Consequences when querying different data types

Taken from: *Pérez Gort, M. L., Olliaro, M., Cortesi, A., and Feregrino Uribe, C., "Semantic-driven watermarking of relational textual databases." Expert Systems with Applications 167 (2021): 114013.*

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

Table 1. Motivating example. Structure of the relation 'Student'.

Student					
Id	**Name**	**Surname**	**Subject**	**Score**	**ProfessorJudgment**
1001	John	Oliver	Mathematics	95	John has improved a lot.
1002	Justin	Fitzgerald	Physics	69	Justin has problems passing Physics.
1003	Andrea	Russo	History	98	Andrea is the first in his History class.
1004	Karla	Olivare	Mathematics	100	Karla is an outstanding student.

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

```
SELECT Name
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 2nd *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 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 1, consider the value of the attribute **ProfessorJudgment** for the tuple with **Id** = 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 [1] or using invisible characters according to the database encoding [2]), 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.

References

[1]. Al-Haj, A. and Odeh, A. (2008). Robust and Blind Watermarking of Relational Database Systems. Journal of Computer Science, 4(12):1024–1029.

[2]. Melkundi, S. and Chandankhede, C., 2015, January. A robust technique for relational database watermarking and verification. In *2015 International Conference on Communication, Information & Computing Technology (ICCICT)* (pp. 1-7). IEEE.