Fingerprinting Numeric Databases with Information Preservation and Collusion Avoidance

Arti Mohanpurkar
Department of Computer Engineering,
Dr. D Y Patil School of Engineering & Technology,
Charholi Bk. via Lohgaon Pune

Madhuri Joshi
Department of Computer Science and Engineering,
MGM's Jawaharlal Nehru Engineering College,
Aurangabad,

ABSTRACT
Information age demands omnipresence of data. Large data sets are created, maintained and outsourced to the third party experts for data mining. Knowledge and patterns are extracted by using advanced data mining algorithms that assist the decision makers to ensure quick, correct and effective decisions to be made in this world of competition. The outsourcing of these large data sets faces the problem of theft and loss of ownership. The problem of data theft can be handled by fingerprinting i.e. embedding buyer specific marks along with ownership identification marks which further leads to the challenge of knowledge preservation. Thus, a technique which performs fingerprinting with knowledge preservation on numeric relational data to be outsourced is proposed here. It is ensured that the usability constraints are not violated. Knowledge preservation is achieved by optimizing the error to be inserted using Particle Swarm Optimization (PSO), a nature – inspired optimization algorithm. Collusion attack is very well-known in the context of fingerprinting techniques. Here, the proposed system provides a mechanism for avoiding collusion. The proposed system is independent of the primary key.

General Terms
Information Security, Copyright Protection of Relational Databases.

Keywords
Watermarking, Fingerprinting, Collusion, Knowledge preservation, Optimization.

1. INTRODUCTION
With the developments in internet, database applications and techniques of remote access, the demand that the numerous databases on the internet to be permitted to remote query and access, for authorized users has become common, and the challenge to be able to protect the copyright of relational databases has arisen.

Although this trend is a blessing to end users, the data providers are exposed to the threat of data theft. Data providers are therefore keen about demanding a technology which facilitates identification of piracy and the traitors of their databases.

Data mining algorithms require large databases to be outsourced to [1, 10] the data analysis experts. This exposes the database to the theft by the traitors [3, 4, 7] who try to prove their ownership over it or may illicitly and illegally redistribute it to the innocent buyers in the market; affecting ownership and usability of data.

Copyright laws exist, but, the original ownership of that database needs to be proved. The ownership verification can be achieved by using the watermarking techniques available. In literature several such watermarking algorithms are available [1, 2, 5, 6, 8, 9, 10, 11, 12]. Culprits are the buyers who tamper their copy of data and resale the same for their personal benefit. The identification of such a traitor and proof of ownership can together be obtained by the popularly known technique called fingerprinting.

Fingerprinting [7] is nothing but a class of information hiding where buyer specific marks are inserted along with owners' identification to prove ownership and be able to identify traitors. An important threat specific to fingerprinting is collusion attack along with other attacks like tuple addition, deletion, additive attack etc.

An important point to be focused is the minimum distortion [12, 15] or prevention of violation of usability constraints after fingerprinting. The amount of error to be inserted is required to be minimal so that it satisfies the usability constraints expressed in terms of mean and variance and/or preservation of values of classification potential [1, 2, 14], so that the classification statistics is not found to be affected. By applying the mining algorithms before and after fingerprinting it can be verified whether the data still remains usable for destined user. This ultimately leads to extraction of correct knowledge from the huge databases.

Information preservation is thus another important objective which is ensured in two ways:

1. Classification Accuracy Preservation: Classification Potential before (C_{p_o}) and after (C_{p_f}) fingerprinting should remain same. $C_{p_o} = C_{p_f}$
2. Effect on mean and variance should be minuscule $\mu_o = \mu_f$ and, $variance_o = variance_f$

Along with the importance of information preservation [1, 2] it is necessary to deal with the collusion attack [7], where some buyers with access to multiple fingerprinted copies of the same relation but different fingerprints embedded collude to form coalitions. The colluders may create their own copy of the database that may not allow identification of any of the members of the coalition. The fingerprint detection algorithm may accuse an innocent buyer or may find an invalid fingerprint.

