Relational Database Watermarking Clustering Based Data-Mining Using K-Means Approach

Rittal Patel¹ Jignesh Vania²
¹,²Department of Computer Engineering
¹,²Gujarat Technological University, Gujarat, India

Abstract— Database watermark is some kind of information that is embedded into underlying data for tamper detection, maintaining integrity, ownership proof, traitor tracing. It provides copyright protection of relational data and maintaining integrity of the database information. This research we introduced a cluster-based database watermarking technique which first applies cluster theory to the database watermarking technology. The cluster theory is used to cluster the source data and the clustering results determine the quantity of embedded watermark information and embedded position. For embedding watermark odd-even modifying method is apply. This method denoted the watermarking information to decline the modification of original database. This method discards the traditional method to partition subsets. This technique improve the performance on invisibility, gives minimum distortion rate and able to defend all type of subset attacks.

Key words: Database watermarking, clustering, odd-even modifying, distortion rate

I. INTRODUCTION

The rapid growth of internet and related technologies has offered an unprecedented ability to access and redistribute digital contents. The piracy of digital assets such as software, images, video, audio and text has long been a concern for owners of these assets. Protection of these assets is usually based upon the insertion of digital watermarks into the data [1]. Recent years, people have made great achievements in the digital watermarking technology of images, audio, video. With the development of database technology and applications, the issue of database security becomes more and more important [2]. Such techniques allow the owner of the data to embed an imperceptible watermark into the data. A watermark describes information that can be used to prove the ownership of data, such as the owner, origin, or receipt of the content [3]. Secure embedding requires that the embedded watermark must not be easily tampered with, forged, or removed from the watermarked data [4]. Imperceptible embedding means that the presence of the watermark is unnoticeable in the data. Furthermore, the watermark detection is blinded, that is, it neither requires the knowledge of the original data nor the watermark. Watermarking techniques have been developed for video, images, audio, and text data and also for software and natural language text [5]. There is a rich body of literature on watermarking multimedia data. Most of the work is developed for still images and then video and audio sources.

There is a many differences between multimedia data and relational database. So watermarking of multimedia data cannot be directly used for relational database [6]. Digital watermarking technique has been successively applied to protect the multimedia works and software products. Similarly, database watermarking has been proposed on large database security-control. However, there are some differences between relational database and multimedia data [7]. Firstly, a relational database table consists of many attributes and tuples, but there is no certain ordering between tuples or attributes of a relation table. Secondly, database maintaining operators could frequently change those tuples unlikely other type of multimedia object. Moreover, database tuples processing rely on logical set operational language such as SQL. So database watermarking should also have the ability of real-time update and blind detection and cannot directly adopt those multimedia watermarking method [8]. It is more difficult to ensure the robustness and reversibility of database watermarking.

II. RELATED WORK

Zhiyong Li proposed a watermarking technique to takes advantage of the disorder character among database tuples to cluster them by Mahalanobis distance, and then combines with the polar angle expansion strategy to embed and extract watermark. The scheme shows a high robustness under blind detection for subset selection, addition and modification attack. Theodoros Tzouramanis [11] proposed a watermarking technique novel watermarking scheme for numeric database attributes which is efficient in defeating a range of attacks that may be used to remove or destroy the mark. The proposed scheme is multipurpose because it can be used for both watermarking and fingerprinting. We did not adopt any error correction mechanism to make the watermark more robust against the various kinds of attacks. In real-world applications it is expected that this methodology will protect the embedded information against noises; error correcting codes exist that have a correcting ability. This paper proposes an algorithm for watermarking numeric relational data. This algorithm sorts the bits of each tuple in a secret order and selects some of its data bits to route the tuple to a specific watermark bit and one data bit to be marked by the value of the assigned watermark bit.

Zhi-Hao Zhang proposed a watermarking technique for an identification image is embedded into the relational data for representing the copyright information. To be effective, the detection process for watermark should be visible, imperceptible, secure, reliable and resistant to attack. A novel method for watermarking relational database, which uses an image as watermark. Approach is more intuitive, and it support easy watermark identification.

III. APPROACH OVERVIEW

Allowing for the disorderliness of tuples and attributes, insufficient redundant space of database, along with weak robustness of the general database watermarking algorithm, it is practicable to realize the database watermarking embedding and robust detection with the stable[10], high-
efficiency and large-capacity database tuples clustering method, which is regarded as the basis of database watermarking algorithm in this paper. There into the similarity among databases tuples is measured by Mahalanobis distance since it can effectively eliminate the influence of dimension and correlation interference. Meanwhile there are frequently database maintaining operators on tuples and attributes which would affect the robustness of database watermarking, and we use majority decision method to solve the problem when extracting watermark. Based on the above tuples clustering and majority decision strategy, we present a robust database watermarking framework shown in Figure 1(a) and (b)\textsuperscript{[11]}

A. Clustering approach

Here we apply the fast clustering method to the classification of database tuples, which begins with classifying samples roughly, then uses certain regulations to adjust the categories gradually based on the distance between samples. It is suitable for clustering analysis of large data sets. The similarity of samples is measured by distance\textsuperscript{[15]}. Due to the disunity of various attributes units in database, in order to eliminate the influence of dimension, this paper adopts k-Mean distance to cluster the tuples, which discards the traditional method to partition subsets. We also introduce an odd-even modifying method to embed watermarking information shows it is effective\textsuperscript{[16]}.

