A Closure Set Based Approach for Identifying Data Dependency in Relation Database

Varade Sagar Balu
M.Tech (CSE) IV Sem, Lord Krishna College of technology Indore, India.

Vijay Kumar Verma
Asst. Prof. Dept. of CSE, Lord Krishna College of technology Indore, India.

Abstract – In DBMS data store in tables and each row represents an object with relationship in the real world. Storing data into tables is not enough. To represent the data accurately we need integrity constraints. These constraints ensure that the data stored in the database reflects accurately the real-world restrictions. The most important integrity constraints are also called data dependencies. Data dependencies are conditions stating that for any value of a set of attributes there is at most one value for a set of target attributes. Data dependency traditionally plays an important role in the design of relational databases. The discovery of FDs from databases has recently become a significant research problem. Several algorithms have been developed in the recent year to solve the problem of data dependency. In this paper we proposed an efficient approach for identify the data dependency in the rational database. In proposed approach we used closure set which help to generate meaningful candidates.

Index Terms – DBMS, Attributes, Relationship, Dependency, Integrity, Closure Set.

1. INTRODUCTION

Database design methodology normally starts with the first step of conceptual schema design in which users' requirements are modeled as the entity relationship(ER) diagram. The next step of logical design focuses on the translation of conceptual schemas into relations or database tables. Conceptual schema and logical designs are two important steps regarding correctness and integrity of the database model. Data normalization is a common mechanism employed to support database designers to ensure the correctness of their design. Normalization transforms unstructured relation into separate relations, called normalized database. The main purpose of this separation is to eliminate redundant data and reduce data anomaly (i.e., data inconsistency as a result of insert, update, and delete operations). There are many different levels of normalization depending on the purpose of database designer. Most database applications are designed to be either in the third normal forms in which their dependency relations are sufficient for most organizational requirements. Dependency discovery has attracted a lot of research interests from the communities of database design, machine learning and knowledge discovery since early 1980s. There are three types of data dependencies are often involved in the discovery these are

1. Functional dependencies.
2. Inclusion dependencies (INDs)
3. Conditional Functional Dependency (CFD).

FDs represent value consistencies between two sets of attributes while INDs represent value reference relationships between two sets of attributes. In recent years, the discovery of conditional functional dependencies (CFDs) has also seen some work. The aim of dependency discovery is to find important dependencies holding on the data of the database. These discovered dependencies represent domain knowledge and can be used to verify database design and assess data quality.

2. RELATED TERMINOLOGY

2.1 Armstrong's Axioms

Following three inference axioms for FDs defined on sets of attributes X, Y, and Z known as Armstrong's Axioms

(1) F1. (Reflexivity) If $Y \subseteq X$, then $X \rightarrow Y$.

(2) F2. (Augmentation) If $X \rightarrow Y$, then $XZ \rightarrow YZ$.

(3) F3. (Transitivity) If $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

2.2 Closure

Let $X,Y \subseteq U$ and F be a set of FDs. The closure of X ($X \neq \Phi$) w.r.t. F, denoted X^+, is defined as $\{Y|X \rightarrow Y$ can be deduced from F using Armstrong's Axioms$\}$. The closure of F denoted F+, is the set of all FDs that can be deduced from F using Armstrong's Axioms. Definition indicates FD $X\rightarrow Y$ holds if and only if $Y \in X^+$. For X, $Y\subseteq U$, we use $(XY)^+$ to denote the closure of X UY. Let $F = \{A \rightarrow C, BD \rightarrow AC\}$. By Definition 2, $A^+ = \{A, C\}$ and$(BD)^+ = \{A, B, C, D[9]\}$.

2.3 Cardinality of the partition

Let $X \subseteq U$ and let $t_1 \ldots\ldots\ldots t_n$ be all the tuples in a relation r(U). The partition over X, denoted π_X, is a set of the groups such that t_i and t_j, $1 \leq i, j \leq n$, $i\neq j$, are in the same group if and only if $t_i[X] = t_j[X]$. The number of the groups in a partition is called the cardinality of the partition, denoted $|\pi_X |$. For a single attribute vi, we use Xvi to denote the partition of the set of attributes $|\pi_{X|}$.

2.4 Nontrivial closure

Let F be a set of FDs and X^+ be the closure of X w.r.t. F. The nontrivial closure of X w.r.t. F, denoted X*, is defined as X* = $X^+ -$ {X}. For X,Y ⊆ U, we use (XY)* to denote the nontrivial closure of the set of attributes X ∪ Y, similarly to how we use $(XY)^+$ to denote the closure of attributes X ∪ Y.We have $(XY)^+$ = (XY)*∪XY[10].