The literature suggests several techniques, which watermark or fingerprint numeric relational databases but have not handled collusion attack avoidance [14, 15, 16].

It is found that only this proposed technique which performs fingerprinting in numeric database without primary key that avoids collusion and achieves non-violation of usability

constraints to preserve information. Thus traitor identification is also facilitated. This technique is tried and tested on several publicly available datasets.

The major contribution of the work presented in this paper is:

1. A novel scheme for embedding fingerprinting in numeric database is introduced which is highly secured and leads to collusion avoidance.
2. The system ensures preservation of knowledge by minimizing the error insertion using PSO
3. Original database is not necessary for fingerprint detection which makes Blind Decoding possible.
4. Finds the guilty user(s) (traitor tracing) who is (are) responsible for redistribution of unauthorised copy.
5. Reduced time complexity as compared to that of the existing system is achieved.
6. The primary key independence is one of the key features.

2. RELATED WORK

Several techniques for watermarking and fingerprinting of numeric relational databases with different approaches have been studied. The exhaustive literature survey has motivated the existence of the proposed system.

In [1], a novel model is presented to automatically define user constraints for any data mining dataset so as to achieve information preservation. Classification potential of the features and several other characteristics are preserved such that the mining of the datasets is not affected. Insertion and detection algorithms are found to have very high complexity for huge databases with a large number of numeric attributes as it marks every attribute of every tuple.

The use of Electronic Medical Records systems (EMR) i.e. use of e-health technology is encouraged in [2]. An information-preserving scheme is preferred over threshold-based scheme. The Particle Swarm Optimization (PSO) algorithm is used for optimization of the error to be inserted. The usability constraints are required to be mentioned explicitly.

In [5], Agrawal and Kiernan (AK scheme for convenience) have presented a unique scheme for embedding watermarks within bits of numeric attributes of relations. The detection of the watermark is possible with high probability only if the secret key is known. The technique has properties of blindness and robustness against several attacks. But it is not found to be suitable when non-violation of usability constraints is crucial and also it is useful only in ownership protection.

A buyer-specific mark can be embedded into a data copy provided to a buyer using a fingerprinting scheme [7]; owner can subsequently detect the mark in pirated data and use the mark to identify the traitor who distributed the data. The AK scheme has been extended in [7] to embed an arbitrary bit string as a fingerprint. The errors introduced by fingerprinting are found to be minuscule after thorough analysis but it is not found to be collusion secure.

Different approaches and types of watermarking numeric relational databases are suggested in [8, 9, 11, 13] where information preservation and fingerprinting are not taken into consideration hence no question of collusion attack.

Here [12] author claims that its decoding accuracy is independent of the usability constraints. A technique having features like robustness, minimum distortion for watermarking relational databases is thus proposed. Here the preservation of information is discussed in terms of mean and variance and performs only watermarking.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.

An exhaustive literature survey [10, 11, 13] done here helps us to conclude that there is a need for a system which is able to insert and detect fingerprints in numeric databases that preserves information and avoids collusion.

3. PROPOSED SYSTEM

The proposed system gives a mechanism to avoid this collusion which is very significant and is presented in this paper. Fingerprint insertion mechanism which avoids collusion attack is introduced. The proposed system is about securing the numeric databases without primary key against the loss of ownership attack and illicit redistribution by using the technique of fingerprinting. The proposed system is keen about preservation of information within the database as the fingerprint insertion may lead to changes in data values, which may further result in loss of knowledge. Mining of data after fingerprinting should result into the same knowledge as that before fingerprinting. Knowledge preservation is expressed in terms of difference in mean, variance and standard deviation before and after fingerprinting, which is expected to be minuscule.

3.1 System Architecture

The proposed system consists of three parts: A usability constraint model is developed as the first step. The architecture of the proposed system is illustrated in fig. 1.

