Volume 3, Issue 6, June 2018 ISSN (online): 2456-0006
International Journal of Science Technology Management and Research
Available online at: www.ijstmr.com

Improving TANE Algorithm to Reduce Dependency and Search Space

Rakhi Prajapati
M Tech(CSE) 4th Sem.
Lord Krishna College of Technology
Indore M. P. India

Vijay Kumar Verma
Assistant Professor Dept. C.S.E.
Lord Krishna College of Technology
Indore M. P. India

Abstract: ***Dependency discovery has attracted a lot of research interests from the communities of database design, machine learning and knowledge discovery since early 1980. 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). Discovering FDs can also help a database designer to decompose a relational schema into several relations through the normalization process to get rid or eliminate some of the problems of unsatisfactory database design. In this paper we introduce an efficient approach which uses heuristic-driven, depth first search to compute minimal FDs from a relation instance***

Keywords: redundant, attribute, discovery, dependency, Integrity, constraints, normalization

I. INTRODUCTION

Integrity constraints play an important role in query optimization. The role of query optimization is to find an efficient way of processing a query using all sorts of information about the data stored in the database such as statistics and integrity constraints. These mapping queries are derived from the correspondences between a target value and a set of source values. But when the value correspondences involve several relations of the source database_ one needs a way of joining the tuples of these relations. Functional dependency plays a key role in database normalization. Discovering FDs can also help a database designer to decompose a relational schema into several relations through the normalization process to get rid or eliminate some of the problems of unsatisfactory database design. Discovered dependencies are to assess the quality of data. Thus by analyzing the discovered dependencies and the missed dependencies that should hold among attributes of data, errors may be identified and inconsistencies among attributes may be located. As a result, the data quality is assessed. In recent years, the demand for improved data quality in databases has been increasing and a lot of research effort in this area has been given to dependency discovery.Functional dependency (FD) traditionally plays an important role in the design of relational databases, and the study of FDs has produced a rich and elegant theory. The problem addressed in this proposed work is to find all functional dependencies among attributes in a database relation. Specifically, we want to improve on previous proposed methods for this problem. Early methods for discovering of FDs were based on repeatedly sorting and comparing tuples to determine whether or not these tuples meet the FD definition

➢ TERMINOLOGY

Axioms

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

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

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

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

Cardinality

Let $X \subseteq U$ and let t1---------- tn be all the tuples in a relation r(U). The partition over X, denoted πX, is a set of the groups such that ti and tj, $1 \leq i, j \leq n$, $i \neq j$, are in the same group if and only if ti[X] = tj[X]. The number of the groups in a partition is called the cardinality of the partition, denoted $|\pi X|$. For a single attribute we use vi to denote the partition of the set of attributes $|\pi X|$.

In 2.1 table, by Definition, πA = {{t1, t2, t3, t4, t7}, {t5, t6}}, and πCE = {{t1, t2}, {t3}, {t4, t7}, {t5}, {t6}}. By Definition 3, $|\pi A| = 2$ and $|\pi CE| = 5$.

II. LITERATURE REVIEW

In 2009 Fabien De Marchi proposed "CLIM: CLosed Inclusion dependency mining in databases". They present a novel approach IND mining can be optimized by a closure operator, as it is done for support-based pattern mining. As a consequence, and through a data pre-

processing, satisfied closed INDs can be mined with very few programming efforts, using closed item set mining procedure as a basic operator They show that how IND mining problem can be solved using existing programs devoted to closed set mining, which is a new and quite unexpected result. They show two main benefices of proposed work 1. The proposed work out perform over existing approaches. 2. Contribute to the story of declarative pattern mining. Indeed, an important issue is to fit problems into common frameworks to allow optimizers to devise the best programs given a mining query.

In 2010 Jixue Li u Jiuyong Li "proposed "Discover Dependencies from Data - A Review". They reviews methods 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 also reviewed the methods for discovering FDs, AFDs, CFDs, and INDs in relational databases and XFDs in XML databases. They show that the dependency discovery problem has an exponential search space to the number of attributes involved in the data .

