DUPLICATE RECORD DELETION IN RELATIONAL DATABASE MANAGEMENT SYSTEMS

B. Bharathi,
M.Tech. Student , DEPT of CSE
KMM Institute of Technology and Sciences
Tirupati
E-mail: bharati.boyapati4@gmail.com

C. Sudarsana Reddy
Asst. Professor, DEPT of CSE
KMM Institute of Technology and Sciences
Tirupati
E-mail:cheruku1sudarsana2reddy3@gmail.com

ABSTRACT: Relational database is defined as a set of individual units called relations. Record repetition or duplication is common in many cases of real life situations in the usages of relational databases. In relational database, if the same record is stored in more than one place then such records are termed as duplicate records. Finding duplicate records and then replacing them by the most suitable one best record or a group of best records is called a new operation, fusion operation. Whenever duplicate tuples are found in the relations of any database, those records ought to be replaced with one individual best approximate record that represents the joint information of all the duplicate tuples. Present paper proposes a new technique to find out duplicate tuples and then remove away all such duplicate tuples with the correct real world tuples. Total procedure consists of two steps. In the first step duplicate copy of tuples in the relation are classified by means of the class label and in the second step of the procedure then for each set of duplicate tuples functional dependency means or union method is applied to replace duplicate tuples with the corresponding correct real world single tuple. One opportunity is to replace one set of duplicate tuples through one correct real world tuple. Another possibility is to replace two or additional sets of duplicate tuples in the relation by one set of new and accurate real world tuples. From time to time removing of duplicate tuples in the relations of any relational database can create problems in referential integrity violations. Any such violation must be controlled and updated both syntactically as well as semantically correct in relations.

Index terms: *Finding duplicate records, removing duplicate records, data propagation, Integrity constraints, Relations Relational databases*

I. INTRODUCTION

In many day to day real applications relations in the databases are inherently associated with duplicate records. Many real life applications belonging to users have duplicate records in relations. Find and then removing of duplicate data tuples in the relational database is the most important and newest research procedure. Data repetition is also known as thing resolution or record connection. replica data tuples are present in one or more relational databases when there exit numerous descriptions of the same real world entity. The presence of duplicate tuples causes many database safeguarding problems. several of the reasons for the existence of duplicate tuples are presence of misplaced attribute values, data entry errors, typing errors and not following principles in data entry and data preservation. Finding and after that removing of duplicate data tuples in the relational database is the mainly important and most recent research topic. In universal, data tuples are duplicated in one or more relations of any relational databases when there exit several descriptions of the same real world entity (record). Duplicate tuple discovery and replacement with accurate tuple is unavoidable in a lot of relations of the relational database. A particular process is needed to take concern of integrity constraint violations that happen when duplicate tuples are detached from the relations.

Many desired data Integrity constraints already enforced on the relational database have to be satisfied before and after deleting the original copy data tuples. First determine all of the duplicate records in the relations of any relational database and then subsequently replace all such replica tuples by a single correct and original tuple or sometimes group of original tuples. Most important thing is that referential data integrity must be considered and restricted without any spread of data fusion. Several integrity constraints managing strategies such as on delete cascade, on update cascade, set null, set not null, and restrict operations are accessible in database modifications. These techniques are syntactically correct but semantically incorrect.

Present paper introduces a new technique delete duplicate tuples in the relations of a relational database. This new method is called union fusion function method that is applicable for attribute values. Current study moreover proposes another duplicate tuple deleting method by using a new method called functional dependency advancing technique. This special and general functional dependency constraint approach controls both the partial preservative functions and also complete preservative functions.

Current study also proposes one more new technique to model and manage duplication of tuples. For example, consider a very big relation called TEMP, in order to find duplicate tuples in the given relation, TEMP, at first one can apply a classification technique in order to classify all the tuples, after this operation based on the class labels, and duplicate tuples are identified and then these duplicate tuples are replaced by the correct real world tuple or tuples. The classification tool called decision tree is the perhaps the majority important and highly interpretable classification technique to the data classification. Decision tree is used as a benchmark technique before applying any other classification technique in data analysis.

II. PROBLEM DEFINITION

Data duplication is frequent in many real time applications and particularly in the relations of any relational databases. Finding duplicate tuples and then replacing them by one original tuple is called a fusion operation. During fusion operation all data integrity constraint violations must be controlled carefully and relational database must be managed in a consistence state before and after database updates and also after removal of duplicate tuples in the relations of relational databases.

