informatics, Botanickà 68A, 60200 Brno, Czech Republic

ARTICLE INFO

Keywords:
Digital watermarking
Relational databases
Semantic similarity analysis

ABSTRACT

In relational database watermarking, the semantic consistency between the original database and the distorted one is a challenging issue which is disregarded by most watermarking proposals, due to the well-known assumption for which a small amount of errors in the watermarked database is tolerable. We propose a semantic-driven watermarking approach of relational textual databases, which marks multi-word textual attributes, exploiting the synonym substitution technique for text watermarking together with notions in semantic similarity analysis, and dealing with the semantic perturbations provoked by the watermark embedding. We show the effectiveness of our approach through an experimental evaluation, highlighting the resulting capacity, robustness and imperceptibility watermarking requirements. We also prove the resilience of our approach with respect to the random synonym substitution attack.

1. Introduction

Relational databases are enterprise software systems, introduced in the 1970s (Codd, 1970), where data are stored and eventually analyzed to find hidden relations among them which are useful to take strategic decisions. Data stored in relational databases can be pirated, illegally redistributed, tempered, and their ownership claimed, as happens to all the other digital assets (images, audio, video and texts). Indeed, the internet growth resulted in a multitude of web-based services through which data are continuously transmitted and easily accessible.

How can we protect the integrity and the intellectual property of data? Relational database watermarking (Agrawal and Kiernan, 2002) has been proposed as a new field of security, to protect data property value. The main element it uses is the watermark, i.e., a stream of binary elements called marks, and it consists of two phases: watermark embedding and watermark extraction (Agrawal et al., 2003). Generally speaking, we can classify relational database watermarking techniques into two categories (Mehta and Aswar, 2014): those embedding the watermark in the data, causing distortions [e.g.,] (Jiang et al., 2009; Kamran and Farooq, 2013; Zhang et al., 2011), and those that do not cause them [e.g.,] (Bhattacharya and Cortesi, 2009a; Bhattacharya and Cortesi, 2009b; Guo, 2011). The distortion-based techniques are mostly oriented to resist aggressive attacks as they are conceived to protect data from false ownership claims (Halder et al., 2010).

The first watermarking technique for relational data was proposed by Agrawal and Kiernan (2002). Also known as the AHK algorithm, this distortion-based approach defines the notation of the elements belonging to a relation R (see Table 1), embeds the marks in numerical attributes, introduces a solid criteria for the selection of the places for the watermark embedding, and it has been used by many authors [e.g.,] (Chang et al., 2014; Farfoura et al., 2012; Sun et al., 2008) as starting point for other watermarking proposals.

Relational database watermarking techniques deal with different issues (Halder et al., 2010). Among them, three are of particular interest: the watermark capacity (i.e., the optimal amount of marks that can be embedded in a relational database), its robustness (i.e., the ability of relational watermarking techniques to resist against malicious or unintentionally cyber incidents), and imperceptibility (i.e., the ability of distortion-based relational watermarking techniques to not affect the usability of the data). Each of these features is highly linked to the other two, and this is the reason why it is necessary to consider a trade-off among them in the design of relational watermarking techniques. Indeed, for example, one way to increase the robustness of a watermarking technique is by embedding more marks, which also increases its capacity, but this may compromises the imperceptibility, affecting the data usability and giving clues to attackers of the watermark presence.

[*] Corresponding author.
E-mail addresses: mlazaro2002es@inaoep.mx (M.L. Pérez Gort), martina.olliaro@unive.it (M. Olliaro), cortesi@unive.it (A. Cortesi), cferegrino@inaoep.mx (C. Feregrino Uribe).

https://doi.org/10.1016/j.eswa.2020.114013
Received 6 December 2019; Received in revised form 12 September 2020; Accepted 12 September 2020
Available online 28 September 2020
0957-4174/© 2020 Published by Elsevier Ltd.

Table 1
AHK approach notation.

Symbol	Description
SK	Secret key only known by the data owner.
PK	Primary key identifying the tuple.
η	Number of tuples in a relation R being watermarked.
γ	Fraction of tuples being watermarked $\gamma \in [1, \eta]$. Also known as TF.
ν	Number of attributes considered for the embedding.
ξ	Range of less significant bits (*lsb*) to embed the mark.
ω	Number of watermarked tuples after the embedding.

On the other hand, the higher the imperceptibility, the higher is the robustness of the watermark, but this happens despite of its capacity.

1.1. Paper contribution

After the approach of Agrawal and Kiernan (2002), several relational database watermarking techniques were proposed using different types of attributes to embed the watermark. Despite that, dealing with semantic perturbations provoked by the distortion is an issue that has been ignored most of the time. Among the few works addressing this problem there are the techniques by Bertino et al. (2005), Franco-Contreras et al. (2014) and by Franco-Contreras and Coatrieux (2015).

In Bertino et al. (2005), authors proposed a technique to protect the privacy and ownership of medical data. They worked with categorical attributes, and they performed the watermark embedding as permutations of categorical values using a Domain Hierarchy Tree (DHT) of the selected attribute. This scheme does not consider more complex data types (e.g., multi-word textual[1]), and despite the use of the DHT for reducing the semantic perturbations, the changes compromise the inter-attribute semantic consistency. Furthermore, the latter approach reduces the watermark capacity by marking only one attribute per tuple, and if there was no DHT, then the watermark synchronization would be compromised, as the DHT is the structure on which the embedding and the extraction processes rely.

In Franco-Contreras et al. (2014) and Franco-Contreras and Coatrieux (2015) the distortion is controlled by using ontologies, seeking the preservation of the inter-attribute semantic consistency. In both the approaches, the requiring ontologies (defined for specific contexts) directly impact the blindness of the techniques, and make the watermark synchronization dependant on additional external information. Furthermore, just a single numerical attribute per tuple is selected to embed the watermark, which limits the application of the ontologies, considering the potential they may offer to increase the watermark capacity, among other things. In the end, these approaches depend on the PK of R to embed the watermark, which makes easy to compromise the watermark detection in scenarios where R is used separately from the database.

In this paper, we propose a semantic-driven approach for watermarking multi-word textual attributes in relational databases, to protect their ownership. In order to preserve the meaning, fluency, grammaticality, writing style and value (Jalil and Mirza, 2009) of the multi-word textual attributes, the distortion is meant as a substitution of words that are strongly semantically similar in a certain context (i.e., synonyms).

Our aims are to achieve a high degree of robustness without compromising the imperceptibility, increment the capacity taking care of the semantic of the data, and preserve the results of the queries performed over the watermarked data in comparison to those that are obtained using the same queries over the unwatermarked data. This avoids that the distortion caused by the watermark embedding from affecting the decision making of the organizations using and deploying the data.

The way our approach is conceived allow us to achieve a high watermark synchronization, making our scheme resilient against attacks based on the elimination of tuples and/or attributes. On the other hand, by involving other aspects to the watermarking process (e.g., the elements forming the relation structure, information corresponding to other data types, and the low redundancy of the stored data), our technique gets resilient against the *random synonym substitution attack*. Our approach's features increase the chaotic nature of the mark embedding places selection, making difficult for attackers to determine their locations and to compromise the watermark detection by overwriting them.

We introduce also a novel approach to analyze watermark capacity through the calculation of the index c_w, which expresses the technique‘s resilience degree to malicious operations. This new measure is described in Section 4 highlighting its differences to traditional capacity measurement for relational data watermarking. Combining our proposal with numeric cover type relational watermarking techniques, allows the increasing of the watermark capacity without compromising its imperceptibility, which increases also the robustness, making our technique effective for any practical scenarios.

1.2. Paper structure

The rest of the paper is organized as follows. Section 1 defines the motivating examples that let clear the need of the elements we introduce in our technique. Section 3 gives an overview about semantic similarity theory and text watermarking techniques. Section 4 introduces our watermarking approach, emphasizing the definition of the elements related to the preservation of the semantic consistency, which constitutes a special feature of our work. Details related to the implementation are also given. Section 5 presents the validation of our approach through an experimental evaluation. Section 6 concludes.

2. Motivating examples

Using numeric attributes to embed the watermark gives high coverage to relational watermarking techniques, making possible the increment of the watermark capacity. Nevertheless, accomplishing the imperceptibility requirement at plain sight, numerical distortion compromises SQL query results based on numeric conditions.

Example 1. Consider the relation **Student** depicted in Table 2, where the attribute **IdNumber** denotes the primary key of the relation itself. According to the query below for selecting the students who have passed a certain grade (**Score** $\geqslant$70), only the students {John, Andrea, Karla} are recovered.

```
SELECT StudentName
FROM Student
WHERE Score ⩾70
```

In the case in which the attribute **Score** is selected to embed a mark, despite performing a passive distortion, e.g., by just using the two less significant bits (*lsb*), the result of the query above would be different. Indeed, Justin Fitzgerald could be given among the students passing the grade if the value of the 2^{nd} *lsb* is modified, changing, for example, the score from 69 to 71.

The distortion caused by the change of the *lsb* of numerical values in a relation is not relevant when the values of the attribute chosen to embed the mark are not in the boundaries of some criteria for data recovery or their classification (e.g., changing the score of Andrea from 98 to 96 or 99, depending of the *lsb* selected as mark carrier, would not produce a different answer to the query above). But when this is not the case, such a distortion may lead to taking decisions based on wrong assumptions. Therefore, it follows that numerical distortions compromise the semantic of the tuples, despite the latter distortions being

[1] We refer to multi-word attributes as textual attributes formed by one or more than one sentence.

Table 2
Motivating example.