In 2011 Wenfei Fan , Floris Geerts , Jianzhong Li , Ming Xiong proposed "Discovering Conditional Functional Dependencies". They investigate the discovery of conditional functional dependencies (CFDs). They show that CFDs are a recent extension of functional dependencies (FDs) by supporting patterns of semantically related constants, and can be used as rules for cleaning relational data. However, finding quality CFDs is an expensive process that involves intensive manual effort. To effectively identify data cleaning rules, we develop techniques for discovering CFDs from relations. Already hard for traditional FDs, the discovery problem is more difficult for CFDs. They provide three methods for CFD discovery. The first, referred to as CFD Miner, is based on techniques for mining closed item sets, and is used to discover constant CFDs, namely, CFDs with constant patterns only. Constant CFDs are particularly important for object identification, which is essential to data cleaning and data integration. 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. It leverages closed-item set mining to reduce the search space. As verified by our experimental study, CFD Miner efficiently discovers constant CFDs. For general CFDs, CTANE works well when a given relation is large, but it does not scale well with the arity of the relation..

In 2013 Sujoy Dutta & Dr. Laxman Sahoo 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. From this functional dependency, we shall be able to reach full functional dependency the major objective of this paper is to reduce imprecise information over databases. Different degrees of similarity to the elements in each domain are introduced and compared with similarity relation for the representation of "fuzziness" in the fuzzy object-oriented data model based on fuzzy similarity database model. An attempt has been made to answer null queries using analogical reasoning in basis of full functional dependency on similarity based fuzzy object-oriented data model. An algorithm to find out full functional dependencies from semantic relations has been provided. The approach is based on considering partitions of the relation and deriving valid dependencies from the partitions. The algorithm searches for dependencies in a level wise manner. They showed how the search space can be pruned effectively, and how the partitions and dependencies can be computed efficiently. Partition and equivalence classes are also used to find out the full functional dependency easily and efficiently.

In 2014 P.Andrew, J.Anishkumar and S.Charany proposed "Investigations on Methods Developed for Effective Discovery of Functional Dependencies" . They give details about various methods to discover functional dependencies from data. Effective pruning for the discovery of conditional functional dependencies is discussed in detail. Di conditional Functional Dependencies and Fast FDs a heuristic-driven, Depth-first algorithm for mining FD from relation instances are elaborated. Privacy preserving publishing micro data with Full Functional Dependencies and Conditional functional dependencies for capturing data inconsistencies are examined. The approximation measures for functional dependencies and the complexity of inferring functional dependencies are also observed. Compression – Based Evaluation of partial determinations is portrayed. This survey would promote a lot of research in the area of mining functional dependencies from data. They also give detailed about various methods to discover functional dependencies from data. Effective pruning for the discovery of conditional functional dependencies is discussed in detail. Di conditional Functional Dependencies and Fast FDs a heuristic-driven, Depth-first algorithm for mining FD from relation instances are elaborated. Privacy preserving publishing micro data with Full Functional Dependencies and Conditional functional dependencies for capturing data inconsistencies are examined[6].

In 2015 R.Santhya1, S. Latha proposed "Further Investigations on Strategies Developed for Efficient Discovery of Matching Dependencies". They give details about various methods prevailing in literature for efficient discovery of matching dependencies. The 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 such as detecting the violations of integrity constraints. The problem of discovering similarity constraints for matching dependencies from a given database instance is taken into consideration. This survey would promote a lot of research in the area of information mining. This paper detailed about various methods prevailing in literature for efficient discovery of matching dependencies. The 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 such as detecting the violations of integrity constraints. The problem of discovering similarity constraints for matching dependencies from a given database instance is taken into consideration. This survey would promote a lot of research in the area of information mining[7].

In 2016 Akshay Kulkarni proposed "Functional Dependencies Discovery in RDBMS". They presented TANE, a proficient algorithm for finding functional dependencies from larger databases. TANE is based on partitioning the sets of rows with respect to their attribute

values which makes testing the validity of functional dependency fast even for big databases. The results have shown that the algorithm is faster in use. It is observed that for benchmark databases the running times have improved. Functional dependencies are important metadata which can used to gain knowledge. Discovery of functional dependency can help in removing inconsistent and redundant data we propose a algorithm, TANE, for the discovery of functional and approximate dependencies from relations. The approach is based on deriving dependencies from partitions and searching for it in level-wise manner.