In the present research paper, a sample set of three relations viz, 1.COLLEGE, 2.CONFERENCE and 3.CONDUCTED_CONFERENCES is considered as running example for understanding purpose. In the relation COLLEGE tuples 1 and 2 are duplicated because of some reasons such as typographic errors, missing of values and lack of standard data representation procedures etc.

Both one and two duplicate tuples describe the same real world entity. These two duplicate tuples are identified and consequently replaced by one equivalent real and correct tuple. Finding and then removing these duplicate tuples with one correct and real world tuple is called a fusion operation. Present study also proposes a new fusion operation called union. Union fusion operation accepts a set of duplicate tuples and then replaces with one correct real world entity. Working principle of union fusion function operation is explained below:

Union of College_Code = {3G}$\cup${3G} = {3G}

Union of College_Name = {KMM}$\cup${KMM} = {KMM}

Union of Principle_Name = {Rama}$\cup${Rama} = {Rama}

Union of Affiliated_University = {null}$\cup${ JNT University} = {JNT University}

In the COLLEGE relation duplicate tuples 1 and 2 are replaced by the following single tuple using proposed new union fusion function technique. The replacing function may be either partially preservative or complete preservative function. Partially preservative function is defined as follows: There exists t$\in$DupSet such that t[A] = REP(Dup)[A]

When A $\subseteq$ DupSet, it is called partial preservative and when A = DupSet, it is called complete preservative replace function. For example, let A = {SNo, College_Code, College_Name, Principle_Name}

Here t[A] = Rep(Dup)[A] and Let B = {JNT University}, then t[B] = Rep[B]

In this particular example, replacing function is partial preservative but not complete (full) preservative. Hence, 1 and 2 duplicate tuples in the COLLEGE relation are mapped with one correct real world tuple. In the COLLEGE relation, tuples 5, 6, and 7 are duplicate tuples. This is an instance for complete preservative. These three duplicate tuples are shown in the FIGURE 6 and then they are replaced by the single tuple shown in the FIGURE 7.

Here, t[all attributes] = RepDup[all attributes]. Complete preservative replacing function replaces a set of tuples with another equivalent and simplified set of tuples.

Consider once again the relation COLLEGE1 with the functional dependency that hold on it, College_Code → {College_Name,Principal_Name,Affiliatedto}. The functional dependency state that when two values on different tuples are same on the attribute College_Code then all values of the three attributes in the right side of the functional dependency are also same. That is, if t_1[College_Code] = t_2[College_Code] then t_1[College_Name,Principal_Name,Affiliatedto] = t_2[College_Name,Principal_Name,Affiliatedto]. Therefore duplicate tuples 1 and 2 in the COLLEGE1 relation are replaced by tuple 1 by applying functional dependency constraints.

First proposed method takes union among the attributes. Second proposed method takes union among the all the tuples. Third, proposed functional dependency method is more generalized version of the above two methods. Third, proposed method also takes care of partial and complete preservative functions also.

SNo	College_Code	College_Name	Principal_Name	Affiliated_University
1	3G	KMM	Rama	Null
2	3G	KMM	Null	JNT University
3	3C	Vidyanikethan	CSReddy	Null
4	2D	SHREE	Null	JNT University
5	6E	Annamacharya	Dr.MuniSwamy	JNT University
6	6E	Annamacharya	Dr.MuniSwamy	JNT University
7	6E	Annamacharya	Dr.MuniSwamy	JNT University

FIG-1 COLLEGE

Coference_Id	Conference_Type
Con1	IEEE
Con2	SPRINGER
Con2	ACM

FIG-2 CONFERENCE

SNo	Confrence_Id	Numberof_days	Start_Date
1	Con1	3	18/6/2009
1	Con2	1	10/12/2006
1	Con3	4	3/9/2012
2	Con1	3	8/6/2009
2	Con2	1	10/12/2006
3	Con1	6	3/5/2013
5	Con1	4	29/12/2010
6	Con1	5	29/12/2010
7	Con1	6	29/12/2010

FIG-3 CONDUCTED_CONFERENCES

SNo	College_Code	College_Name	Principal_Name	Affiliated_University
1	3G	KMM	Rama	Null
2	3G	KMM	Null	JNT University

FIG-4

SNo	College_Code	College_Name	Principal_Name	Affiliated_University
1	3G	KMM	Rama	JNT University

FIG-5

SNo	College_Code	College_Name	Principal_Name	Affiliated_University
5	6E	Annamacharya	Dr.MuniSwamy	JNT University
6	6E	Annamacharya	Dr.MuniSwamy	JNT University
7	6E	Annamacharya	Dr.MuniSwamy	JNT University

