International Journal of Research in Advent Technology, Special Issue, August 2018
E-ISSN: 2321-9637
International Conference on "Topical Transcends in Science, Technology and Management" (ICTTSTM-2018)
Available online at www.ijrat.org

Survey on Foreign Key Identification, Keyword Search and Inclusion Dependencies

KAMALAMMA. K. V
Associate Professor, Department of CS&E
R. L. Jalappa Institute of Technology
Doddaballapur, Karnataka, India

Dr. AJEET. A. CHIKKAMANNUR
Professor, Department of CS&E
R. L. Jalappa Institute of Technology
Doddaballapur, Karnataka, India

ABSTRACT

Information searching from relational database is difficult to the ordinary users; they don't have knowledge of database schema and structured Query language (SQL). Keyword search is a solution to the above problem, where keyword query is a simple and user friendly search model. Foreign keys form one of the most fundamental constraints for relational database, since they do not always defined in existing database for various reasons. Relationships between attributes can be detected automatically are inclusion dependencies. It provides solid basis for deducing foreign key constraints. This paper presents in detail literature survey about Keyword searching, foreign key identification, Inclusion Dependency as a base for our feature work of blending Relational Database Management System (RDBMS) and NOSQL.

Keywords:SQL; Foreign KeyIdentification; Keyword Searching; Inclusion Dependency.

I. INTRODUCTION

RDBMS provides a facility for the database to assists user to query the well-structured information using SQL, and also user can search the unstructured information by using the keyword based on scoring and ranking, and do not need users to understand any database schema.

RDBMS contains vast amount of data at government agencies, research organizations and home user personal computers, this data can be accessible only through SQL query or Database schema. If the users are not familiar with the above, it is hard to retrieve the data in RDBMS. To facilitate access to this data keyword search is used. Keyword search allows user to query the database quickly without the knowledge of database schema and SQL. Keyword search also helps in identifying unexpected answers that are difficult to identify from SQL queries. Currently to search the keywords search engines are available on top of sets of documents. When a user provides a set of keywords the searching engine retrieves all the documents that are associated with these keywords.

In RDBMS contains millions of relational tables, to establish the relationship between these tables one of the most important constraint is primary/ foreign key identification. Identification of foreign key is difficult step in working and understanding with the data. So in the database schema explicit specification of foreign key constraint is allowed by database system. A FKC(Foreign Key Constraint) requires that tuples of a relation containing the foreign keys are dependent on tuples of a relation containing the primary key. So it represents the relationship between the tuples. Most of the databases do not provide identification of foreign keys for several reasons, such as lack of support for checking foreign keys constraints, or lack of database knowledge.

An inclusion dependency is the existence of foreign key attributes in a table whose values must be a subset of the values of the corresponding primary key attributes. Inclusion dependency shows the little influence on designing of databases. Split the group of attributes that participate in an inclusion dependency is not functional dependencies. RDBMS has not given much interest to discover the INDs for several reasons such as, the difficulty of the problem due to the potential number of candidate INDs and Lack of popularity.

II. METHODS

A.Foreign key identification

Jan motlandPavelKordík [1],presented relational learning, they described the relationship with foreign key constraints in relational database. And also presented how to identify primary key and foreign key constraints from metadata about the data quickly. He decomposed the relationship problem into two sub problems: Identification of primary key and identification of foreign key constraints. Identification of primary key is performed in two stages: scoring and optimization. The same approach is taken for foreign key identification also. The different features used by primary key scoring are: data types, Doppelganger, ordinal position, string distance and keywords. Two additional features are: contain Null and isUnique. He used ILP solver to optimize the primary key. The different features taken to identify foreign key are: Data types, Data

lengths, String distance. Forign key optimization based on the following constraints: Unity, Acyclicity, Completeness, Doppelganger.

Alexandra Rostin, Oliver Albrecht, Jana Bauckmann, Felix Naumann, and Ulf Leser [2],presented machine learning approach to identify the foreign keys. First compute all IND (Inclusion Dependencies) then judge by a binary classification algorithm. FKs essentially states that tuples of a relation (containing the foreign key) are dependent on tuples of another relation (containing the primary key). Common characteristics to identify foreign keys are: set of values of FKs often covers almost all values of its primary key, FK attribute name often contains exactly or approximately the name of its PK. IND is a precondition for a FKs, some features for classifying INDs are; Distinct dependent values, Coverage, Dependent and Referenced, Multi dependent, Multi referenced, Column name, value length Diff, Out of range, Typical NameSuffix, tableSizeRatio. Machine learning approach is heavily dependent on the data set used and the differences and similarities to the data sets used for evaluation.