In 2017 Thorsten Papenbrock et al. proposed classify the algorithms into three different categories, explaining their commonalities. They describe all algorithms with their main ideas. The descriptions provide additional details where the original papers were ambiguous or incomplete. Our evaluation of careful re-implementations of all algorithms spans a broad test space including synthetic and real-world data. We show that all functional dependency algorithms optimize for certain data characteristics and provide hints on when to choose which algorithm. In summary, however, all current approaches scale surprisingly poorly, showing potential for future research. Shown that FD discovery is still an open research : None of the state-of-theart algorithms in our experiments scales to datasets with hundreds of columns or millions of rows. Given a dataset with 100 columns and 1 million rows, which is a reasonable size for a table, DFD , Dep-Miner, FastFD will starve in runtime, whereas Tane, Fun, and FD Mine will use up any available memory. This observation indicates potential for future research.

III. PROBLEM WITH TANE

In a level wise manner the algorithm searches for free sets of increasing sizes. At the level corresponding to FDs with a lefthand side of size s, the algorithm knows from the previous level the free sets of size s and their quasi closure as well as the collection of candidate free sets of size s+1. It first computes the closure of the free sets of size s and displays the FDs of the form

$X \longrightarrow A$ where X is a free set of size s and $A \epsilon X^+ \setminus X^0$ Then_ it computes the quasiclosure of the candidate free sets of size s+1 using the closure of the free sets of size s, Then it prunes the candidate free sets X of size s" that are not free sets_ based on the number of distinct values of X and of its maximal subsets that are free sets. Finally it generates the candidate free sets of size s " from the free sets of size s_ The authors found that their approach outperforms TANE in all configurations investigated_ which they explain by the fact that the number of FDs tested in their approach is less than in TANE

table 1 Dependency by TANE

X	$\Pi_x(r)$	X^0	X^+	Dependency
A	6	A	A	-
B	5	B	B,D,E	**B→D,E**
C	6	C	C,E	**C→E**
D	4	D	D,B,E	**D→B,E**
E	3	E	E	-

The problem addressed in this paper is to find functional dependencies among attributes in a database relation by removing redundant attribute. Specifically, we want to improve on previous proposed methods for this problem. Early methods for discovering of FDs were based on repeatedly sorting and comparing tuples to determine whether or not these tuples meet the FD definition. The disadvantage of this approach is that it does not utilize the discovered FDs as knowledge to obtain new knowledge. If A →B has been discovered, a check is still made to determine whether or not AC→ B holds, by sorting on attributes AC and comparing on attribute B. Instead, AC→ B can be directly inferred from the previously obtained A→B without sorting and comparing tuples again. This approach is inefficient because of this extra sorting and because it needs to examine every value of the candidate attributes to decide whether or not a FD holds.
TANE generate 14 dependencies in these dependencies some of the dependencies are redundant dependencies. Like B derives D and D derive B (B→D, D→B). So they are equal equivalent (B↔D). From the generated dependencies AB→C AD→C, B→E, D→E are redundant dependencies. Our approach is to remove these redundant dependencies and generate correct minimal dependencies.

IV. PROPOSED APPROACH

Discovering minimal functional dependencies
Input: a relation r
Output: minimal functional dependencies for relation r
(1) Agree Set: computes agree sets from r
(2) Cmax Set: derives complements of maximal sets from agree sets
(3) find quasiclosure
(3) Left Hand Side: computes lhs of functional dependencies from complements of maximal sets
(4) Delete Redundant Candidates And Dependencies: find equivalence and remove them also replace deleted attribute by their equivalent
(5)Fd Output: outputs functional dependencies

Table 1 Dependency by modified TANE

X	$\Pi_x(r)$	X^0	X^+	Dependency
A	6	A	A	-
B	**5**	**B**	**B,D,E**	**B→D,E**
C	6	C	C,E	**C→E**
E	3	E	E	-

V. RESULT AND ANALYSIS

Comparison based on number of dependencies

For comparing the performance of the proposed approach compare with TANE. We have taken 7 tuple with five attribute. We compare the TANE and Proposed approach using number of dependency at different level. Table 3 number of dependency at different level.