The second part contains fingerprint construction, encoding, decoding, and tracing the traitor. The fingerprint insertion algorithm takes the usability constraint model as an input [1].

The analysis of the effect of fingerprinting on efficiency, mean and variance and collusion avoidance is done in the third part of the system. Here, mean, variance and standard deviation are considered as global constraints on the system.

3.2 Algorithms

Unique buyer specific identification marks are inserted into the database using fingerprinting technique. Each buyer is marked with a different identification mark. Identification of owners, buyers and traitors (in case of illicit redistribution) of databases is achieved with the help of these marks.

3.2.1 Fingerprint Construction

The techniques like [3, 4] Boneh Shaw or Tardos etc. can be used for construction of fingerprint code. The proposed scheme uses Tardos's [4] scheme. The unique code can be constructed using any method for unique code generation. As the proposed technique claims to avoid collusion using a typical insertion scheme, the fingerprinting code need not be collusion secure i.e. it can be any unique bit stream.

3.2.2 Fingerprint Insertion

The fingerprint insertion [14] is shown in Algorithm 1 of fig. 3. The hashing technique uses owner's secret key, buyer's identification and the value of an attribute having high classification potential to calculate the hash value H (row) for each row. The different hash value sequences are generated for each buyer. The complexity of the insertion algorithm in [1] is reduced to a large extent due to the proposed method of