B. K-mean method

The K-Means clustering algorithm is a partition-based cluster analysis method\textsuperscript{[10]}. According to the algorithm we firstly select k objects as initial cluster centers, then calculate the distance between each cluster centre and each object and assign it to the nearest cluster, update the averages of all clusters, repeat this process until the criterion function converged.

Algorithm: K-means algorithm

K-means (k, D)

Input: The no of the cluster k; the database D.

Output: k clusters

Method:

Init clusters();
do{
    find closest cluster(object);
    calc New clustcenters ();
} until (the centroids no longer change);

C. Odd-even modifying method

Watermark is converted into a binary flow which can embed the information a bit by a bit. When embed it into database, compare the parities of watermark information and original data. If they have same parity does not modify the original data. If the watermark information is 1 while the data is an even number, then the original data is modified to the nearest odd number. If watermark information is 0 while the data is odd number, then modify the original data to the nearest even number.

Algorithm: Embedding of odd-even modifying method

Embedding (watermark, data)

Input: watermark, data

Output: watermarking-data

Method:

lastnum =f(data);
flag=lastnum%2;
if (watermark==1)
    if(flag==0)
        lastnum =lastnum+1;
    if(watermark==0)
        if(flag==1)
            if(lastnum == 9)
                lastnum =lastnum-1;
            else
                lastnum =lastnum+1;
        wdata =update(data,lastnum);
        return wdata;

D. Extraction of odd-even modifying method

Extraction(data)

Input: watermarking data

Output: watermark

Method:

lastnum =f(data);
flag=lastnum%2;
if(flag==1)
    wat =1;
else

Fig.1: Robust Database Watermarking Framework based on Tuples Clustering and Majority Decision Strategy
IV. PROPOSED METHODOLOGY

In this section, we describe the watermark embedding Algorithm Steps for Proposed System Embed the Watermark into Original Database using k-means and Odd-even modifying method. In this research methodology, it clusters the original data. Calculate the length of watermark and attributes numbers to embed watermark information in each cluster. Hash function is used to select the tuples to embed watermark. For the selected tuples, it compares the parities of original data and watermark information and then modifies the parities of original data according to the odd-even modifying method.

Algorithm: Watermark-Embedding algorithm

Watermark – embed (database, Key)

Input: database, Key

Output: watermark

Method:

results=k-means (database, Key);
for(int i=0;i<k;i++)
{
 Ni=Ci.Number;
 If(Ni-Numt/k>δ1 II Ni-Numt/k> δ2)
 regulate(IWIi);
}
for(i=0;i<k;i++)
{
 for(int j=0; j<IwIi;j++)
 {
 flag=hash(Pi);
 if(flag)
 Embedded (Ci.field[m]);
 }
 Save picture(binarystr,output);

V. RESULTS AND ANALYSIS

A. Effect to the data

We use mean value and variance to get the difference between original database and database embedded. We select five attributes which are embedded watermark and calculate mean value changes and variance changes which is shown in table.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Mean Value</th>
<th>Variance</th>
<th>Mean value changes</th>
<th>Variance changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>13.8857</td>
<td>1.8589</td>
<td>1.7*10^3</td>
<td>2.24*10^-7</td>
</tr>
<tr>
<td>A2</td>
<td>15.4115</td>
<td>2.7683</td>
<td>1.3*10^3</td>
<td>2.50*10^-8</td>
</tr>
<tr>
<td>A3</td>
<td>24.6553</td>
<td>2.3556</td>
<td>5.2*10^-6</td>
<td>5.50*10^-6</td>
</tr>
<tr>
<td>A4</td>
<td>17.4567</td>
<td>7.5678</td>
<td>4.3*10^-3</td>
<td>4.50*10^-6</td>
</tr>
<tr>
<td>A5</td>
<td>21.8765</td>
<td>8.6834</td>
<td>5.8*10^-6</td>
<td>2.65*10^-7</td>
</tr>
</tbody>
</table>

Table 1: Calculation of variance and Mean Value

Table 1 shows that after we embed the watermark, the mean value and variance change very small and ensure the availability of database.

B. Distortion

We use some targeted values from the dataset1 and dataset2 to find the distortion. $r.Ai$ is the numeric attribute $Ai$ of tuple $r$, where $i = 1, ..., s$.