Student					
IdNumber	StudentName	StudentSurname	Subject	Score	ProfessorJudgment
1001	John	Oliver	Mathematics	95	John has improved a lot.
1002	Justin	Fitzgerald	Physics	69	Justin has problems to pass Physics.
1003	Andrea	Russo	History	98	Andrea is the first in his History class.
1004	Karla	Olivare	Mathematics	100	Karla is an outstanding student.

traditionally controlled by defining the maximum amount of tolerable error over the numerical attributes being watermarked.

Embedding the marks in textual attributes avoids compromising the results of queries based on numerical conditions, but applying the distortion over the *lsb* of a textual value compromises the watermark imperceptibility requirement.

Example 2. Given the relation defined in Table 2, consider the value of the attribute **ProfessorJudgment** for the tuple with **IdNumber** $= 1002$, i.e., “Justin has problem to pass Physics.”. Changing one of the two *lsb* of this textual value will provoke changing “Physics” to “Physicr” or “Physicq” making perceptible the distortion and creating a meaningless word.

Notice that, even when the marks are embedded in textual attributes exploiting the limitations of the human vision for increasing the watermark capacity (e.g., by adding extra white spaces between words (Al-Haj and Odeh, 2008) or using invisible characters according to the database encoding (Melkundi and Chandankhede, 2015)), is easy for the attacker to detect the position of the marks through computational techniques. Thus, for the aforementioned reasons, when the marks have to be embedded into textual attributes, a different approach is required.

3. Prerequisites

As mentioned in Section 2.1, the relational watermarking technique we propose marks multi-word textual attributes and preserves the semantic consistency between the original database and the watermarked one performing semantically similar words substitutions, taking care of the context in which the words fall. Thus, below we give a brief overview of semantic similarity theory and semantic-based text watermarking techniques.

3.1. Semantic similarity theory

Semantic similarity is about computing the resemblance between the meanings of textual entities (e.g., words, sentences, texts), that are not necessarily lexically similar (Batet and Sánchez, 2015; Hliaoutakis et al., 2006) and it has application in many research fields (Agirre et al., 2009; Petrakis et al., 2006; Seco et al., 2004; Taieb et al., 2014) such as: Natural Language Processing tasks (e.g., Word Sense Disambiguation and Synonym Detection), Artificial Intelligence, Cognitive Science, Psychology, Information Retrieval and Bio-Informatics.

In the literature, several methods for measuring semantic similarity between textual entities have been proposed (Hliaoutakis et al., 2006). They depend on one or several knowledge sources (e.g., taxonomies, thesaurus, ontologies) and rely on different theoretical properties (Batet and Sánchez, 2015). A measure of similarity takes as input two textual entities and returns a numeric score that quantifies how much they are alike (Taieb et al., 2014). Formally,

let e_1 and e_2 be two textual entities,

$ss(e_1, e_2)$ denotes the semantic similarity between e_1 and e_2.

Different words that have highly related meanings are called synonyms. Generally speaking, synonym words belong to the same node in a hierarchical knowledge organization scheme and their semantic similarity is maximized (Slimani, 2013).

3.2. Text watermarking

According to classifications of text watermarking techniques presented in (Taleby Ahvanooey et al., 2018; Jalil and Mirza, 2009; Kamaruddin et al., 2018; Zhou et al., 2009), we summarise them as follows:

(a) Image-based approaches, where a text document, whose content is seen as a series of text images, is used to embed the watermark bits. In general, these methods are resilient against typical image watermarking attacks and format-based attacks.
(b) Structure-based approaches, where imperceptible changes to the text structure, features and font are made, into which the watermarking information to be hidden is encoded. These techniques are more likely to be vulnerable to very simple attacks, such as: the text retyping attack and the copy paste to notepad attack, and to the use of Optical Character Recognition (OCR) technologies.
(c) Syntactic-based approaches apply syntactic transformation to plain text document structures in order to embed the watermark. These schemes have been proved to be efficient with agglutinative languages like Turkish, but in general they are not adequate for English language.
(d) Semantic-based approaches embed the watermark into the semantic structure of text documents, where text meaning analysis and text transformations are performed using natural language processing algorithms. Semantic-based text watermarking schemes are resilient against retyping attacks or to the use of OCR programs, but prone to weaknesses related to natural language processing.

Notice that syntactic-based and semantic-based text (or content-based) watermarking schemes fall into the linguistic-based approach category. Therefore, they are highly dependant on the type of language in use, which represents a disadvantage in a scenario where languages rapidly evolve, and focused on do not (or minimally) alter the meaning of the cover text.

3.2.1. Synonym substitution approach

Image-based and structure-based approaches are useful when the text is forced to be displayed by using specific means. On the other hand, in the case in which the text is stored as content (e.g., stored as multi-word textual attribute in relational databases), independent from its graphical representation, the text can actually be displayed in multiple ways, making those techniques useless. Consequently, content-based text watermarking techniques are better suited to be used in the context of relational textual database watermarking. In particular, we focused on the synonym substitution method for watermarking textual documents, where certain words are replaced with their synonyms preserving the semantic consistency between the original cover text and the watermarked one.

Firstly exploited by steganography (Winstein, 2000), synonym substitution technique was later used to watermark plain text document (Topkara et al., 2006). Still, this watermarking technique is limited to the English language and highly depends on the quality of the text processing tools, like the word sense disambiguator (i.e., a technique that aims to identify which sense of a word is used in a sentence). Moreover, synonym substitution watermarking techniques are vulnerable to synonym substitution attacks.

In Topkara et al. (2006), to overcome random synonym substitution attacks, authors proposed a lexical watermarking system based on substituting words with homographs[2] from their synonym set, and using meaning-preserving generalizing substitutions. Then, in order to guarantee the context-dependency between synonyms, they implemented a semi-automatic interactive encoding mechanism that allows a person designated to decide on the acceptability of the substitutions given by the system.

4. Semantic-based watermarking approach

As pointed out in Example 2, modifying the *lsb* of a textual attribute value may, for example, introduce syntactic errors or cause semantic inconsistencies, leading to the imperceptibility requirement violation of the watermarking technique.

In this section, we present our semantic-based watermarking approach for relational data. Necessary condition for our technique to be applied is that the target relation must contain at least one multi-word textual attribute into which we will embed the watermark. The distortion is thought as synonym substitution.

Our proposal is focused on increasing the watermark capacity, without affecting its imperceptibility, achieving an high degree of robustness against typical relational watermarking attacks and textual watermarking attacks. Indeed, by considering multi-word textual attributes as cover type, multiple synonym substitutions can be performed over a single attribute value, resulting in the increment of the embedded marks, despite some sentences being composed of just few words, overcoming the downside of watermarking short documents, which reduces the watermark capacity (Jalil and Mirza, 2009). Moreover, the capacity of the watermark increases when numerical attributes, in addition to multi-word textual attributes, are considered to embed the watermark. Also, taking care of preserving the meaning of the watermarked text (by using the synonym substitution approach with a proper word sense disambiguation), the imperceptibility remains untouched, which results in a direct increment of the technique's robustness.

Notice that, when dealing with multi-word textual cover type, there are malicious operations focused on compromising the watermark embedded in textual documents (e.g., random synonym substitution attack) that must be considered to prove the effective robustness of our approach.

4.1. Architecture of the proposal

Fig. 1 depicts the architecture of the watermark embedding process of our proposal. As usual, the watermark extraction involves the same elements of the embedding procedure but it is performed in the opposite direction.

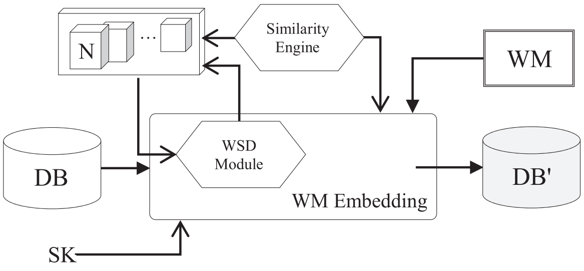


Fig. 1. Embedding process architecture.

Our watermark embedding procedure (fully described in Section 4.2) relies on a relational database DB storing one or several relations R with at least one multi-word textual attribute, one or several knowledge sources N, and a watermark WM. The choice of the knowledge source(s) is let free, but it has to take care of the semantic links between words. So that, on the latter, we can use a similarity engine to verify semantic consistency properties. A word sense disambiguation WSD module is needed for selecting the proper set of synonyms, depending on the context in which the words that are candidates to be replaced fall.

We also encourage the use of meaningful sources for the generation of the watermark considering that over this kind of signals can be applied methods to enhance the quality of the extracted watermark, contributing to its recognition despite the execution of aggressive attacks over the watermarked data. The secret key SK will be only known by the data owner and its complexity will be crucial against malicious operations (e.g., the brute force attack), as the security of watermarking techniques is based on the secrecy of the parameter values (Halder et al., 2010). Finally, a distorted database DB' is produced. Notice that, our watermarking technique modifies R only for the values that are selected for the embedding. Every other value remains the same.

4.1.1. Similarity engine

Consider a relation R(PK, A_1, …A_m) belonging to a database DB which stores at least one multi-word textual attribute. Precisely, let A_h (with h ∈ [1,m]) be a multi-word attribute in R, and let r_k be the k-th instance of R. Then, we refer to $r_k.A_h$ as the value of the attribute A_h with respect to the tuple r_k. Moreover, let s be a sentence in $r_k.A_h$, and assume that the word w part of s has been selected by a procedure $\mathscr{P}$ (see Algorithm 1 in Section 4.2.1) to embed the watermark, i.e., to be replaced with its synonym w' (see Algorithm 1 in Section 4.2.1). The embedding is performed if and only if the replacement complies the *intra-attribute consistency* and the *inter-attribute consistency* semantic properties.

Notice that, when it is possible, we replace only one word per sentence of a multi-word attribute value, and the semantic distortion can be embedded in more than one attribute per tuple.