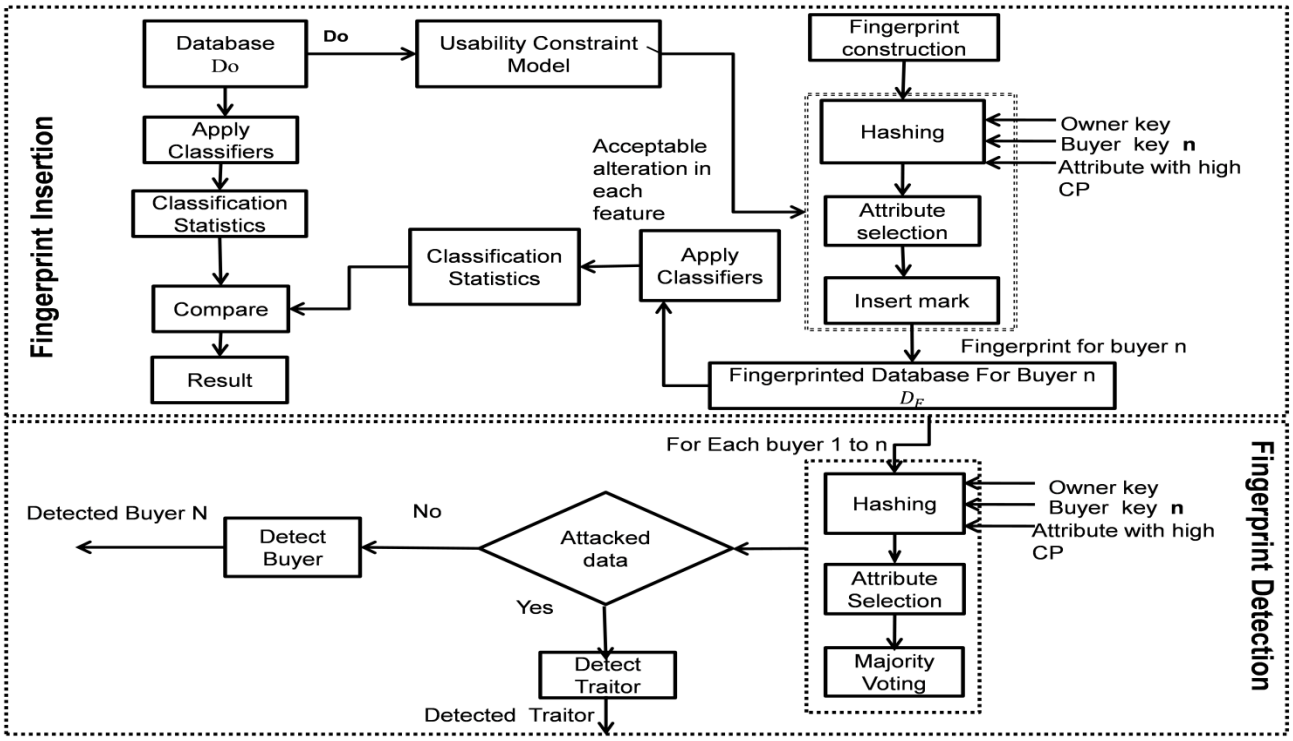

Fig. 1: System Architecture

insertion. The optimization of the alteration to be inserted is done using Particle Swarm Optimization (PSO) algorithm [1, 2].

Calculation for the attribute number is done using a hash value. Input for hash value calculation is owner ID K_o, buyer ID B and the feature in database with high classification potential. It is calculated as,

*Hash value = (K_o * B * High Classification Potential attribute) Mod (Number of attributes)*

Hash value = Round (Hash value, 0)

3.2.3 Fingerprint detection

Algorithm 2 in fig. 3 shows the fingerprint detection [14]. The detection algorithm 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.

3.2.4 Traitor Tracing

To trace the traitor the fingerprint detection algorithm is used on the attacked database. The detected fingerprint is compared with each buyer's fingerprint to identify the traitor.

4. EXPERIMENTS AND RESULTS

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

The **r**esults 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.While [1] has inserted watermark into dataset the proposed system has extended this work to insert fingerprints using a novel technique. Watermark can only be used for ownership protection but fingerprinting is used to trace traitor in addition to ownership protection.

Algorithm1. Fingerprint Insertion

Input: Original dataset Do , Owner's secret key K_o, Buyer's ID B , Acceptable alteration in each feature Δ and Fingerprint code F
Output: Fingerprinted Database D_F for buyer n, Alteration table
Temp==Do
For each row r
Attribute =HASH (K_o, B, Attribute with high CP)
If(Attribute not equal to Attribute with high CP)
If F(bit)==1
Attribute(value) = Attribute(value) + Attribute(Δ)
Alteration table= Attribute(Δ)
Else
Attribute(value) = Attribute(value) - Attribute(Δ)
Alteration table= Attribute(Δ)
End if
End If
End For
return D_F , Alteration table

Fig. 2: Fingerprint Insertion Algorithm

4.1 Effect on Efficiency

The comparison of efficiency of the method in [1] and the proposed system is shown in Table 1 and the graphical representation of the same is shown in fig. 4. The insertion complexity of the technique in [1] is found to be too high. The complexity of the insertion technique is the function of the number of attributes chosen for marking (i.e. columns 'C'). The insertion technique of the proposed system inserts whole fingerprint in only one attribute chosen pseudo randomly from the tuple. Thus it reduces the complexity O(C) to a large extent i.e. more than (approx.) 80%. Thus the efficiency is

inversely proportional to the number of attributes of a tuple selected for marking.

4.2 Effect on Mean and Variance

The effect on statistics like mean, variance and standard deviation can also be represented. It is observed that the statistics are found to be the same even after fingerprinting

Algorithm 2: Fingerprint Detection

Input: Fingerprinted Database D_F , Owner's secret key K_o, Alteration table, Buyer's Fingerprints
Output: Buyer Fingerprint

```
Checking threshold=50%
One=0;
Zero=0;
For each buyer 1 to n
  For each Row
    Attribute =HASH (Ko B, Attribute with high CP)
    Val=Attribute (Δ)
    If alteration>Val
      F'(bit) = 1
      One++
    End if
    If alteration<Val
      F'(bit) = 0
      Zero++
    End if
    End for
Apply majority voting to get Fingerprint F'
Match: F' with F(Buyer)
{ If  Match <= Checking threshold
    Continue to detect F'
    Row = total no of rows
Else if Match >= Matching threshold
    Detected a buyer i          where i ϵ1 to n
          Stop
    Else
          Buyer++
 End if
}
End for
Return Buyer F(n)
```

Fig. 3: Fingerprint Detection Algorithm

and hence information preservation is reassured to be achieved. Detailed observations of the classification statistics (for first 5 attributes out of 60) are shown in Table 2. The effect of fingerprint insertion on mean is graphically shown in figure 5.