Table 3 Number of dependency at different level

Level	TANE	Proposed approach
1	8	8
2	14	10
Total	30	16

CONCUSION

We have suggested a new approach to reduce extra attribute to find out data dependency form a given relational database algorithm. Discover FDs which utilizes the mathematical prosperities of the database. In the proposed work we us the concepts of concur set, agree set and closure properties. The aim of proposed algorithm is generate meaning full dependencies and optimize the time and memory requirements. We compare proposed approach with TANE algorithm. From the example and implementation it is clear that proposed approach works efficiently as compared to the depth minor and has a better performance

REFERENCE

1. Jixue Liu, Jiuyong Li " Discover Dependencies from Data - A Review" School of Computer and Info. Sci., University of South Australia {jixue.liu, jiuyong.li}@unisa.edu.au 2 Faculty of ICT, Swinburne University of Technology cliu@swin.edu.au 2010.
2. Wenfei Fan , Floris Geerts , Jianzhong Li & Ming Xiong" Discovering Conditional Functional Dependencies TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL.23, NO. 5, May 2011.
3. Thierno Diallo & JeanMarc Petit "Discovering Editing Rules For Data Cleaning" This article was presented at the 9th International Workshop on Quality in Databases (QDB) 2012.
4. Jixue Liu, Jiuyong Li, Chengfei Liu, and Yongfeng Chen "Discover Dependencies from Data—A Review" IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 24, NO. 2, FEBRUARY 2012.
5. Sujoy Dutta "Mining Full Functional Dependency to Answer Null Queries and Reduce Imprecise Information Based on Fuzzy Object Oriented Databases" International Journal of Computer Science & Engineering Technology (IJCSET) ISSN: ISSN: 2229-3345 Vol. 4 No. 03 Mar 2013.
6. P.Andrew, J.Anishkumar & S.Balamurugan "Investigations on Methods Developed for Effective Discovery of Functional Dependencies International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 3, Issue 2, February 2015.
7. R. Santhya, S. Latha & S. Balamurugan "Further Investigations on Strategies Developed for Efficient Discovery of Matching Dependencies" International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 3, Issue 1, January 2015.
8. Thorsten Papenbrock & Jens Ehrlich "Functional Dependency Discovery: An Experimental Evaluation of Seven Algorithms" International Conference on Very Large Data Bases, August 31st September 4th 2015, Kohala Coast, Hawaii. Proceedings of the VLDB Endowment, Vol. 8, No. 10 Copyright 2015 VLDB Endowment 21508097/ 15/06.
9. Zbigniew W. Ras "Data Mining, Modelling and Management". Department of Computer Science, University of North Carolina, Charlotte, NC 28223, USA and Institute of Computer Science, Warsaw University of Technology, 00-665 Warsaw, Poland Vol. 4, No. 3, 2012.
10. Felix Naumann "Detecting Functional Dependencies" Profiling & Cleansing Summer 2013
11. Kanika Sood "Comparison of Functional Dependency Extraction Methods and an Application of Depth First Search" Graduate School of the University of Oregon in partial fulfillment of the requirements for the degree of Master of Science June 2014.
12. Fabien De Marchi "CLIM : CLosed Inclusion dependency Mining in databases" This work has been partially Funded by the French National Research Agency DEFIS 2009 Program, project DAG ANR-09-EMER-003-01.
13. Jalal Atoum, Dojanah Bader and Arafat Awajan "Mining Functional Dependency from Relational Databases Using Equivalent Classes and Minimal Cover" Journal of Computer Science 4 (6): 421-426, 2008 ISSN 1549-3636 © 2008 Science Publications.
14. Katalin Tunde Janosi Rancz and Viorica Varga "A Method For Mining Functional Dependencies In Relational Database Design Using FCA" Studia Univ. Babes_{Bolyai, Informatica, Volume LIII, Number 1, 2008.