Definition 1. (*intra-attribute consistency*) Let $r_k.A_h \in R$ be the value of the multi-word attribute A_h for the k-th instance of R. Let s be a sentence in $r_k.A_h$, w ∈ s be the word to replace, and let w' be the candidate substitute word. Then, s* denotes the sentence s where replacement has been applied, i.e., $s[w/w']$. Finally $r_k.A_h[s/s^*]$ denotes the distorted result. We say that $[w/w']$ is a substitution intra-attribute consistent if the semantic similarity score between $r_k.A_h$ and $r_k.A_h[s/s^*]$ is higher than or equal to a threshold δ. Formally:

$$ss(r_k.A_h, r_k.A_h[s/s^*]) \geqslant \delta$$

Definition 2. (*inter-attribute consistency*) Given r_k, i.e., the k-th instance of R, where watermark will be embedded, let r_k^* be the distorted result, i.e., $r_k[r_k.A_h/r_k.A_h[s/s^*]]$. Moreover, let ϕ be a function mapping tuple values to their concatenation (by means of “ and ”). Following Definition 1, we say that $[w/w']$ is a substitution inter-attribute consistent if the semantic similarity score between $\phi(r_k)$ and $\phi(r_k^*)$ is higher than or equal to a certain threshold μ. Formally:

$$ss(\phi(r_k), \phi(r_k^*)) \geqslant \mu.$$

The value of the thresholds δ and μ depends on the chosen semantic similarity measure.

4.1.2. The word sense disambiguation module

The correct functioning of the WSD module is a key element for the success of our approach. Two important issues depend on the WSD module performance: (i) maintaining the semantic value of the database being protected (ii) and the guarantee of achieving a high watermark synchronization.

[2] Two or more words are homographs if they are spelled the same way but differ in meaning and origin, and sometimes in pronunciation (Topkara et al., 2006).

WSD is considered an open research field in natural language processing. The main challenges come due to the fact that words often change meanings depending on the context they are used. For example, the noun *tree* can be used to refer to the programming data structure, but also to the living organism belonging to the vegetable kingdom. Thus, the set of synonyms allowed to replace *tree* must be shrunken according to certain context. The same happens with words used as adjectives. The adjective *hard* can be used to refer to someone with a sturdy temperament, to express determinism in business dealings, or to describe a feature of a solid object. If the ambiguity of the word is not taken away considering the context, there is a high probability the value of the text will be compromised once the word replacement is performed (Jalil and Mirza, 2009).

Let D be a function that returns the ordered set $\mathscr{Z}$ of synonyms of a word w given a context ς, denoted by $\mathscr{Z} \leftarrow D(\mathrm{w}, \varsigma)$. The following rules need to be accomplished:

1. $\mathrm{w} \in D(\mathrm{w}, \varsigma)$
2. $\forall \mathrm{t} \in D(\mathrm{w}, \varsigma) : D(\mathrm{t}, \varsigma) = D(\mathrm{w}, \varsigma)$

We denote by $\mathscr{Z}[\mathrm{t}]$ the t-th element of the set $\mathscr{Z}$.

For any word belonging to $\mathscr{Z}$, the set of synonyms for the given context must be the same to maintain the semantic value of the text from where w and ς were selected. Of course, if the WSD module does not work properly, these rules can be violated considering the same word can be part of different synonym sets given by other contexts (see Fig. 2 where the word "point" has multiple sets of synonyms relying on different contexts in which it can be used). In general, the correct functioning of the WSD module will depend on the accuracy of the implementation of the function D, responsible for obtaining the set of synonyms of w that more fit the context ς.

On the other hand, synchronization is the process of aligning two signals in time or space (Cox et al., 2007). Considering the embedded and extracted watermarks as those signals, achieving a high watermark synchronization relies on extracting the exact same marks that were embedded. If WSD fails, rules 1. and 2. are not accomplished. Then, for example, there is the possibility that a mark embedded considering the synonyms of the set $\mathscr{Z}_1$ is extracted looking at synonyms of the set $\mathscr{Z}_4$. Because of that, wrong mark values will be detected, compromising the quality of the extracted watermark and its synchronization. On the contrary, if previous rules are not violated, the value of the textual attribute being watermarked is preserved and the watermark synchronization is guaranteed.

4.1.3. *Data quality preservation*

The keeping of watermarked data quality will mainly depend on the WSD module and the parameters defining the maximum allowable semantic distortion to perform the marks embedding. According to Topkara et al. (2006), using ambiguous words increases the resilience of the watermarking scheme against attacks, but not all documents being watermarked tolerate this kind of operation, which reduces the options to perform word replacement.

On the other hand, as long as rules 1. and 2. are accomplished, the word used to replace w will be obtained from $\mathscr{Z}$, and the value of the mark detected during the extraction process will match the embedded one. Also, by defining the maximum tolerated semantic distortion, it is avoided the use of words belonging to $\mathscr{Z}$ that might involve ambiguity, increasing the probability of falling in another synonym set. Thus, the equivalency above guarantees the preservation of the data quality, no matter the nature of the text of R being protected.

4.2. *Watermarking procedure*

Distortion-based relational watermarking techniques consist of two processes: (i) the embedding of the watermark (ii) and its extraction (Halder et al., 2010). To achieve the watermark synchronization it is required the use of the same parameter values in both processes. Moreover, the extraction is performed when it is required to demonstrate the watermark presence in the data, as evidence in case of ownership claims, among others.

4.2.1. *Watermark embedding*

Algorithm 1 presents the details of the watermark embedding procedure of our approach. Given a relation R, for each tuple $\mathrm{r} \in \mathrm{R}$, the values of the multi-word textual attributes composing the list $\mathscr{A}$ are analyzed. Then, the virtual primary key k_r is generated using the VPK function (line 3). The input of the latter function results from the concatenation ($\circ$) between a secret key SK, and data represented by r_K identifying the tuple r (e.g., the relation's PK or other virtual primary keys generated by external schemes).

Following, in the case in which the `if-statement` condition is satisfied, a filter φ is applied (by the function Θ) to the multi-word attribute values in $\mathscr{A}$, in order to exclude those considering their content and links with the other attributes of the tuple (e.g., exclusion of sentences containing acronyms or abbreviations). In this way, we add an extra step to help maintain *inter-attribute consistency* while high unpredictability is incorporated into the technique [3]. Attributes passing the latter filter are stored in the set $\mathscr{A}'$ (lines 4–5).

Similarly as above, for each multi-word attribute value v in $\mathscr{A}'$, a filter χ is applied (by the function Λ) to the sentences in v, to exclude those that do not accomplish the conditions to be considered for the embedding process (e.g., involving just sentences composed of more than certain number of words). Sentences accomplishing the conditions defined in χ are finally considered for the embedding, and they are stored in the set $\mathscr{S}$ (lines 6–7).

For each sentence in $\mathscr{S}$ the key k_s is generated (from a one-way hash function H that takes as input the concatenation of SK, k_r and the elements of the sentence do not tolerating changes obtained by the Γ function) that identifies the sentence inside the multi-word attribute value, and using k_s the word w to be replaced is selected (lines 8–11). Notice that Υ is a function that returns, as an array of words, the elements of a sentence tolerating substitutions.

Algorithm 1: *watermarkingEmbedding* procedure.

```
1: **Input:** R, SK, $\gamma$, $\mathscr{A}$, WM, $\varphi$, $\chi$, N, $\mathfrak{C}$, $\delta$, $\mu$
2: **foreach** tuple $\mathrm{r} \in \mathrm{R}$
3:    $k_\mathrm{r} = \mathrm{VPK}(\mathrm{SK} \circ \mathrm{r_K})$
4:    **if** $(k_\mathrm{r} \bmod \gamma) = 0$
5:       $\mathscr{A}' \leftarrow \Theta(\mathscr{A}, \mathrm{r_K}, \varphi)$
6:       **foreach** $\mathrm{v} \in \mathscr{A}'$ **do**
7:          $\mathscr{S} \leftarrow \Lambda(v, \chi)$
8:          **foreach** sentence $\mathrm{s} \in \mathscr{S}$
9:             $k_\mathrm{s} = H(SK \circ k_\mathrm{r} \circ \Gamma(\mathrm{s}))$
10:              $i = k_\mathrm{s} \bmod \Upsilon(\mathrm{s}).\mathrm{length}$
11:              $\mathrm{w} = \Upsilon(\mathrm{s})[i]$
12:              $\varsigma \leftarrow$ *getSense*(w, s, N)
13:              $\mathscr{Z} \leftarrow$ *getCandidates*($\varsigma$, w, N)
14:              $\mathrm{w}' =$ *getSubstitute*(WM, $k_\mathrm{s}, \mathscr{Z}, \mathfrak{C}, \vartheta$)
15:              $\mathrm{s}^* \leftarrow \mathrm{s}[\mathrm{w}/\mathrm{w}']$
16:              $\mathrm{v}^* \leftarrow \mathrm{v}[\mathrm{s}/\mathrm{s}^*]$
17:              **if** $\mathrm{ss}(\mathrm{v}, \mathrm{v}^*) < \delta$
18:                 rollback embedding
19:              **else**
20:                 $\mathrm{r}^* \leftarrow \mathrm{r}[\mathrm{v}/\mathrm{v}^*]$
21:                 **if** $\mathrm{ss}(\phi(\mathrm{r}), \phi(\mathrm{r}^*)) \geqslant \mu$ **then**
22:                    $\mathrm{r} \leftarrow \mathrm{r}^*$
23:                    $\mathrm{v} \leftarrow \mathrm{v}^*$
24:                    **commit embedding**
25:                 **else**
26:                    rollback embedding
```

[3] Each attribute's value contributes itself in varying the elements of R selected as mark carriers.