4.3 Collusion Avoidance

A hashing function is used by the proposed system or fingerprint insertion. The attribute in each row where a fingerprint mark is to be inserted is identified using the hash function. Using this technique a unique sequence of attributes to be marked for each buyer is generated. As a result a randomized pattern of insertion is generated for each of the buyers. Hence the fingerprinted copies for different buyers of same databases are so different that they cannot collude to find the places of insertion of fingerprint marks.

5. CONCLUSION

The fingerprinting technique facilitates with security against the ownership theft and a provision for traitor tracing (if any unauthorized copy is found). The insertion of fingerprint bits in numeric databases may change the numeric data to some extent. A loss of knowledge may be observed due to these changes in numeric data. Here the work in [1] is extended by finding a novel way for inserting a fingerprint in the database along with the assurance of information preservation. The information preservation is shown in terms of effect on mean, variance and standard deviation after fingerprinting, which is found to be minuscule. A hashing technique is used to randomly find the attribute for fingerprint insertion and achieve avoidance of collusion. The complexity of insertion is reduced by more than 80% over the insertion technique in [1]. The proposed insertion algorithm is primary key independent and it can efficiently perform traitor (if any) tracing. In future the effect of fingerprinting on classification statistics can be studied and copyright protection of Big Data which is publicly available on cloud can be achieved.

6. ACKNOWLEDGMENTS

Special thanks to the researchers in the field who have made the literature available.

7. 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 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, p.p.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,Volume 2, Issue 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 ,Volume: 2 Issue: 6, June 2015, 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.

8. APPENDIX

Table 1. Comparison of efficiency (complexity) of existing system and the proposed system

Comparison of Efficiency	No. of Rows R	No. of Cols C	Length of Fingerprint L	No. of buyers N	Complexity O(C)	**Inference**
Existing System	208	60	2552	5	208 * 60* 2552 * 5 =159244800	98.33 % improvement of Proposed System over that of Existing system
Proposed System	208	60	2552	5	208 * 1 * 2552* 5 = 2654080 (Per row Only one attribute selected for marking)	

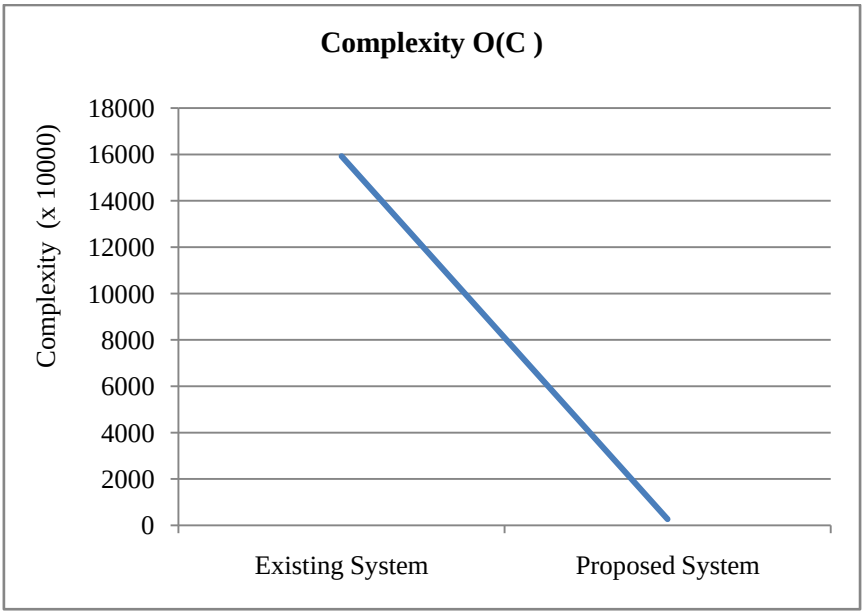

Fig. 4: Graphical representation of comparison of complexity of existing system and the proposed system for the results shown in Table 1

