The Effect of the Novel Anti-Collusion Fingerprinting Scheme on the Knowledge from Numeric Databases

Arti Mohanpurkar, Madhuri Joshi

Abstract— The effect of applying the novel anti-collusion fingerprinting scheme on the knowledge obtained from numeric databases is elaborated in this paper. Here, how the classification statistics are affected after fingerprinting numeric datasets is depicted. Several different classifiers are used for the purpose. This technique is primary key independent and resilient to additive attack. It is found to be highly secured against collusion attack due to the special insertion technique and the secret key used during fingerprinting.

Index Terms— Collusion, Copyright Protection, Distortion Minimization, Fingerprinting, Knowledge Preservation, Numeric databases, Particle Swarm Optimization, Additive attack**.**

1 INTRODUCTION

THE popular copyright protection techniques for numeric relational databases are watermarking and fingerprinting. Ownership identification can be done using watermarking while fingerprinting is used for traitor tracing. Each of these techniques involves insertion of acceptable alteration to the database in some or the other form such that the usability of the data is not lost.

A novel anti-collusion technique proposed by the authors of this paper is presented in [17]. The collusion attack which is very specific to fingerprinting technique is avoided by this novel fingerprint insertion technique used. The copyright protection is achieved while ensuring the knowledge preservation. As the numeric databases are highly sensitive to errors, the knowledge preservation can be achieved by optimizing the error to be inserted and avoiding the violation of usability constraints applied. The proposed system performs this error optimization using the Particle Swarm Optimization (PSO) technique.

The proposed system uses owner's secret key for fingerprint insertion, detection and several other intermediate locations too, which makes it highly secure.

The technique proposed in [17] is studied and the effect on the classification statistics of datasets after applying this novel anti–collusion fingerprinting technique is illustrated here. The non-violation of usability constraints is checked in terms of classification statistics like Classification Potential, TPrate and FPrate.

The related work done by the researchers is discussed in the section 2. The overall architecture of the proposed system, the notations used in the mathematical model and the algorithms are described in brief in section 3. Further in section 4, important definitions related to classification statistics, the evaluation method followed by the experimental results with attack analysis are presented.

2 RELATED WORK

There are several popular techniques used for watermarking multimedia data available in literature. Similarly several techniques have come up for watermarking numeric databases. Fingerprinting is comparatively less handled. There are different ways in which the marks of watermark / fingerprint are inserted into the numeric database which is very sensitive to the alteration caused. The different ways of insertion of marks lead to varied amount of effect on the usability and also on the vulnerability. There are several different ways of insertion of marks in numeric databases, each of them have different effects on the usability. Many of them are highly vulnerable while others are robust to the attacks like tuple addition, tuple deletion, additive attack etc. However, the collusion attack is specific to fingerprinting algorithms.

Although there are numerous watermarking and fingerprinting techniques available for numeric databases, but a few closely related techniques are discussed here.

2.1 Multimedia Data

A lot of techniques for watermarking and fingerprinting multimedia data have been studied in past [5], [7]. But the techniques for multimedia data are not suitable for the numeric database which has a typical nature. In multimedia data there is lot of scope for hiding the error which is not true for the database, especially when it is numeric in nature. The insertion, deletion or updating is never required to be done in case of the multimedia data which is very usual about the databases. Moreover insertion of errors in the numeric values of a database may violate the usability constraints which are not acceptable in any circumstance. Thus, the copyright protection algorithms for multimedia data cannot be applied as it is to the numeric databases.

2.2 Bit-level Marking

Bit-level watermarking / fingerprinting techniques mentioned in [5], [6], [7], [8], [9], [10] involve insertion of marks at certain bit positions. Slight alteration in the values may lead to loss of watermark which is not desirable. These are highly vulnerable to alteration attacks.

2.3 Optimized Alteration Embedding

In [1], [2], [12], [14], [15], [16], [17] an optimized alteration is identified based on usability constraints using Particle Swarm Optimization (PSO) technique. This alteration is added or subtracted in all attributes selected for marking based on whether the waterrmark bit [1], [2], [12] to be inserted is one or zero respectively, so that the knowledge is preserved. On the other hand the alteration is added to or subtracted from only one pseudo randomly selected attribute out of all the attributes available for marking, based on whether the fingerprint bit [14], [15], [16], [17] to be inserted is one or zero respectively. The marking sequence will be different for every buyer because of the uniqueness of each buyer's fingerprint. Collusion avoidance is achieved due to the novel insertion technique proposed in [17].