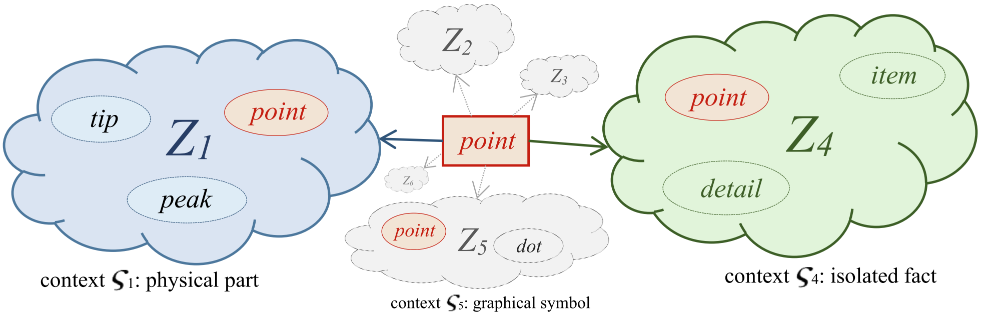

Z2
Z3
tip
point
Z1
peak
point
Z6
point
Z5
dot
point
item
Z4
detail
context ς1: physical part
context ς5: graphical symbol
context ς4: isolated fact

Fig. 2. Synonyms sets linked to a word w.

Then, according to the sense ς of the selected word in the sentence under consideration (obtained using the *getSense* method), the set $\mathscr{Z}$ of synonyms of w is obtained (by using the *getCandidates* method) (lines 12–13). Both *getSense* and *getCandidates* functions are based on the set of rules and the definitions given by the knowledge source(s) N.

Finally, the mark to be embedded and the new word w' to replace w are selected (line 14). Algorithm 2 defines the function *getSubstitute* where the mark is selected according to the value of k_s. The set of candidate substitute words is sorted according to the criteria $\mathfrak{C}$, and the new word is chosen depending on the value of the mark to be embedded. In lines 15–26, the replacement of the sentence in the carrier attribute and in the corresponding tuple is performed. The substitution will only be carried out if *intra-attribute* and *inter-attribute* consistencies properties are not violated.

Algorithm 2: *getSubstitute* procedure.

```
Input: WM, $k_s$, $\mathscr{Z}$, $\mathfrak{C}$, $\vartheta$
$i = k_s$ mod WM.length
$m \leftarrow$ WM[$i$]
set_order($\mathscr{Z}$, $\mathfrak{C}$)
if $m = 1$ then
    return $\mathscr{Z}[0]$
else
    return $\mathscr{Z}[\vartheta]$
```

On the other hand, if a word is not detected in the knowledge sources (i.e., *cross-linguality problem* (Agirre et al., 2009)), our approach ignores the position from the marking's candidates and proceed with the rest of the data stored in the relation.

4.2.2. Watermark extraction

The watermark extraction process is similar to the embedding process but is performed in the opposite direction. Also, it must be performed using the same parameter values employed to embed the watermark to guarantee its right synchronization. In the extraction process, for the same mark position in the watermark, several elements are recovered, and before assigning the mark final value, a majority voting is performed. In this way, the effect of attacks based on low aggressive data modifications are avoided.

The extraction is performed with no need to check the semantic distortion between the words replaced to carry out the marks embedding (i.e., without considering the similarity metrics value). Nevertheless, the extracted watermark quality depends on the knowledge source(s) and on the precision of the WSD module, since words can be assigned to a set of synonyms different from those considered for the embedding, adding noise to the extracted signal. This is because, in the new set of synonyms, the original word can occupy a different position, assigning to the extracted mark a wrong value.

4.3. Analysis of the watermark capacity

Since relational data have no fixed order, sequential watermarking approaches are vulnerable to subset reverse order attacks (Halder et al., 2010). To overcome this vulnerability, techniques have been designed performing a *pseudo-random selection* of both the watermark source elements used to generate the marks and the embedding locations in R. In general, this operation has been achieved by using a specific definition of Eq. (1). Nevertheless, besides contributing to robustness against subset reverse order attacks, *pseudo-random selection* makes it difficult for the attackers to predict embedding locations for overwriting or removing of marks.

$$\mathscr{V}(x,y)\ mod\ \mathscr{M}_X \tag{1}$$

where, $\mathscr{V}(x,y)$ is a value generated using data from a generic position (x,y) of R, and $\mathscr{M}_X$ is the maximum value of a given range. An example of a particular definition of this expression can be seen in line 10 of Algorithm 1, where $\mathscr{V}(x,y)$ is defined in line 9 as k_s.

Because of *pseudo-random selection*, during the embedding process some marks are selected more than once while others are entirely ignored. Embedding the same mark multiple times leads the technique to be resilient against update attacks, if a majority voting is performed over each mark position in the extraction process. On the other hand, if the number of excluded marks is too high, the watermark synchronization can be compromised. Indeed, the latter process would suffer from the same negative consequences as when aggressive attacks based on updating or deleting data are performed.

The *pseudo-random selection* downside can be reduced if the number of times the watermark source elements are considered increases. This can be achieved by marking a higher volume of data while the same watermark source is used.

In our approach, new elements to increase the watermark capacity, without affecting the imperceptibility requirement, are introduced. Moreover, we propose a new metric for measuring the watermark capacity which considers the different number of times each mark is selected during the embedding process. The latter metric allows the evaluation of the capacity in function of the technique's robustness, since it assigns to each mark a weight depending on the number of times it is embedded in R. In this way, the difficulty of compromising each mark in the watermarked data is reflected in the metric.

For techniques embedding one mark per selected tuple, the number of embedded marks E is equal to the number of marked tuples ω. If, besides the numeric cover type, multi-word textual attributes are considered (following the approach we proposed above), the number of embedded marks increases for each tuple according to Eq. (2), where λ_n represents the number of numeric attributes, λ_s the number of marked sentences, and $\eth$ the number of marks embedded in each sentence. If the watermark capacity increases due to marks embedded on multi-word textual attributes (besides those embedded on numerical attributes), using an effective WSD module, the technique becomes more resilient without compromising data usability.

$$E \approx \omega * (\lambda_n + \eth * \lambda_s) \tag{2}$$

Even if no numerical attributes are considered and only one multi-word textual attribute is selected to perform the embedding, more than one mark can be embedded per tuple, according to ð, which still constitutes an increment of the capacity compared to other watermarking techniques.

On the other hand, the number of marks selected for the embedding (denoted by m_e) using as reference the watermark size (denoted by n) is commonly used to measure the technique's watermark capacity. We define that metric as the binary capacity, given by c_b according to Eq. (3). The downside of c_b is that all marks present the same weight, and only their inclusion/exclusion represents information for the measurement.

$$c_b = m_e * 100/n \tag{3}$$

We consider the number of times each mark is embedded and we propose the weight-based capacity metric, given by c_w according to Eq. (4).

$$c_w = \sum_{i=0}^{n-1}\left(\varkappa\left(m_i\right)\right)/n \tag{4}$$

where, $\varkappa(m_i)$ represents the number of times the mark m_i was embedded. This is given since not all marks are selected the same number of times. In Fig. 3 we used a scale of colors to illustrate with an example the differences between c_b and c_w. The value of c_w corresponds to the mean of the number of times all marks of the watermark are embedded.

The higher c_w the better, but it is also required $\rho_w \approx 0$, being ρ_w the standard deviation of the number of times each mark is embedded. This tells us that each mark was selected multiple times evenly, adding higher relevance to the watermark recognition, increasing the probability of its detection despite the execution of benign updates and attacks over R.

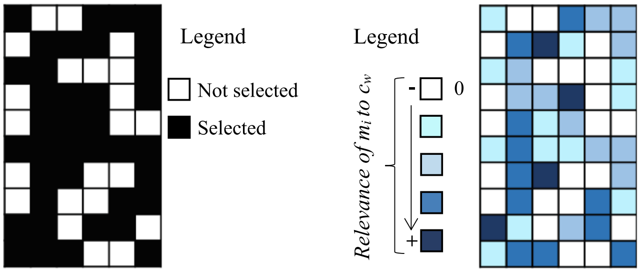
Notice that the highest value of c_w will be difficult to achieve as this would mean embedding each mark evenly the maximum possible number of times, and since the process presents *pseudo-random nature*, this is not expected.

In general, c_b does not reflect how embedding each mark multiple times contributes to obtain a different degree of resilience as c_w does. This is important to measure as when more redundancy is achieved for the embedding of a mark value, it is more difficult to compromise its value with update attacks, if a majority voting of all recovered values is performed during watermark extraction.

4.4. *Considerations for the adversary model*

One of the major challenges textual watermarking techniques based on synonyms substitution face is the *random synonyms substitution* attacks. This vulnerability is linked to the tolerance a text has with respect to synonyms substitution on it, depending on the context in which the data was used. Performing embedding of marks in multi-word textual attributes through semantically similar words replacement, we must consider adding resilience to this threat, as well as to the rest of the malicious operations an attacker may perform over the database relation.

Depending on the context in which data are used, we have a margin of marks allowed to be embedded. If the number of marks that can be embedded is high, an high degree of robustness can be achieved, otherwise, the technique's resilience gets compromised. Moreover, the data context also reduces the attacker's freedom to perform aggressive operations if he wants to preserve the quality of the data. Therefore, using textual watermarking to relational data increases the difficulty to perform effective attacks.



a) Binary criterion (c_b) b) Weight-based criterion (c_w)

Fig. 3. Criteria to evaluate the watermark capacity.