Table 2. The effect on statistics like mean, variance and standard deviation before and after fingerprint insertion

Effect of Fingerprinting on Mean, Variance and Standard Deviation	Mean	Variance	Standard Deviation	Mean	Variance	Standard Deviation
Attribute ID	Before Insertion			After Insertion		
1	0.02924	0.02299	5.29E-04	0.02930	0.02281	5.20E-04
2	0.03856	0.03296	0.00109	0.03862	0.03293	0.001084
3	0.04389	0.03843	0.00148	0.04404	0.03834	0.00147
4	0.0541	0.04653	0.00216	0.05415	0.04653	0.002165
5	0.07555	0.05555	0.00309	0.07557	0.05556	0.003087

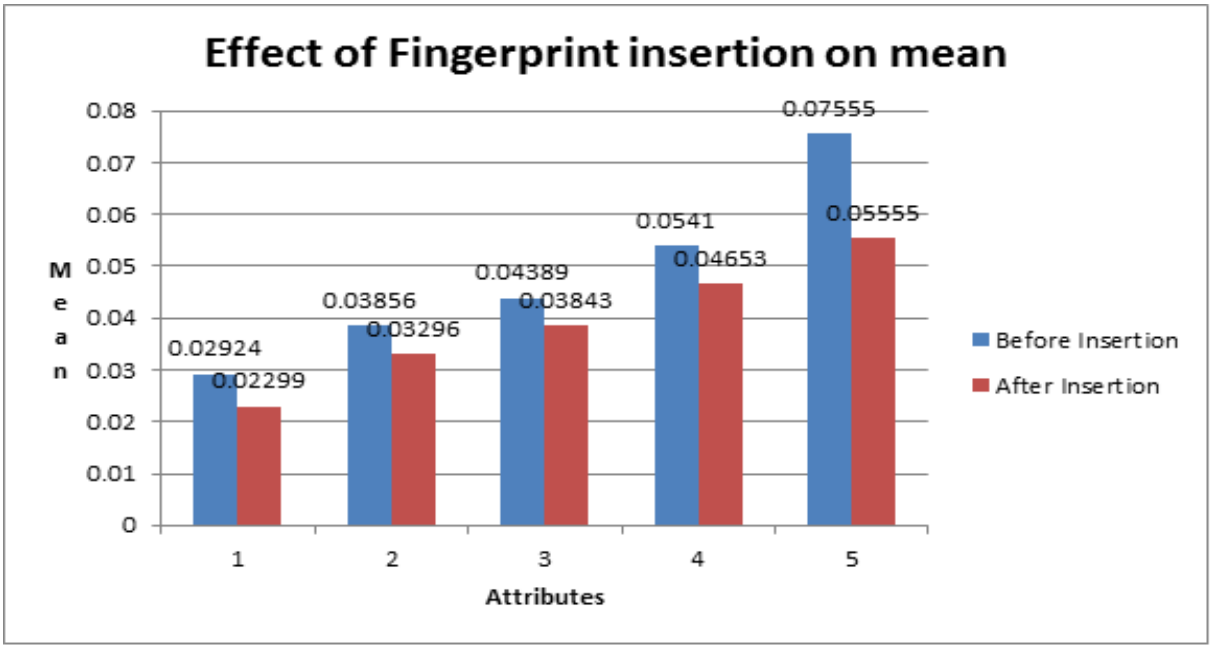

Fig. 5: Graphical representation of effect of fingerprint insertion on mean of (1 to 5) attributes for results shown in Table 2