Several techniques for watermarking are summarized in [11] and [13] to realize that there is a need for a fingerprinting technique as proposed in [14], [15], [16], [17].

Another approach to achieve collusion avoidance with minimum distortion is discussed in [15], [16] and the system is found to be more robust as it takes into account the primary key. This approach is applied on Numeric Relational databases.

After indicating the effect of the proposed system on mean and variance in [17], the effect on classification statistics is presented in this paper along with the attack analysis.

3 System Description

An overall description of the system architecture and the mathematical model along with notations is described in this section.

3.1 System Architecture

An overall architecture of the fingerprinting technique is illustrated in fig. 1. This novel anti-collusion fingerprint insertion technique requires the original database to be available as input along with the usability constraint model and the optimized alteration (error) to be inserted. The error to be inserted is optimized using the Particle Swarm Optimization (PSO) technique. The owner's secret key, the buyer id and unique fingerprint for the buyer are also given as input to the system.

The fingerprinted database is further given as input to the fingerprint detection algorithm which can successfully identify the buyer. In case of suspicion of the fingerprinted copy being maliciously attacked, the traitor tracing algorithm is applied which identifies the traitor (culprit's buyer id) or the innocent buyer's id is returned.

3.2 Mathematical Model and Notations

The notations used in the system are presented in Table 1 followed by the mathematical model of the system in fig. 3.

3.3 Algorithms

There are six important steps in this fingerprinting scheme which are as given below:

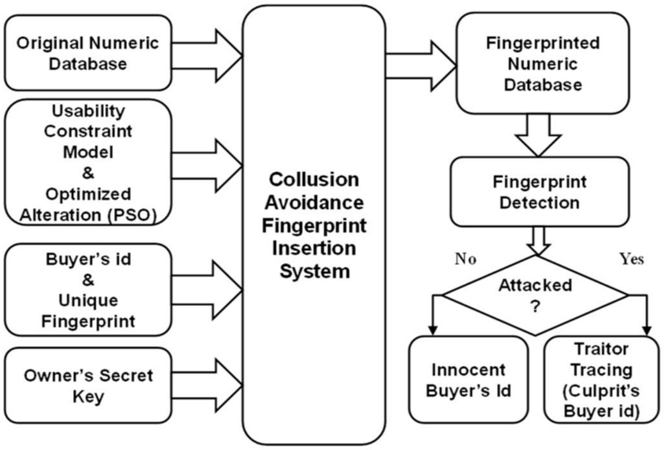
Original Numeric Database
Usability Constraint Model & Optimized Alteration (PSO)
Buyer's id & Unique Fingerprint
Owner's Secret Key
Collusion Avoidance Fingerprint Insertion System
Fingerprinted Numeric Database
Fingerprint Detection
No
Yes
Attacked ?
Innocent Buyer's Id
Traitor Tracing (Culprit's Buyer id)

Fig. 1. An Overall System Architecture

TABLE 1

NOTATIONS USED

Notation	Description	Notation	Description
D_o	Original database	L	Local constraints
D_F	Fingerprinted database	G	Global Constraints
f_{FI}	Fingerprint insertion function	SGV	Secret Grouping value
f_{FD}	Fingerprint detection function	N	Buyer id. N = 1……n
f_{CS}	Function that compares classification statistics	K	Owners Secret Key
f_{TI}	Traitor identification function	F	Fingerprint obtained using Tardos's scheme
f_{AA}	Function to perform additive attack	F`	Fingerprint obtained using Detection Algorithm scheme
f_{PSO}	Function to perform PSO	C_{STO}	Classification statistics of original database
∂	Constraints on system	C_{STF}	Classification statistics of fingerprinted database
TPrate	True Positive rate	FPrate	False Positive rate
S_o	Class Label before fingerprinting	S_F	Class Label after fingerprinting
H_o	Data distribution of original database	H_F	Data distribution of fingerprinted database

1. Fingerprint creation:

Unique fingerprints can be generated for the buyers using different techniques like Boneh-Shaw scheme or Tardos's scheme [3], [4].