Increasing the capacity by considering both, the numerical and textual cover types, allows achieving higher robustness, making hard to compromise the watermark detection. Several elements of the relation, such as attributes storing float numbers, single sentences or several paragraphs, can be selected to embed the marks, based on multiple features (e.g., numerical ranges, minimum number of nouns required per sentence, among others) which highly increase the complexity to detect the marks embedding locations. Furthermore, beyond the complexity of the *pseudo-random selection* of the marks embedding places in the relation, we take advantage of the multi-word textual data structure to add a high entropy to the embedding procedures.

Agrawal and Kiernan (2002) defined Eq. (5) to get the probability for the attacker to successfully detect the embedding locations used by the data owner. While ω refers to the number of tuples watermarked by the data owner, γ_A, ν_A, and ξ_A denote respectively the tuple fraction, the number of attributes, and the *lsb* considered by the attacker.

$$P\left\{success|\omega\right\} = \left(1 - \frac{1}{2\gamma_A\nu_A\xi_A}\right)^{\omega} \tag{5}$$

Considering all those elements, it is difficult to detect the marks embedding positions used by the data owner. Despite that, we increase the difficulty by adding the element ζ_A to Eq. (5). Eq. (6) extends Eq. (5) as follows:

$$P\left\{success|\omega\right\} = \left(1 - \frac{1}{2\gamma_A\nu_A(\xi_A + \zeta_A)}\right)^{\omega} \tag{6}$$

The term ζ_A defines the complexity that derives from the consideration of marks embedding locations among multi-word textual cover types. It expresses the possibility of embedding one mark in the whole attribute value, or marking each sentence with one mark, or embedding multiple marks in each sentence. Since all embedding positions for textual attributes are also generated using *pseudo-random selection* and considering the increment of the number of elements to know by the attacker, the probability of performing successful attacks reduces. Furthermore, by accomplishing security and public system requirements (Agrawal et al., 2003), we add secrecy to parameters' values, increasing the difficulty of attackers to detect embedding locations.

5. Experimental results

Following the recommendation given in Section 4.1 of considering meaningful sources to generate the watermark WM, we validated our approach by using binary images as WM sources. Besides the benefits previously mentioned, this type of data allows taking the simplest pixel value for the mark generation, which contributes to perform less aggressive distortions during the watermark embedding compared to techniques generating marks from color (e.g., Zhang et al. (2005)) or gray-scale images (e.g., Zhang et al. (2004)).

To analyze the effect of the watermark length variation, images of

a) Universiti Teknologi Malaysia logo b) World Wildlife Fund Logo c) Chinese character "dào"

Fig. 4. Samples of the binary images used as WM sources.

different sizes were used. Samples of them are shown in Fig. 4: (a) the Universiti Teknologi Malaysia (UTM) logo (82 × 80 pixels), (b) the logo of the World Wildlife Fund (WWF) (40 × 45 pixels), and (c) the Chinese character dào (20 × 21 pixels). By convention, we used the red color to highlight the missed pixels due to watermark incomplete embedding or malicious operations by attackers.

To know the quality of the extracted WM two metrics were used: the Correction Factor (CF) and the Structural Similarity Index (SSIM). The Correction Factor, defined by Eq. (7), is used to compare the pixels of the image generated from the embedded WM (given by Img_{emb}) against the pixels of the image generated from the extracted WM (given by Img_{ext}). In the equation, variables h and w represent the height and the width of the images respectively. The maximum value of CF is 100, meaning that the WMs are identical. In the case in which CF = 0, then the embedded and the extracted WMs are completely different.

$$CF = \frac{\sum_{i=1}^{h}\sum_{j=1}^{w}\left(Img_{emb}\left(i,j\right) \oplus \overline{Img_{ext}(i,j)}\right)}{h \times w} \times 100 \tag{7}$$

The SSIM is oriented to obtain an appreciation of the image's quality closer to human perception. The index is calculated according to Eq. (8), using multiple windows. The windows are defined by x and y and present common size $N \times N$. The range of possible values taken by this metric in this work is between 0 and 1, where 1 means there exists a perfect structural similarity between the embedded and the extracted image, and 0 indicates no structural similarity.

$$SSIM\left(x,y\right) = \frac{(2\mu_x\mu_y + C_1) + (2\sigma_{xy} + C_2)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)} \tag{8}$$

The symbols μ_x and μ_y represent the average of x and y respectively, σ_x^2 and σ_y^2 their respective variance, and σ_{xy} their covariance. The elements C_1 and C_2 are two stabilization constants.

The data to embed the marks was the data set *Amazon Fine Food Reviews*. The structure is depicted in Table 3, from where we mostly used the attribute '*Text*', also storing the text with the highest length. We also used the first 30.000 tuples out of 500,000 to compare our results with previous works.

We used the relation's PK to perform the watermark synchronization. To avoid the use of the PK we recommend the generation of virtual primary keys by using the Ext-Scheme (Gort et al., 2017) or the HQR-Scheme (Gort et al., 2019). These schemes were originally proposed for numerical attributes, but they can be applied by combining numerical and textual cover types as well.

On the other hand, as knowledge source, we used WordNet (Miller, 1998), which consists of a lexical database of the English language, where nouns, verbs, adjectives, and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept (Princeton-University, 2010). So, given a word w, and the context where w is used (i.e., the meaning of the sentence to which w is part of), WordNet returns the appropriate synset of w. Notice that, in our experiment, the set of synonyms $\mathscr{Z}$ (introduced in Section 4) will correspond to a synset in WordNet.

Table 3
Structure of the dataset "Amazon Fine Food Reviews".

Attribute	Type	Description
ProductId	String	Id. of the product
UserId	String	Id of the user
ProfileName	String	Name of the user
HelpfulnessNumerator	Numeric	Numerator of the fraction of users who found the review helpful
HelpfulnessDenominator	Numeric	Denominator of the fraction of users who found the review helpful
Score	Numeric	Rating of the product
Time	Numeric	Time of the review (unix time)
Summary	String	Review summary
Text	String	Text of the review

For the evaluation of our approach, we implemented a client–server architecture application using Java 1.8 programming language for the client-side, the Oracle Database 12c for the server-side. We used WordNet 3.1 database files with the Java API jwnl 1.4.1 rc2 for accessing and working with WordNet resources and ws4j 1.0.1 for using Semantic Relatedness/Similarity algorithms already developed. The WSD module was based on the Lesk algorithm (Vasilescu et al., 2004), which compares the word definitions with the definitions of the rest of the words presented in the sentence, finding the more convenient context for its use. According to that, the most appropriate set of synonyms can be chosen. Finally, the runtime environment was a 3.60 GHz Intel i7-4790 PC with 16.0 GB of RAM running on Windows 10 OS.

5.1. Improvement of the watermark capacity

We performed the watermark capacity analysis comparing our approach with two other techniques, Sardroudi and Ibrahim (2010), which uses only one attribute, and Pérez Gort et al. (2017) with two attributes. Of all techniques using an image to generate the watermark (i.e., Image-Based Watermarking (IBW) (Halder et al., 2010)) these techniques constitute a representation of the ones more recent, used to mark one or several attributes per tuple. We selected only one attribute to mark with our approach to show WM capacity improves even compared to two-attributes embedding, thanks to the selected cover type. The techniques we compare with performed the watermark embedding on the numerical cover type, but by involving the same number of tuples we can appreciate how much the watermark capacity increases for our approach.

Table 4 shows the value of c_b (see Eq. (3) Section 4.3) obtained for each technique. In the table, columns titles "*S & H*" refers to Sardroudi and Ibrahim (2010)'s technique, "*G. et al.*" to Pérez Gort et al. (2017)'s and "*Prop.*" to our proposal. Given that, the number of marks missed by using our approach is lower than the number of marks missed by using the other techniques. Indeed, there are fewer red pixels in the images of the WMs synchronized by our proposal. The main reason for WM improvement is because of for some cases the values stored in the attribute *"Text"* are composed of more than one sentence. If allowed, we only embed one mark per sentence selecting a common *nouns* from it. By more than one mark can be used as the carrier, in which case WM capacity will be even higher.

Table 5 shows the value of c_w (see Eq. (4) Section 4.3) with the correspondent ρ_w for each case, giving a clear idea about how we not only improve the value for c_b but for c_w as well, increasing the probability of overcoming attacks based on data updates.

The experiments to register the capacity values were applied over a set of 30000 tuples. For the case of Sardroudi & Ibrahim's and Pérez Gort et al.'s, it was used the numerical data set *Forest Cover Type* (Colorado-State-University, 1999) as these techniques were designed for marking numerical values. Also, the watermark embedding with Pérez Gort et al.'s technique was performed with Attribute Fraction equal to 5 in order to mark only two attributes per tuple.

Once WM capacity increase was proven, it is critical to guarantee marks detection, otherwise, it will not be possible to recognize the WM signal in the protected data. In the following, the results focused on testing the WM detectability are shown. It is also analyzed the way the WSD module precision determines the quality of the extracted WM.