<table>
<thead>
<tr>
<th>r.A</th>
<th>Watermarked r.A</th>
<th>Distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td>.44890</td>
<td>.44892</td>
<td>0</td>
</tr>
<tr>
<td>.02842</td>
<td>.02835</td>
<td>0.24</td>
</tr>
<tr>
<td>.65000</td>
<td>.65001</td>
<td>0</td>
</tr>
<tr>
<td>.00570</td>
<td>.00560</td>
<td>175</td>
</tr>
<tr>
<td>.03672</td>
<td>.03673</td>
<td>0.02</td>
</tr>
<tr>
<td>.00020</td>
<td>.00020</td>
<td>0</td>
</tr>
<tr>
<td>.07625</td>
<td>.07631</td>
<td>0.07</td>
</tr>
<tr>
<td>.54540</td>
<td>.54538</td>
<td>0</td>
</tr>
<tr>
<td>.03608</td>
<td>.03611</td>
<td>0.08</td>
</tr>
<tr>
<td>.02544</td>
<td>.02551</td>
<td>0.27</td>
</tr>
</tbody>
</table>

Average Distortion=24(%)  

Table 2: Distortion for Existing System

<table>
<thead>
<tr>
<th>r.A</th>
<th>Watermarked r.A</th>
<th>Distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td>181.62</td>
<td>181.62</td>
<td>0</td>
</tr>
<tr>
<td>181.03</td>
<td>181.50</td>
<td>0.47</td>
</tr>
<tr>
<td>184.10</td>
<td>179.33</td>
<td>0.77</td>
</tr>
<tr>
<td>181.66</td>
<td>173.20</td>
<td>0.46</td>
</tr>
<tr>
<td>181.96</td>
<td>187.81</td>
<td>0.85</td>
</tr>
<tr>
<td>184.31</td>
<td>170.50</td>
<td>1.31</td>
</tr>
<tr>
<td>166.10</td>
<td>180.69</td>
<td>14.59</td>
</tr>
<tr>
<td>181.93</td>
<td>180.40</td>
<td>0.53</td>
</tr>
<tr>
<td>181.74</td>
<td>184.16</td>
<td>0.42</td>
</tr>
<tr>
<td>179.59</td>
<td>166.07</td>
<td>13.52</td>
</tr>
</tbody>
</table>

Average Distortion=6.58(%)  

Table 3: Distortion of proposed system for dataset1

<table>
<thead>
<tr>
<th>r.a</th>
<th>Watermarked r.a</th>
<th>Distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td>181</td>
<td>181</td>
<td>0</td>
</tr>
<tr>
<td>239</td>
<td>235</td>
<td>4</td>
</tr>
<tr>
<td>235</td>
<td>219</td>
<td>16</td>
</tr>
<tr>
<td>219</td>
<td>217</td>
<td>2</td>
</tr>
<tr>
<td>217</td>
<td>212</td>
<td>5</td>
</tr>
</tbody>
</table>

Average Distortion=5.4(%)  

Table 4: Distortion of proposed system for dataset2

The results of experiment show this technique has a good performance on invisibility and is able to defend all kinds of subset attacks.
C. Comparison and Graph

![Graph showing comparison of distortion](image)

Table shows the difference of distortion between existing system and proposed system. Some targeted attributes selected for the comparison between original database and watermarked database. rAi is the numeric attribute Ai of tuple r, where i = 1, …, s.

VI. PERFORMANCE MEASURES

A. Blind detection

At the time of watermark extraction, the algorithm only needs to provide the secret key and does not need original database.

B. Invisibility

The number of clusters and initial cluster center are only known by database owner, so position of watermark is not able to be detected by attackers. At the same time, watermark information is embedded at different attributes in different clusters which make the watermark more hidden. The number of clusters and initial cluster center are only known by database owner, so position of watermark is not able to be detected by attackers. At the same time, watermark information is embedded at different attributes in different clusters which make the watermark more hidden.

C. Robust

Firstly, the k-means method assures the location of the embedded watermark is irregular. Secondly, the technique in this paper chooses grey image which has more information as watermark. These properties ensure our technique efficiently defends the attack of subset selection, subset adding and subset updating. It improves the robust of this technique in a great degree.

D. Imperceptibility

The modifications caused by marks should not reduce the usefulness of the database. In addition, the commonly used statistical measures such as mean and variance of the numerical attributes should not be significantly affected.

VII. CONCLUSION & FUTURE WORK

In this paper, we developed a new watermarking technique for relational database using the cluster theory. This technique partitions subset through clustering the data in original database, and determines the quantity of embedding and embedding position by clustering results. The cluster method makes watermarking information more disperse and hidden. Furthermore, at the period of embedding watermark, we adopt odd-even modifying method which assures the minimum modification to original database. The results of experiment prove the efficiency of the technique. Future work will focus on designing a semi-fraile watermarking scheme so that the embedded watermarks can be robust to small modifications and fragile to severe modifications and watermarking scheme for both numeric and non-numeric attributes.

REFERENCES


All rights reserved by www.ijsrd.com 1494