3. RELATED TERMINOLOGY

In 2011 WenfeiFan,FlorisGeerts&JianzhongLi , Ming Xiong proposed "Discovering Conditional Functional Dependencies". They provide three methods for CFD discovery. The first, referred to as CFD Miner, is based on techniques for mining closed item sets. The other two algorithms are developed for discovering general CFDs. One algorithm, referred to as CTANE, is a level wise algorithm that extends TANE, a well-known algorithm for mining FDs. The other, referred to as Fast CFD, is based on the depth-first approach used in Fast FD, a method for discovering FDs[1].

In 2012 ThiernoDiallo&JeanMarcPetit Dirty proposed "Discovering Editing Rules for Data Cleaning". They proposed efficient techniques to address the discovery problem of Editing Rules (eRs) and heuristics to clean data. They implemented and evaluated proposed techniques on real-life databases. Experiments show the feasibility, the scalability and the robustness of our proposed method [2].

In 2012 Jixue Liu, Jiuyong Li, Chengfei Liu, &Yongfeng Chen proposed "Discover Dependencies from Dataa Review". They proposed reviews for functional dependency, conditional functional dependency, approximate functional dependency, and inclusion dependency discovery in relational databases and a method for discovering XML functional dependencies. They reviewed the methods for discovering Ds, AFDs, CFDs, and INDs in relational databases and XFDs in XML databases[3].

In 2012 Zbigniew W. Raś&Li-ShiangTsayproposed 'Discovering (frequent) constant conditional functional dependencies 'Authors mainly focus on two types of approaches: one which extends the notion of agree sets and the second extending the notion of non-redundant sets closure, and quasi closure[4].

In 2013 SujoyDutta& Dr. LaxmanSahoo proposed "Mining Full Functional Dependency to Answer Null Queries and Reduce Imprecise

Information Based on Fuzzy Object-oriented Databases". They proposed the concept of fuzzy functional dependency is extended to full functional dependency on similarity based fuzzy object oriented data model. They also add a data mining algorithm to discover all functional dependencies among attributes. Their major objective is to reduce imprecise information over databases[5].

In 2014 P.Andrew, J.Anish kumar&S.Balamurugan, proposed "Investigations on Methods Developed for Effective Discovery of Functional Dependencies". They give the details about various methods to discover functional dependencies from data. They also discussed Effective pruning for the discovery of conditional functional dependencies[6].

In 2015 R.Santhya, S.Latha& S.Balamurugan, "Further Investigations on Strategies Developed for Efficient Discovery of Matching Dependencies". They give details about various methods prevailing for efficient discovery of matching dependencies. They also show that concept of matching dependencies (MDs) has recently been proposed for specifying matching rules for object identification. Similar to the functional dependencies with conditions, MDs can also be applied to various data quality applications[7].

In 2015 Thorsten Papenbrock & Jens Ehrlich proposed "Functional Dependency Discovery:

An Experimental Evaluation of Seven Algorithms". They describe, evaluate, and compare the seven most cited and most important algorithms. They classify the algorithms into three different categories, explaining their commonalities. The descriptions provide additional details. Their evaluation of careful re-implementations of all algorithms spans a broad test space including synthetic and real-world data [8].

4. PROPOSED METHOD

Consider a simple database with five attribute

Employee Number(Emp_N), Department No (D_N), Year, department Name(D_Name) and Manager No(Mgr_N)

Table 1 simple employee Database

S N	Emp_N (A)	D_N (B)	Year (C)	D_Name (D)	Mgr_N (E)
1	1	1	1985	Production	5
2	1	5	1994	Marketing	12
3	2	2	1992	Sales	2
4	3	2	1998	Sales	2
5	4	3	1998	Purchase	2
6	5	1	1975	Production	5
7	6	5	1988	Marketing	12

Table 2attribute with cardinality

Attribute	Cardinality	candidate set
A	6	A
B	4	B

C	6	C
D	4	D
E	3	E
AB	Free set	A,B,D,E
AC	Free set	A,C,E
AD	Free set	A,B,D,E
AE	Free set	A,E
BC	Free set	B,C,D,E
BD	Not free	------
BE	Not free	-----
CD	Free set	B,C,D,E
CE	Not free	------
DE	Not free	-------

Maximal equivalence class are{ {1,2},{1,6},{2,7},{3,4,5}} the concur set for the pair of tuples (1,2) is concur set con(1, 2)={A} Similarly, we have con(1,6) = con(2,7) = con(3,4) ={B,D,E}, con(3,5) = {E}, con(4,5) ={C,E} so the concur set of r

con(r)={A, BDE , E CE}

Table 3 Attribute closures cardinality

Attribute	Closure	FD
A	A	
B	B,D,E	B → D, E
C	C,E	C → E
D	B,D,E	B → D, E
E	E	
AB	A,B,C,D,E	AB → C
AC	A,B,C,D,E	AB → D, E
AD	A,B,C,D,E	AD → C
AE	A,B,C,D,E	AE → B,C,D
BC	A,B,C,D,E	BC → A
BD	------	
BE	------	
CD	A,B,C,D,E	CD → A
CE	------	
DE	------	