FIG-6

SNo	College_Code	College_Name	Principal_Name	Affiliated_University
5	6E	Annamacharya	Dr.MuniSwamy	JNT University

FIG-7

SNo	Confrence_Id	Numberof_days	Start_Date
1	Con1	3	18/6/2009
1	Con2	1	10/12/2006
1	Con3	4	3/9/2012
1	Con1	3	18/6/2009
1	Con2	1	10/12/2006
3	Con1	6	3/5/2013
5	Con1	4	29/12/2010
5	Con1	5	29/12/2010
5	Con1	6	29/12/2010

FIG-8 CONDUCTED_CONFERENCES AFTER PROPAGATION

SNo	Confrence_Id	Numberof_days	Start_Date
1	Con1	3	18/6/2009
1	Con2	1	10/12/2006
1	Con3	4	3/9/2012
1	Con1	3	18/6/2009
1	Con2	1	10/12/2006

FIG-9

SNo	Confrence_Id	Numberof_days	Start_Date
3	Con1	6	3/5/2013

FIG-10

SNo	Confrence_Id	Numberof_days	Start_Date
5	Con1	4	29/12/2010
5	Con1	5	29/12/2010
5	Con1	6	29/12/2010

FIG-11

SNo	Confrence_Id	Numberof_days	Start_Date
1	Con1	3	18/6/2009
1	Con2	1	10/12/2006
1	Con3	4	3/9/2012
3	Con1	6	3/5/2013
5	Con1	4	29/12/2010

FIG-12 FINAL CONDUCTED_CONFERENCES_CORRECTED

SNo	College_Code	College_Name	Principal_Name	Affiliated_University
1	3G	KMM	Rama	JNT University
3	3C	Vidyanikethan	CSReddy	Null
4	2D	SHREE	Null	JNT University
5	6E	Annamacharya	Dr.MuniSwamy	JNT University

FIG-13 FINAL COLLEGE_CORRECTED relation

III. ALGORITHM

Let R be a relation of tuples and assume that set of all the duplicate tuples are denoted by theta. That is, theta is a subset of R. Let R_1 be the child relation corresponding to the parent relation, R. this algorithm will be executed in two steps. In the first step duplicate records are identified and then in the second step identified duplicated records are replaced with the correct real world original records and also these changes are trasfered to the dependent (referenced) relations in a more semantically correct way in addition to the syntactic correctness of relations with respect to many database operations such as insert, delete, and update.

Assume that sample parent relation R = COLLEGE, and the dependent child relation of the parent relation is taken as R'' = CONDUCTED_CONFERENCES. Also assume that tuples t ϵ theta $\subseteq$ R and tuples $t^* \subseteq R''$. The relationship between parent and child relations is one to many from COLLEGE to CONDUCTED_CONFERENCES. In the COLLEGE relation tuples 1 and 2 are duplicated and this type of duplication is deleted using union operation of between or among the attributes. Tuples 5, 6, and 7 are also duplicated and these types of duplication of records are removed by taking the union operation among the tuples but not among the attributes. In the second case sets of duplicate records are identified and then replaced with the one or more sets of real world and original or correct records.

INPUT:

Relations containing duplicated records

OUTPUT:

Relations with duplicate records replaced with original and correct records

1. for each of the tuple t ϵ original-set do
2. In the relation R_1 find a set of tuples whose foreign
3. key matches with the primary key of the tuple t in
4. R and suppose that S_t be the set of all such tuples
5. for all t^{star} ϵ S_t replace foreign key values in R_1 with
6. the respective primary key value of the tuple t ϵ R
7. end-of-for
8. end-of-for
9. for each of the set s_t find projected set of all tuples based on
10. their primary key value
11. end-of-for
12. Now apply and compute union operation for all of the S_t sets

IV. CONCLUSIONS

Data repetition either knowingly or unknowingly is common in many real life applications in the day-to-day life of human beings. Records are presented more than once in many relational databases of RDBMSs because of many causes such as typing errors, modification errors, inclusions of null values, non-standard method representation, and typographic errors. There exists no standard and deterministic method for the identification of duplicated records in the

relations of relational database systems. When there exist no specific and deterministic standard method for detecting duplicate records it is very difficult to find and delete or replace duplicate records in relational databases. Hence, there is a need for creating a good framework that consists of specific standard methods for duplicate record identification and then replacement.

REFERENCES

[1] Antoon Bronselaer, Daan Van Britsom, and Guy De Tr_e
IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 27, NO. 5, MAY 2015

[2] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, third ed. Morgan Kaufmann, 2011.

[3] Arun K. Pujari – Data Mining