2. Usability Constraint Model:

The usability constraint model [1] is designed to automatically identify the local and global constraints based on the classification potential and ranking of the features. High ranking features having high classification potential may not sustain alteration beyond a predefined threshold and it is vice-versa with the low ranking features.

3. Optimization of Alteration using PSO:

Because of the characteristics like its huge rate of success, better quality of result, and smaller amount of time for processing, PSO [1], [12] has been identified as appropriate for constrained optimization problems [8],[9], as compared with several other optimization techniques.

Such optimization of the alteration to be inserted leads to minimum error insertion and hence the knowledge is preserved.

4. Fingerprint Insertion:

It uses a hashing technique. A hash value is calculated for each row using owner's secret key, Buyer's identification and the value of attribute having high classification potential. Dissimilar hash value sequences are created for each buyer. This identifies the attribute within which the mark will be inserted.The tolerable alteration is subtracted from an attribute value if the fingerprint bit is 0 and it is added to the attribute value if the bit is 1. Here the fingerprint is inserted at different locations for different buyers. Thus it becomes impossible for the colluders to form a coalition and guess the location where the fingerprint is inserted. The proposed insertion algorithm given in [17] also reduces insertion complexity to a large extent.

The System S is:
S = {D_o, D_F, f_{FI}, f_{FD}, f_{CS}, f_D, f_{TI}, f_{PSO}, ∂, L, G, SGV, N, K, T, f_{AA}, f_T , Success, Failure}
Where,
$D_o = \{R_1 \ldots\ldots R_n\}$
$D_F = \{R_{1f} \ldots\ldots R_{nf}\}$
$f_F = D_o \rightarrow D_F$
f_{cs} = Compare (C_{STF} , C_{STO}) where,
C_{STO} ={TPrate, FPrate}
C_{STF} ={TPrate_F, FPrate_F}
f_{TI} = Compare (F, F`)
where
F`= {$f_0 \ldots\ldots f_{L-1}$} = {?....?} is Detected Fingerprint and
F = Fingerprint obtained using Tardos's scheme
L = Local constraints
G = Global Constraints
SGV = Secret Grouping value
N =Buyer id. N = 1………n
K = Owners Secret Key
∂ = {Numeric attributes are only identified, Row insertion or deletion prohibited, Tolerable Alteration using PSO}
Success = $C_{STO} = C_{STF}$
Failure = $C_{STO} \neq C_{STF}$

Fig. 2 Mathematical Model

5. Fingerprint Detection:

The detection algorithm [17] takes the key of each buyer and owner's key. Buyer ID for which it detects correctly is a buyer for the fingerprinted database at hand. The same hash function which is used at the time of insertion is used to identify the marked attribute for each row. The usability constraint model is applied to calculate the acceptable alteration Val for each feature.This Val is compared with alteration table value. If the stored alteration is greater than Val then the bit is decoded as 1 else 0.The same procedure continues for each buyer until a correct buyer is detected. This detection algorithm is found to have 100% accuracy in decoding the fingerprint inserted.

6. Traitor Tracing:

The fingerprint detection algorithm itself is applied to the attacked or suspicious database to trace the traitor. The detected fingerprint is compared with each buyer's fingerprint till the traitor is found. If no match is found then the database is not attacked.

4 Experimental Results and Analysis

The system configuration used for experimentation is i5-3210M CPU and 4 GB RAM. The implementation is done using JDK1.5 and Netbeans IDE7.1.0.

The results are obtained on Sonar, Mines vs. Rocks database obtained from UCI repositories with the following specifications:

Database name: Sonar, Mines vs. Rocks, No. of tuples: 208, No. of Attributes: 60, Owner's Secret Key: 3, Secret Grouping Parameter: 0.3, Total No. of buyers: 5

4.1 Verification system for knowledge preservation:

Learning Statistics: Learning statistics [1] contains the classification statistics (or accuracy) of a particular learning algorithm. These statistics include, TPrate, FPrate etc. and they are defined as in (1):

$$TP_{rate} = \frac{TP}{TP+FN} \ , FP_{rate} = \frac{FP}{FP+TN} \qquad (1)$$

where,

TP (True Positive): TP denotes the number of instances of a particular class detected as instances of that class.