5.2. Detectability Analysis

The detectability of marks in our approach is directly linked to the precision of the WSD module. Since for WM embedding are used set of synonyms of the selected word, it is not expected data quality degradation, but if for WM extraction, the WSD module does not assign the same set of synonyms used for the embedding, then several marks will be

Table 4
Value of c_b for different techniques.

c

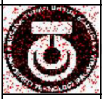
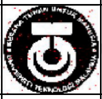
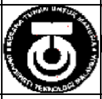
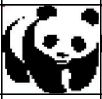
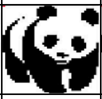
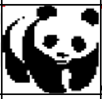
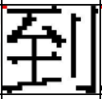
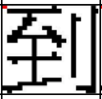
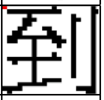
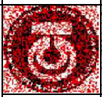
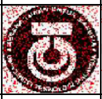
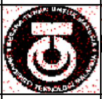
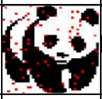
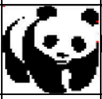
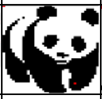
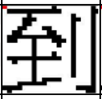
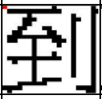
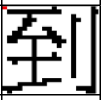
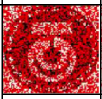
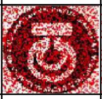
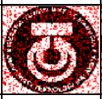
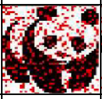
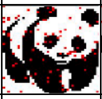
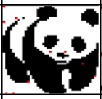
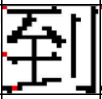
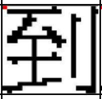
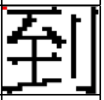
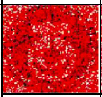
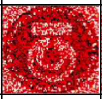
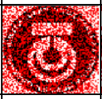
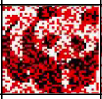
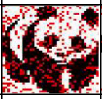
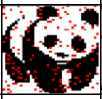
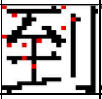
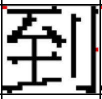
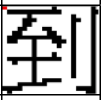
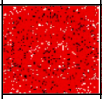
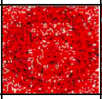
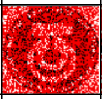
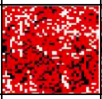
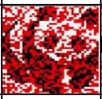
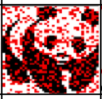
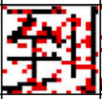
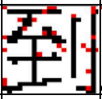
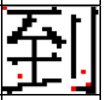
TF	UTM			WWF			Dào		
	S & H	G et al.	Prop.	S & H	G et al.	Prop.	S & H	G et al.	Prop.
2									
	88.34%	98.61%	99.91%	99.94%	99.94%	99.94%	99.76%	99.76%	99.76%
5									
	57.93%	81.46%	93.67%	95.16%	99.72%	99.89%	99.76%	99.76%	99.76%
10									
	35.64%	56.99%	75.35%	78.77%	94.88%	99.44%	99.29%	99.76%	99.76%
20									
	19.25%	34.28%	48.96%	53.88%	78.66%	91.83%	97.38%	99.52%	99.76%
40									
	9.66%	18.14%	27.99%	31.27%	51.88%	69.56%	80.95%	94.04%	98.81%

Table 5
Value of c_w with its correspondent ρ_w for each experiment.

TF	UTM			WWF			Dào		
	S & H	G et al.	Proposal	S & H	G et al.	Proposal	S & H	G et al.	Proposal
2	2 (±1.47)	4 (±2.12)	6 (±2.73)	7 (±2.95)	15 (±4.20)	24 (±5.12)	33 (±5.92)	67 (±8.96)	105 (±11.18)
5	0 (±1.27)	1 (±1.51)	2 (±1.87)	3 (±1.81)	6 (±2.56)	10 (±3.27)	13 (±3.65)	26 (±5.46)	43 (±6.79)
10	0 (±0.80)	0 (±1.26)	1 (±1.23)	1 (±1.43)	3 (±1.79)	5 (±2.28)	6 (±2.72)	13 (±3.73)	21 (±4.67)
20	0 (±0.51)	0 (±0.78)	0 (±1.07)	0 (±1.18)	1 (±1.36)	2 (±1.64)	3 (±1.86)	6 (±2.72)	10 (±3.20)
40	0 (±0.34)	0 (±0.50)	0 (±0.66)	0 (±0.72)	0 (±1.16)	1 (±1.12)	1 (±1.37)	3 (±1.89)	5 (±2.33)

recovered with wrong values, adding noise to the extracted WM signal. If the signal is too noisy, WM synchronization can be compromised, making impossible its identification.

The main goal of this work is not to improve WSD algorithms, but to use those already defined that guarantee high precision for marks detection. The WSD module precision will be denoted as $\mathscr{P}$, which is obtained according to Eq. (9), where $\mathscr{W}_T$ represents the number of tagged words (i.e., words selected for sense disambiguation) and $\mathscr{W}_{CT}$ the number of words correctly tagged (i.e., words that during the extraction process were linked to the same synonym set used for WM embedding).

$$\mathscr{P} = \mathscr{W}_{CT}/\mathscr{W}_T \tag{9}$$

The WSD module we used is based on the Lesk algorithm (Vasilescu et al., 2004). The precision described by this module was registered through a set of experiments, whose results are shown in Table 6.

Table 6
WSD precision during WM detection.

TF	UTM	WWF	Dào
2	0.9479	0.9498	0.9518
5	0.9478	0.9500	0.9517
10	0.9518	0.9513	0.9516
20	0.9482	0.9422	0.9466
40	0.9532	0.9472	0.9463

Notice that when combining WSD along with the watermarking technique, WSD lack of precision can be compensated by the majority voting performed over WM extraction. Then, the higher c_w with ρ_w closer to zero, the stronger the effect of the majority voting to overcoming low WSD precision. Table 7 shows the results experimentally supporting this point, linking the quality of the detected WM to the results of Table 5 despite the precision weaknesses shown in Table 6.

The results shown in Table 7 correspond to a set of experiments with different values for TF and WM sources. For each case, it is shown the embedded WM (the small image), the detected one (the big image), and the value of SSIM with a percentage corresponding to the number of pixels not matching between the two images. The low value of this metric compared to those shown in Table 6 directly endorses our statement.

Previous results show how wrong mark values are not reflected in red pixels, but in black and white, added to wrong regions of the image. This can be understood as a *Gaussian noise* which degree is directly linked to the precision of the WSD module. In Table 8 are shown other results, which describe the quality of the extracted WM in more detail. This time, the quality of the detected WM is given respect to both, the embedded WM and the original one. Of course, since depending on the parameter's values the original WM is not usually entirely embedded, the higher quality will be the one given respect to the embedded WM.

Data detectability is benefited from the combination between the WSD module, which takes care of imperceptibility, and majority voting,

Table 7
Quality of the detected WM respect to the embedded one.

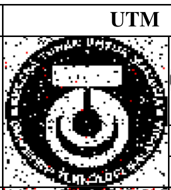
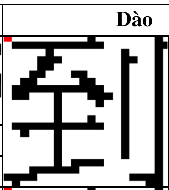
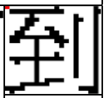
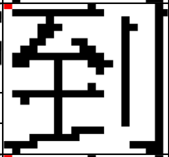
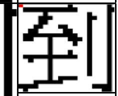
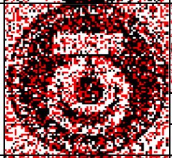
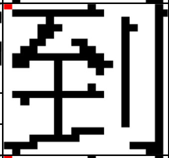
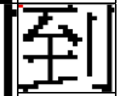
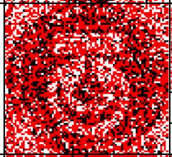
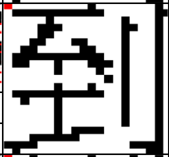
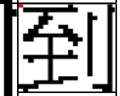
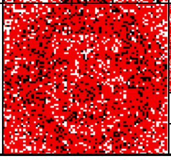
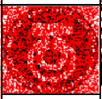
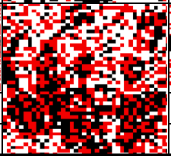
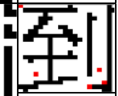
TF	UTM		WWF		Dào	
2		 0.69 *4.92%*		 1.00 *0*		 1.00 *0*
5		 0.55 *18.00%*		 0.94 *1.94%*		 1.00 *0*
10		 0.56 *24.56%*		 0.75 *8.33%*		 1.00 *0*
20		 0.58 *21.55%*		 0.61 *19.89%*		 1.00 *0.48%*
40		 0.63 *13.70%*		 0.62 *23.66%*		 0.84 *7.38%*

Table 8
Detectability achieved for each WM over different numbers of marked tuples.

TF	UTM				WWF				Dào			
	vs. Embedded		vs. Original		vs. Embedded		vs. Original		vs. Embedded		vs. Original	
	CF	SSIM	CF	SSIM	CF	SSIM	CF	SSIM	CF	SSIM	CF	SSIM
2	99.42%	0.69	95.06%	0.68	100%	1	99.94%	0.99	100%	1	99.76%	0.99
5	88.34%	0.55	80.06%	0.49	100%	0.94	98.00%	0.93	100%	1	99.76%	0.99
10	71.11%	0.56	61.32%	0.37	98.77%	0.75	91.72%	0.75	100%	1	99.76%	0.99
20	54.42%	0.58	37.85%	0.25	83.54%	0.61	76.66%	0.54	100%	1	99.28%	0.99
40	46.51%	0.63	21.9%	0.17	67.41%	0.62	55.33%	0.41	98.07%	0.84	91.9%	0.83

which allows being tolerated WSD lack of precision. The positive impact of this effect increase when WM size decreases or the number of tuples being watermarked increases. The following experiments are meant to analyze how data usability and watermark imperceptibility are maintained. This two WM requirements are critical, since if are not accomplished, the technique becomes useless for practical scenarios.

5.3. Watermark imperceptibility

As it was mentioned before, the embedding process is carried out through the replacement of a *pseudo-randomly* selected word by a synonym from a specific set of synonyms. The word selected from the latter set will depend on the value of the mark extracted from the binary image. Occasionally, the word selected from the set of synonyms is the same one that was selected from the sentence. That is the scenario that best contributes to WM imperceptibility since the mark is embedded without any modification in the data. We compute the rate of marks embedded without performing word replacement through the rate of fixed words given by $T_w = \mathscr{W}_F/\mathscr{W}_T$, where $\mathscr{W}_F$ represents the words that do not change during the embedding process and $\mathscr{W}_T$, as previously defined, represents the number of tagged words. The value of T_w for each one of the experiments is shown in Table 9, where more or less for all cases a third of the selected words allows mark embedding without being replaced, which positively contributes to achieving WM imperceptibility.