So numbers of functional dependencies are

BC A, → CDA, D B, → AC B, → →

AEB, → AB C →

AD C, → AE C, BD, → AC D, → AE D, →

B E, → → C E, → D E →

5. PROPOSED ALGORITHM

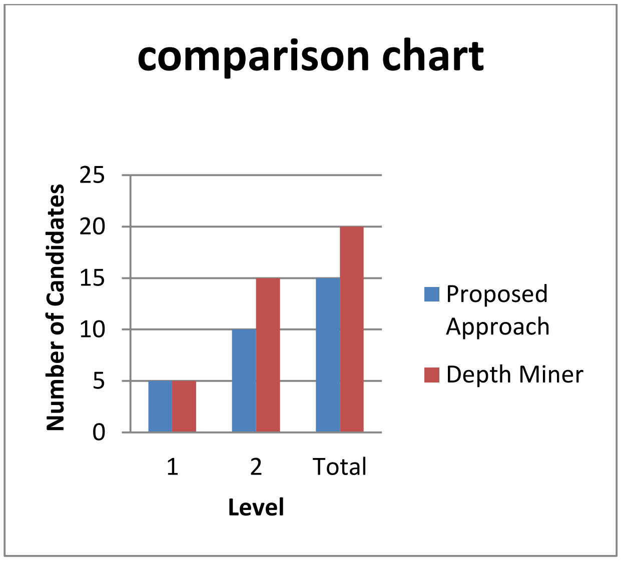
The proposed algorithm has the following steps

1. Find out the cardinality of each of the attribute.
2. Generate candidates set.
3. Identify the free set.
4. Find out closure set from the free set.

Identify the dependency according to closure set

6. COMPARATIVE ANALYSIS

We have compared the proposed method with Dep_Miner and counted number of candidates traversal at different level



7. CONCLUSION AND FUTURE WORKS

Form the comparison graph it is clear that the proposed method generates less candidates for mining data dependency form the relational database. The proposed methods use simple calculation and closure set to identify the required dependency. In future we try to find conditional dependency and inclusion dependency.

REFERENCES

[1] Wenfei Fan , FlorisGeerts&Jianzhong Li "Discovering Conditional Functional Dependencies IEEE Transactions On Knowledge And Data Engineering, Vol.23, No. 5, May 2011.

[2] ThiernoDiallo& JeanMarcPetit "Discovering Editing Rules For Data Cleaning" 9th InternationalWorkshop on Quality in Databases (QDB) 2012.

[3] Jixue Liu, Jiuyong Li, Chengfei Liu, &Yongfeng Chen "Discover Dependencies from Data—A Review" IEEE Transactions On Knowledge And Data Engineering, Vol. 24, No. 2, February 2012 251.

[4] Zbigniew W. Raś&Zbigniew W. Raś"Data Mining, Modeling and Management", Vol. 4, No. 3, 2012.

[5] SujoyDutta& Dr. LaxmanSahoo "Mining Full Functional Dependency toAnswer Null Queries and Reduce ImpreciseInformation Based on Fuzzy ObjectOriented Databases" International Journal of Computer

[6] Science & Engineering Technology (IJCSET) ISSN : 2229-3345 Vol. 4 No. 03 Mar 2013.

[7] P.Andrew,J.Anishkumar&S.Balamurugan "Investigations on Methods Developed forEffective Discovery of FunctionalDependencies" International Journal of Innovative Research in Computerand Communication Engineering(An ISO 3297: 2007 Certified Organization)Vol. 3, Issue 2, February 2015.

[8] R.Santhya, S.Latha, &S.Balamurugan, "Further Investigations on StrategiesDeveloped for Efficient Discovery of MatchingDependencies "International Journal of Innovative Research in Computerand Communication Engineering(An ISO 3297: 2007 Certified Organization)Vol. 3, Issue 1, January 2015.

[9] Thorsten Papenbrock2 Jens Ehrlich "Functional Dependency Discovery:An Experimental Evaluation of Seven Algorithms" Proceedings of the VLDB Endowment, Vol. 8, No. 10Copyright 2015 VLDB Endowment 21508097/15/06.

[10] Hong Yao · Howard J. Hamilton "Mining functional dependencies from data" Data Min Knowl Disc (2008) 16:197–219DOI 10.1007/s10618-007-0083-9.

[11] Jalal Atoum, Dojanah Bader and Arafat Awajan "Mining Functional Dependency from Relational DatabasesUsing Equivalent Classes and Minimal Cover Journal of Computer Science 4 (6): 421-426, 2008ISSN 1549-3636© 2008 Science Publications.