FP (False Positive): For a particular class, the number of instances of other class (es) detected as instances of that particular class.

TN (True Negative): For a particular class, the number of instances detected as instances of other class (es).

FN (False Negative): For a particular class, the number of instances of that class detected as instances of other class (es).

Let C_{STO} and C_{STF} be classification statistics of database before and after fingerprinting. If $C_{STO} = C_{STF}$ Then $S_O = S_F$, $C_{PTO} = C_{PTF}$, $H_O = H_F$ which in turn means if classification statistics of the database before and after fingerprinting remains same then knowledge is preserved [1].

C_{STO} = {TPrate, FPrate}

C_{STF} = {$(\text{TPrate})_F$,$(\text{FPrate})_F$}

The information loss

$$\text{CST}_{\text{Loss}} = \frac{\text{CST}_\text{O} - \text{CST}_\text{F}}{\text{CST}_\text{O}} \times 100 \qquad (2)$$

The preservation of knowledge is achieved [1] if $CST_{Loss} = 0$, so this verification system checks whether the knowledge is preserved. The percentage of loss in the knowledge (if any) can be obtained using the formula in (2).

These statistical results are verified using different classifiers. The classifiers like Naives Bayes, Bagging, IBk, JRip, J48 are used here.

The effect on classification statistics can also be represented using a single classifier e.g. Naives Bayes for different users keeping the other parameters constant. It is observed that the classification statistics is found to be the same even after fingerprinting and hence information preservation is achieved. The detailed observations of the classification statistics are shown in Table 2 and its graphical representation is shown Fig.4

TABLE 2

EFFECT ON STATISTICS AFTER FINGERPRINTING FOR DIFFERENT BUYERS USING NAIVES BAYES CLASSIFIER

Effect on statistics	Do TPrate	DWF TPrate	Δ TPrate	Do FPrate	DWF FPrate	Δ FPrate
Buyer 1	0.731	0.731	0	0.25	0.25	0
Buyer 2	0.731	0.731	0	0.25	0.25	0
Buyer 3	0.731	0.731	0	0.25	0.251	0.001
Buyer 4	0.731	0.731	0	0.25	0.25	0
Buyer 5	0.731	0.731	0	0.25	0.25	0

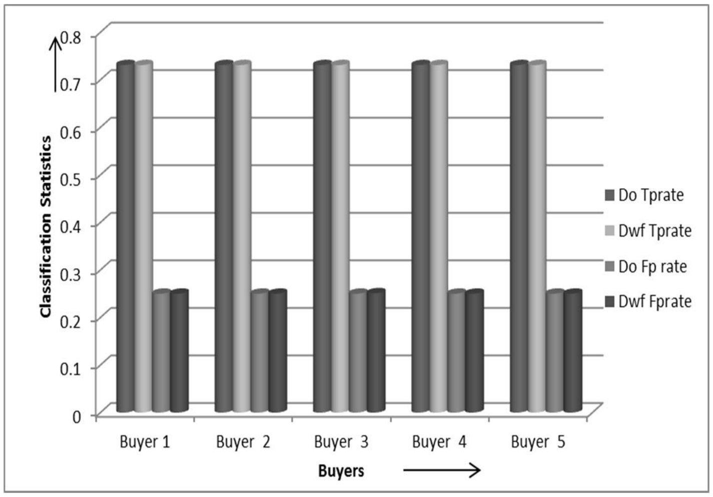

Fig. 4 Graphical representation of the effect on statistics after fingerprinting for different Buyers using Naives Bayes Classifier

The effect of fingerprint insertion on TPrate and FPrate after applying the same usability constraints is shown in Table 3 and its graphical representation is shown in Fig. 5. The classification of the original dataset and fingerprinted dataset is done by applying different Learning algorithms. Table 3 also shows the difference between TPrate and FPrate before and after fingerprinting for a buyer using different classifiers.