From the knowledge source point of view, it is also possible to detect the quality of the WM imperceptibility. For this case, since we are using WordNet, we can use a set of similarity metrics that are defined to measure the relatedness or similarity between words. According to WordNet structure, and the way the metrics are defined, when two words are selected from the same synonym set, the metrics report the

Table 9
Value of T_w for previous experiments.

TF	UTM	WWF	Dào
2	0.3752	0.3655	0.3146
5	0.3765	0.3652	0.3142
10	0.3763	0.3597	0.3153
20	0.3768	0.3591	0.3094
40	0.3961	0.3590	0.3068

Table 10
Similarities metrics for WM UTM.

TF	Iter.	WUP	JCN	LCH	LIN	RES	PATH	LESK
2	44510	44.64×10^3	4.73×10^{11}	16.40×10^4	44.45×10^3	38.02×10^4	44.46×10^3	44.22×10^4
5	18321	18.39×10^3	1.96×10^{11}	67.55×10^3	18.31×10^3	15.71×10^4	18.31×10^3	18.17×10^4
10	9091	91.20×10^2	9.72×10^{10}	33.52×10^3	90.84×10^2	78.00×10^3	90.85×10^2	89.75×10^3
20	4427	44.37×10^2	4.73×10^{10}	16.32×10^3	44.22×10^2	38.00×10^3	44.23×10^2	43.05×10^3
40	2156	21.63×10^2	2.31×10^{10}	79.53×10^2	21.55×10^2	18.65×10^3	21.56×10^2	21.45×10^3

highest possible value between them. Table 10 gives the accumulated value for some metrics commonly used, for the experiments performed during mark embedding using UTM as WM source. The table's column "Iter." refers to iterations, meaning the number of times the measurement between words was carried out.

Thanks to the use of similarity metrics it is possible to determine and control the amount of distortion introduced during WM embedding. This contributes to maintaining data usability and WM imperceptibility, goals that highly depend on the knowledge source, the similarity engine and the WSD module. For our case, as long as the words belong to the same set of synonyms, quality results are guaranteed.

Even though, since our technique is meant to be used for relational data copyright protection, we have to guarantee robustness against malicious operations. The core of our approach has been proved to be resilient against common malicious operations oriented to compromise relational data watermarking techniques (Agrawal and Kiernan, 2002; Pérez Gort et al., 2017; Sardroudi and Ibrahim, 2010). Nevertheless, we need to consider another threat more focused on compromising WM detection over text documents. The next section is oriented to analyze the resilience of our technique.

5.4. Technique's robustness

The major threat our technique faces is linked to *random synonym substitution* attacks from watermarking techniques created for document protection. In a similar way that WSD and majority voting combined allow overcoming WSD lack of precision, using textual as WM cover type in relational data reduces the probability of performing a successful *random synonym substitution* attack. This is because to successfully compromise the mark value, the right words need to be selected. But first, it is required to detect the right tuple, and the attribute selected for marking inside the tuple. The high number of parameters involved in the technique makes that very difficult to achieve.

This is one of the benefits of combining both cover types. The probability of successfully overwriting marks decreases if, besides relational elements, textual's are considered. To that, once the position is correctly detected in the relation, it is necessary to know the type of word selected for the embedding (e.g., noun, adverbs, adjectives, etc.), the sentence itself, and break the secrecy of k_s (see Algorithm 1). As we mentioned in subSection 4.4, this is ruled by the adversary model obtained as a result of extending Eq. (5) to Eq. (6).

To study our approach's resilience to *random updates* we performed two types of experiments, random tuple deletion and random actualization of words stored in the same attribute we use for marking. In Table 11 is depicted the degree of damage the WM gets while the number of pseudo-randomly deleted tuples increases. This experiment was performed attacking the relation marked with the WM generated from the image WWF with parameters TF = 2, detected with quality of SSIM = 1 with no pixels in contradiction with respect to the embedded WM. (see Table 7).

The second robustness experiment was performed under the same conditions of the previous one, but updating the value of the attribute, randomly selecting the tuple according to the attack's percentage. In this case, the update operation is based on selecting the same attribute value but replacing a word from the sentence for a synonym. The selection of the tuple, the word being replaced, and the synonym was made pseudo-randomly. This experiment was performed this way to simulate random synonym substitution attacks focused on compromising textual

Table 11
WMs detected after pseudo-random tuple deletion attacks.

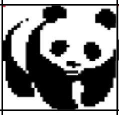
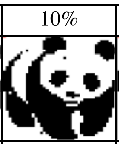
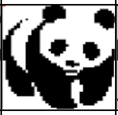
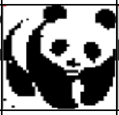
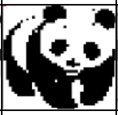
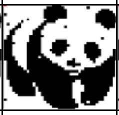
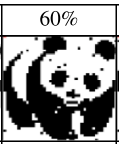
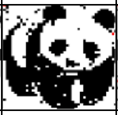
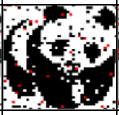
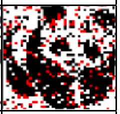
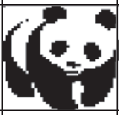
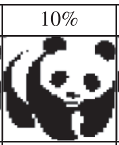
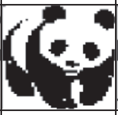
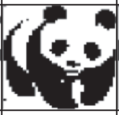
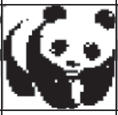
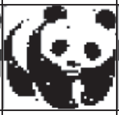
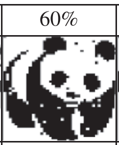
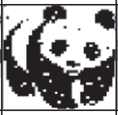
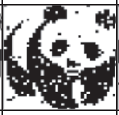
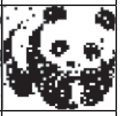
Datum	Percentage of tuples deleted (attack degree).									
	0	10%	20%	30%	40%	50%	60%	70%	80%	90%
Image										
SSIM	1	0.99	1	0.99	0.94	0.93	0.90	0.82	0.74	0.55
CF	100%	100%	100%	100%	100%	100%	100%	99.88%	97.10%	78.65%

Table 12
WMs detected after pseudo-random update attacks.

Datum	Percentage of the attributes updated (attack degree)									
	0	10%	20%	30%	40%	50%	60%	70%	80%	90%
Image										
SSIM	1	1	0.99	0.99	0.97	0.92	0.89	0.89	0.78	0.78
CF	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

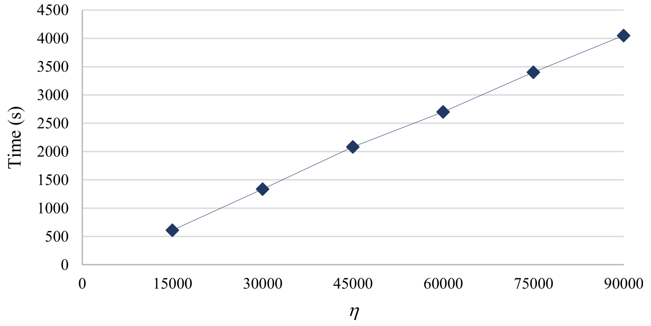


Fig. 5. Correlation described by the times required for watermarking different amount of tuples.

Table 13
Time required for WM embedding involving different tuples number.

η	Time (s)	
	Average	Proportion
15000	609.75 (±2.70)	none
30000	1336.00 (±7.35)	2.19
45000	2081.57 (±3.40)	1.55
60000	2699.07 (±7.90)	1.30
75000	3401.25 (±14.49)	1.26
90000	4050.67 (±27.41)	1.19

watermarking. If the mark value during the WM detection is given by exclusion (assigning 1 if the word is the first one in the synonym set and 0 if it is not), the probability of success for this type of attack decreases considerably.

From the experiments performed in this section, it is clear our technique describes a resilience that guarantees WM preservation despite data degradation due to malicious operations (see Tables 11 and 12). Considering attackers are also interested in maintaining data quality, it is not expected they will exceed the degree of damage caused to the data in the experiments we performed. Then, we claim our technique is resilient, being recommended for practical scenarios to guarantee copyright protection.

5.5. *Scalability and complexity*

Since WM embedding and extraction processes are similar in complexity, in this section we report the time required for WM embedding. We carried out a set of experiments, performing WM embedding multiple times involving a different number of tuples. Fig. 5 depicts the linear correlation between the time required by WM embedding and the number of tuples in R. This experiment was performed using $\gamma = 5$, the WM source WWF and the attribute “Text”, increasing the value of η of 15000 units each time.

For each amount of tuples, the same experiment was performed several times until reaching a standard deviation of the time required as close as zero as possible. Table 13 shows along with the standard deviation, the mean of the time recorded. According to column “Proportion”, which compares the average of the time required for marking the tuples with respect to the average of time required for the previous row, our approach describes a linear behavior.

The approach’s complexity will depend on the Similarity Engine, the WSD module, and the number of sentences stored in the attribute being watermarked. Despite all these factors, for the conditions given by the experimental set up to validate our work, it is recorded a linear behavior. Then, it can be established that the overall time complexity corresponds to $O(n)$, being reliable its application to different sizes of data stored in R.

6. Conclusions

In this paper, we proposed a watermarking technique for relational data that uses multi-word textual attributes as cover type. The embedding of the marks in our approach is performed by substitutions of synonym words in sentences, guaranteeing the semantic preservation of the data, and the total imperceptibility of the watermark. Despite multiple attributes can be considered for each tuple, when paragraphs are stored, the selection of one word per sentence allows the increment of the watermark capacity with respect to previous techniques, and with it, its robustness.