Zhimin Chen, VivekNarasayya, and SurajitChaudhuri [3], described the technique for fast foreign key detection in power pivot. Power pivot extends pivot table functionality by allowing it to be specified over foreign key joins of multiple tables. He used different components to identify the foreign key: Local Pruning, Candidate Scoring, Join Path enumeration, Containment verification and Output foreign keys connecting F,D. Local Pruning is based on the following rules LP1: uniqueness requirement of S.b (LP1: (|s.b|/|s|) <1). LP2: containment strictly holds (LP2: (|R.a|/|S.b|)>1+€).LP3: prune if either R.a or S.b belongs to floating point and Boolean data types. It is very unlikely that keys or foreign keys are defined on these data types. Where R and S are two distinct tables in a data base and a and b are two columns from R and S respectively. Candidate scoring is again based on String Similarity Function and Overall scoring function.

Meihui Zhang, MariosHadjieleftheriou, Beng Chin Ooi, CeciliaM. Procopiuc, and DiveshSrivastava [4],proposed a robust algorithm for discovering single column and multi column foreign keys. Proposes a general rule termed Randomness that subsumes a variety of other rules. Randomness is a strong indicator of the quality of an Foreign key and Primary key pairs. He Presented four definitions: D1: Randomness test: it requires two tests Domain order and Randomness measure. D2: Quantile distance, it is independent of the types of values in multi columns as long as a total ordering of the values in each dimension is defined. D3: Quantile Histogram, it is defined as the number of values of primary key within each grid cell of quantile grid. D4: Distribution histogram is the number of distinct values of foreign keys within each grid cell of the quantile grid.

Charlotte Vilarem [5], extracted foreign keys in three ways: 1. Generate foreign key candidates by unary inclusion dependencies. 2. Pruning pass over the foreign key candidate to eliminate quickly the candidates that are the farthest from being foreign keys. 3. Testing the remaining foreign key candidate. The table 1 describes the features and measures implemented in the previous work of foreign key identification.

Table 1: Foreign Key Identification Results summary

References	Features	F-measure	Run time[s]
Zhang et al. [4]	Data	1.00	501
Chen et al. [3]	Data, Metadata	1.00	14
Rostin et al. [2]	Data, Metadata	0.95	450
Jan motl et al. [1]	Metadata	0.77	1

B.Keyword Search

MayssamSayyadian, LeKhac, AnHai Doan, Luis Gravano [6] ,describes KITE a solution to the Keyword Search problem over heterogeneous relational database. Keyword search takes place in single database search and multiple data base search. KITE operates in two phases, Offline preprocessing and online query phase. In multiple data base search KITE's join discovery module needs to find all foreign key joins by the following: Finding keys in table U, Finding joinable attributes in V, Generating FK Join Candidates, removing semantically incorrect candidates. Kite scales well with multiple database search. Described how Kite applies condensed CN-Generator and the top k searcher to database to produce top-k answers to user queries.

Fang Liu, Clement Yu, WeiyiMeng, AbdurChowdhury [7],proposed a novel IR ranking strategy for effective keyword search. Given keyword query is processed in three steps 1) the system generates all answers (Tuple trees) for the query. 2)the system computes a ranking score for each answer and ranks them. 3) top-k answers are returned with semantics. It identifies and uses Four Normalizations, Tuples tree size Normalization, Document Length Normalization, Document Frequency Normalization and Inter Document Weight

Normalization. This strategy also uses phrase based and concept based model to improve search effectiveness. This approach not only can be used at the application level for keyword search in relational databases but also can be incorporated into the core of a RDBMS.