TABLE 3

EFFECT ON STATISTICS AFTER FINGERPRINTING USING DIFFERENT CLASSIFIERS FOR SINGLE USER

Effect on statistics	Do TPrate	DWF TPrate	Δ TPrate	Do FPrate	DWF FPrate	Δ FPrate
NaiveBayes	0.731	0.731	0	0.25	0.25	0
SMO	0.875	0.875	0	0.125	0.125	0
IBK	1	1	0	0	0	0
Bagging	0.933	0.928	0.005	0.072	0.077	0.005
JRip	0.851	0.822	0.02	0.138	0.174	0.036
J48	0.976	0.981	0.005	0.026	0.021	0.005

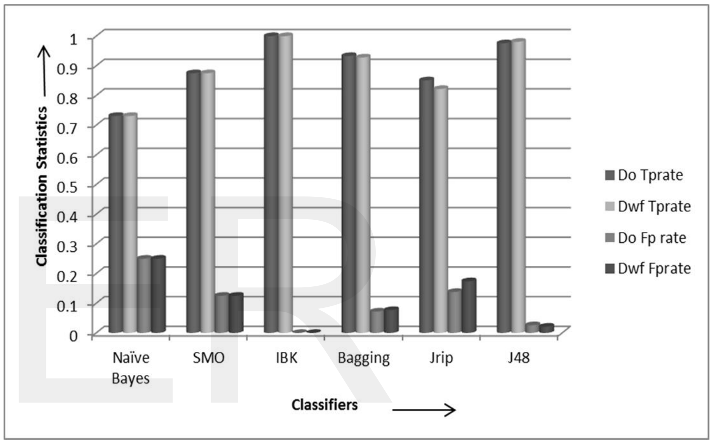

Fig. 5 Graphical Representation of the effect on statistics after fingerprinting using different Classifiers for single user

4.2 Robustness against additive attack:

Even if a malicious buyer tries to insert his own mark on our fingerprinted database we are still able to prove our ownership on the database as the secret key of the owner is not revealed.

4.3 Collusion attack avoidance:

It is already explained in [17] how the collusion attack can be avoided due to the typical way of insertion of the fingerprint in the arbitrarily chosen attribute value. Apart from this, owner's secret key is used during fingerprint insertion. The hash function used for attribute selection is provided with a secret key of the owner along with the value of the attribute having the highest classification potential.

Furthermore, the predefined optimized alteration is added to the chosen attribute value 'fingerprint length' number of times which makes it difficult to identify the fingerprint bits inserted. Thus, not only the fingerprint remains unrevealed for the attacker but it becomes challenging to identify the positions where the fingerprint bits must have been embedded. The effort to find the inserted fingerprint by comparing marked copies of the same dataset sold to different buyers thus goes in vain. This leads to collusion avoidance. Collusion-secure fingerprinting codes are required to be used otherwise for making the fingerprinting system collusion-secure. These schemes have their own way of traitor tracing. But here, the novel fingerprint insertion scheme has resulted in to collusion avoidance and the fingerprint detection and traitor tracing method introduced is found to be highly effective.

5 Conclusion

This paper is an extension of the work done in [14] and [17]. The experimental results after applying the algorithms proposed by the authors of this paper earlier are presented here followed by attack analysis. It is observed that the effect on the classification statistics is minuscule and thus knowledge is preserved. As it does not use the primary key it cannot sustain the tuple insertion and tuple deletion attacks. But the system is found to be robust against additive attack and the collusion attack, where the later is specific to fingerprinting. In future similar techniques can be identified for big data and the data shared on cloud.

Acknowledgment

The authors wish to thank all the researchers in this area who have contributed with several techniques to achieve copyright protection of numeric databases.

References

[1] M. Kamran and Muddassar Farooq, “A Formal Usability Constraints Model for Watermarking of Outsourced Datasets”, *IEEE Transactions On Information Forensics And Security*, Vol. 8, no. 6, June 2013, pp. 1061-1072.