This technique works in combination with a WSD module, a semantic similarity engine, and one or several knowledge sources, linking its complexity and precision to the behavior of those external elements. For the experimental validation we used WordNet as knowledge source and the Lesk algorithm for WSD. The results show that how our technique guarantees the watermark embedding, its detection, and robustness against subset attacks and random synonym substitution attacks, making it a valuable tool for ownership protection, and the data integrity validation. For the case of random synonym substitution attacks, which constitute a serious threat for techniques focused on watermarking textual documents, the combination of the relational data structure and the multi-word textual data type guarantees the watermark persistence independently the attack’s degree. The preservation of the semantic was defined as the main priority for this approach, adding as a feature the tolerance to the watermark embedding in a way no other technique has considered before, according to the literature published so far.

As future work we aim to design a module of semantic preservation for numerical and textual data, considering the elements of watermarking techniques and approximate query processing.

CRediT authorship contribution statement

Maikel Lázaro Pérez Gort: Writing - original draft, Writing - review & editing, Data curation, Investigation, Software, Methodology, Validation. **Martina Olliaro:** Writing - original draft, Writing - review & editing, Data curation, Investigation, Methodology. **Agostino Cortesi:** Methodology, Supervision, Validation. **Claudia Feregrino Uribe:** Conceptualization, Supervision, Writing - review & editing, Resources.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was partially supported by a Ph.D. Grant No. 714270 and project Grant No. PN 2017–01-7092 from CONACyT, Mexico.

References

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., & Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, NAACL '09*.

Agrawal, R., Haas, P. J., & Kiernan, J. (2003). Watermarking relational data: Framework, algorithms and analysis. *The VLDB Journal-The International Journal on Very Large Data Bases, 12*(2), 157–169.

Agrawal, R., & Kiernan, J. (2002). Watermarking relational databases. In *VLDB'02: Proceedings of the 28th International Conference on Very Large Databases* (pp. 155–166). Elsevier.

Al-Haj, A., & Odeh, A. (2008). Robust and Blind Watermarking of Relational Database Systems. *Journal of Computer Science, 12*, 1024–1029.

Batet, M., & Sánchez, D. (2015). A review on semantic similarity. In *Encyclopedia of information science and technology* (3rd ed., pp. 7575–7583). IGI Global.

Bertino, E., Ooi, B.C., Yang, Y., and Deng, R.H. (2005). Privacy and ownership preserving of outsourced medical data. In null, pages 521–532. IEEE.

Bhattacharya, S. and Cortesi, A. (2009b). A generic distortion free watermarking technique for relational databases. In Information Systems Security, 5th International Conference, ICISS 2009, Kolkata, India, December 14–18, 2009, Proceedings, volume 5905 of Lecture Notes in Computer Science, pages 252–264. Springer.

Bhattacharya, S., & Cortesi, A. (2009a). A distortion free watermark framework for relational databases. In *ICSOFT 2009 – Proceedings of the 4th International Conference on Software and Data Technologies, Volume 2 July 2009, 26–29, Sofia, Bulgaria* (pp. 229–234).

Chang, C.-C., Nguyen, T.-S., & Lin, C.-C. (2014). A blind robust reversible watermark scheme for textual relational databases with virtual primary key. In *International Workshop on Digital Watermarking* (pp. 75–89). Springer.

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM, 13*(6), 377–387.

Colorado-State-University (1999). Forest CoverType, The UCI KDD Archive.

Cox, I., Miller, M., Bloom, J., Fridrich, J., & Kalker, T. (2007). *Digital watermarking and steganography*. Morgan Kaufmann.

Farfoura, M. E., Horng, S.-J., Lai, J.-L., Run, R.-S., Chen, R.-J., & Khan, M. K. (2012). A blind reversible method for watermarking relational databases based on a time-stamping protocol. *Expert Systems with Applications, 39*(3), 3185–3196.

Franco-Contreras, J., & Coatrieux, G. (2015). Robust watermarking of relational databases with ontology-guided distortion control. *IEEE Transactions on Information Forensics and Security, 10*(9), 1939–1952.

Franco-Contreras, J., Coatrieux, G., Cuppens-Boulahia, N., Cuppens, F., & Roux, C. (2014). Ontology-guided distortion control for robust-lossless database watermarking: Application to inpatient hospital stay records. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (pp. 4491–4494). IEEE.

Gort, M. L. P., Díaz, E. A., & Uribe, C. F. (2017). A highly-reliable virtual primary key scheme for relational database watermarking techniques. In *2017 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 55–60). IEEE.

Gort, M. L. P., Feregrino-Uribe, C., Cortesi, A., & Fernández-Peña, F. (2019). HQR-scheme: A high quality and resilient virtual primary key generation approach for watermarking relational data. *Expert Systems with Applications, 138*, Article 112770 .

Guo, J. (2011). Fragile watermarking scheme for tamper detection of relational database. In *2011 International Conference on Computer and Management (CAMAN)* (pp. 1–4). IEEE.

Halder, R., Pal, S., & Cortesi, A. (2010). Watermarking techniques for relational databases: Survey, classification and comparison. *Journal of Universal Computer Science, 16*(21), 3164–3190.

Hliaoutakis, A., Varelas, G., Voutsakis, E., Petrakis, E. G. M., & Milios, E. E. (2006). Information retrieval by semantic similarity. *International Journal on Semantic Web and Information Systems, 2*(3), 55–73.

Jalil, Z., & Mirza, A. M. (2009). A review of digital watermarking techniques for text documents. In *2009 International Conference on Information and Multimedia Technology* (pp. 230–234). IEEE.

Jiang, C., Chen, X., & Li, Z. (2009). Watermarking relational databases for ownership protection based on DWT. In *2009 Fifth International Conference on Information Assurance and Security* (Vol. 1, pp. 305–308). IEEE.

Kamaruddin, N. S., Kamsin, A., Por, L. Y., & Rahman, H. (2018). A review of text watermarking: Theory, methods, and applications. *IEEE Access, 6*, 8011–8028.

Kamran, M., & Farooq, M. (2013). A formal usability constraints model for watermarking of outsourced datasets. *IEEE Transactions on Information Forensics and Security, 8*(6), 1061–1072.

Mehta, B. B., & Aswar, H. D. (2014). Watermarking for security in database: A review. In *IT in Business, Industry and Government (CSIBIG), 2014 Conference on* (pp. 1–6). IEEE.

Melkundi, S., & Chandankhede, C. (2015). A Robust Technique for Relational Database Watermarking and Verification. In *Communication, Information & Computing Technology (ICCICT), 2015 International Conference on* (pp. 1–7). IEEE.

Miller, G. A. (1998). *WordNet: An electronic lexical database*. MIT Press.

Pérez Gort, M. L., Feregrino Uribe, C., and Nummenmaa, J. (2017). A minimum distortion: High capacity watermarking technique for relational data. In Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security. pp. 111–121. ACM.

Petrakis, E. G. M., Varelas, G., Hliaoutakis, A., & Raftopoulou, P. (2006). X-similarity: Computing semantic similarity between concepts from different ontologies. *JDIM, 4* (4), 233–237.

Princeton-University. (2010). *About WordNet*. WordNet: Princeton University.

Sardroudi, H. M., & Ibrahim, S. (2010). A new approach for relational database watermarking using image. In *5th International Conference on Computer Sciences and Convergence Information Technology* (pp. 606–610). IEEE.

Seco, N., Veale, T., and Hayes, J. (2004). An intrinsic information content metric for semantic similarity in WordNet. In Proceedings of the 16th European Conference on Artificial Intelligence, ECAI'2004, including Prestigious Applicants of Intelligent Systems, PAIS 2004, Valencia, Spain, August 22–27, 2004. pp. 1089–1090.

Slimani, T. (2013). Description and evaluation of semantic similarity measures approaches. *International Journal of Computer Applications, 80*(10), 25–33.

Sun, J., Cao, Z., and Hu, Z. (2008). Multiple watermarking relational databases using image. In MultiMedia and Information Technology, 2008. MMIT'08. International Conference on. pp. 373–376. IEEE.

Taieb, M. A. H., Aouicha, M. B., & Hamadou, A. B. (2014). A new semantic relatedness measurement using WordNet features. *Knowledge and Information Systems, 41*(2), 467–497.

Taleby Ahvanooey, M., Li, Q., Shim, H. J., & Huang, Y. (2018). A comparative analysis of information hiding techniques for copyright protection of text documents. *Security and Communication Networks, 2018*, 1–22.

Topkara, U., Topkara, M., and Atallah, M.J. (2006). The hiding virtues of ambiguity: Quantifiably resilient watermarking of natural language text through synonym substitutions. In Proceedings of the 8th workshop on Multimedia and security. pp. 164–174. ACM.

Vasilescu, F., Langlais, P., and Lapalme, G. (2004). Evaluating variants of the lesk approach for disambiguating words. In Lrec.

Winstein, K. (2000). Lexical steganography through adaptive modulation of the word choice hash, January 1999. Was disseminated during secondary education at the Illinois Mathematics and Science Academy. The paper won the third prize in the.

Zhang, Z.-H., Jin, X.-M., Wang, J.-M., and Li, D.-Y. (2004). Watermarking relational database using image. In Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 04EX826). Vol. 3. pp. 1739–1744. IEEE.

Zhang, L., Gao, W., Jiang, N., Zhang, L., & Zhang, Y. (2011). Relational databases watermarking for textual and numerical data. In *2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)* (pp. 1633–1636). IEEE.

Zhang, Y., Niu, X., Wu, D., Zhao, L., Liang, J., & Xu, W. (2005). A method of verifying relational databases ownership with image watermark. In *The 6th International Symposium on Test and Measurement* (pp. 6316–6319). PR China: Dalian.

Zhou, X., Zhao, W., Wang, Z., & Pan, L. (2009). Security theory and attack analysis for text watermarking. In *2009 International Conference on E-Business and Information System Security* (pp. 1–6).