[2] M. Kamran and Muddassar Farooq, “An Information-Preserving Watermarking Scheme for Right Protection of EMR Systems", *IEEE Transactions On Knowledge And Data Engineering*, Vol. 24, No. 11, November 2012, pp. 1950-1962.

[3] Dan Boneh and James Shaw, “Collusion Secure Fingerprinting For Digital data”, *IEEE Transaction on Information Theory*, Vol. 44, No. 5, September 1998, pp. 1897 – 1905.

[4] Gabor Tardos, “Optimal Probabilistic Fingerprint Codes”, *Journal of ACM*, Vol. 55, No. 2, Article 10, May 2008, pp. 10 – 24.

[5] Rakesh Agrawal, Peter J Haas, Jerry Kiernan, “Watermarking relational data: framework, algorithm and analysis”, *The VLDB Journal* (2003) / Digital object identifier (DOI) 10.1007 / s00778-003-0097-x, pp. 157-169.

[6] Radu Sion, Mikhail Atallah, Sunil Prabhakar “Rights Protection for Relational Data”, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, No. 06, June 2004, pp. 1509-1525.

[7] Yingjiu Li, "Fingerprinting Relational Databases: Schemes and Specialties", *IEEE Transactions On Dependable And Secure Computing*, Vol. 2, No. 1, January-March 2005, pp. 34-45.

[8] Mohamed Shehab, Elisa Bertino, Arif Ghafoor, “Watermarking Relational Databases Using Optimization-Based Techniques”, *IEEE Transactions on Knowledge and Data Engineering*, January 2008, Vol. 20, No. 1, pp. 116-129.

[9] Julien Lafaye, David Gross-Amblard, Camelia Constantin, and Guerrouani, “Watermill: An Optimized Fingerprinting System for Databases under Constraints”, IEEE Transactions on Knowledge and Data Engineering, Vol. 20, No.4, APRIL 2008, pp. 532-546.

[10] Ersin Uzun and Bryan Stephenson, “Security of Relational Databases in Business Outsourcing”, HP Laboratories, HPL-2008-168, pp. 1-21.

[11] Raju Halder, Shantanu Pal, Agostino Cortesi, "Watermarking Techniques for Relational Databases: Survey, Classification and Comparison", Journal of Universal Computer Science, Vol. 16, No. 21 (2010), pp. 46-52.

[12] M. Kamran, Sabah Suhail, and Muddassar Farooq, “A Robust, Distortion Minimizing Technique for Watermarking relational Databases Using Once-for-all Usability Constraints”, *IEEE Transactions on Knowledge and Data Engineering* , Vol. 25, No. 12, 2013, pp. 2694 – 2707.

[13] A. A. Mohanpurkar, M. S. Joshi, “Applying Watermarking For Copyright Protection, Traitor Identification And Joint ownership: A Review”, presented and published at *International IEEE Conference WICT 2011*, co-organized by Machine Intelligence Research Labs (MIR Labs) and University of Mumbai, India, during 12th to 14th Dec. 2011, pp.1018-1023.

[14] Ms. Varsha Waghmode, Ms. A.A. Mohanpurkar, “Collusion Avoidance in Fingerprinting Outsourced Relational Databases with Knowledge Preservation”, *International Journal on Recent and Innovation Trends in Computing and Communication*, Vol. 2, No. 5, May 2014, pp.1332 – 1337.

[15] Ms. Namrata Gursale, Ms. Arti Mohanpurkar, “A Robust, Distortion Minimization Fingerprinting Technique for Relational Database”, *International Journal on Recent and Innovation Trends in Computing and Communication* ,Vol. 2 Issue: 6, June 2014, pp. 1737 – 1741

[16] Arti Mohanpurkar, Madhuri Joshi, “A Fingerprinting Technique for Numeric Relational Databases with Distortion Minimization”, 2015, *International Conference on Computing Communication Control and Automation*, IEEE DOI 10.1109 / ICCUBEA.2015.134, pp. 655-660.

[17] Arti Mohanpurkar, Madhuri Joshi, “Fingerprinting Numeric Databases with Information Preservation and Collusion Avoidance”, *International Journal of Computer Applications*, Vol. 132, No. 5, 2015, pp. 13-18.