DeokminHaam, Ki Yong Lee, Myoung Ho Kim [8],proposed a keyword search method on relational database. This method finds joined tuples as answers, partition them by interpretations of the query, and rank those groups of answers. Ranking methods are needed to rank the answers and return important one first, in this method, each joined tuple, as an answer for the keyword query is ranked by the importance of its CITS (Common Interpretation Tuples set). To rank CITS introduced scoring function that considers the following, 1) Number of matched query keyword K as a ranking factor (K). 2) Number of attributes matching query keyword as a ranking factor, 3) Number of relations in CITS as a ranking factor and 4) use of affinity weight as a ranking factor.

MyintMyintThein and Mie Mie Su Thwin [9] , presented efficient schema based keyword search in relational databases. They proposed a candidate network generation algorithm to generate minimum number of joining tuples according to the maximum number of tuple sets. It works on 4 phases they are: 1) Query cleaning phase, 2) Constructing Matched tuple set phase, 3) Generating Candidate Network phase and 4) Evaluating Candidate Network phase. Candidate Network Generation consist of two algorithms namely, Heuristic CN-GEN algorithm to reduce the computation cost and memory cost and AT-GEN algorithm to improve the performance of heuristic CN-GEN algorithm.

Wei Wang, Xuemin Lin, Yi Luo [10],used SPARK to search a keyword on relational database. SPARK uses its own ranking function that considers the following factors: 1) Information retrieval relevance score, 2) Completeness of the result and 3) Size of the result. It calculates the IR relevance score seperatly for each tuple in the result. When the query is short, users prefer results that matches most if not all, of the search keyword. Completeness factor based on the extended Boolean model is designed to capture this observation. Normalization based on the size of the result is needed as a larger result is more to achieve higher IR relevance and completeness scores. These three factors multiplied together to derive the score of the result.

Table 2: Representative Keyword Search system for Relational Database.

References	Proposed Algorithms	Data Models	Distance	Conclusion Extracted
S. Agrawal et al.[14]	DBXplorer	Schema Graph	Number of joins	Keyword based search
G. Bhalotia et al. [16]	BANKS	Data Graph	Edge weights, Node Weights	Reduces the effort involved in publishing relational data on the web.
V. Hristidis et al. [15]	DISCOVER	Schema Graph	Number of joins	System that performs keyword search in relational databases.
L. Gravano et al. [17]	IR-STYLE	----	TF- IDF	Hybrid algorithm has the best overall performance.
Wei Wang et al. [10]	SPARK	Schema Graph	Number of joins, IF-IDF, Normalization	SPARK uses its own ranking Factor based on Relevance score, Completeness and Size
Haixun Wang et al. [18]	BLINKS	Data Graph	Tightness	BLANKS improves the query performance by more than an order of magnitude.

C.Inclusion Dependencies

Fabien De Marchi, St´ephane Lopes, and Jean-Marc Petit [11], proposed a new and efficient technique to discover Unary IND satisfied in a database. Data preprocessing with a new algorithm for unary IND inference, if all values of attributes A can be found in values of B then by construction B will be present in all lines of the binary relation containing A. To find all INDs used an algorithm called A level wise algorithm. They used depth first version of an algorithm.

J. Bauckmann, U. Leser and F. Naumann [12], presented efficient algorithms for finding unary INDs that works on three statements 1) Join, 2) Minus and 3) Not in. They developed two algorithms to compute inclusion dependencies outside of the database are Brute force approach to creates all IND candidates while iterating over all dependent and referenced attributes and A single pass algorithm to minimizes the amount of input or output over the sets of attribute values.

A. Koeller, E. A. Rundensteiner [13], mapped the IND discovery problem to a graph problem by using K- uniform hypergraphs. To extend K-hypergraphsthey used the clique finding problem it works on NP complete graph problem. They developed an algorithm called hyperclique that finds clique in K uniform hypergraphs, and while NP complete shows satisfactory performance for the space limitation. They introduced FIND2 algorithm to find inclusion dependencies over high dimensional databases, it applies on clique and hyperclique finding techniques.

Table 3: Inclusion Dependency Results summary

References	Title	Proposed Algorithm	Conclusion Extracted
Fabian Tschirschnitz et al.[19]	Detecting Inclusion Dependencies on Very Many tables.	MANY Algorithm	MANY creates a space efficient bit signature for each column in the input data set using Blooms filters
VerniqueTietz et al.[20]	Efficiently Detecting Inclusion Dependencies.	SPIDER	SPIDER is applicable for very large database with huge number of IND candidates.
Jana Bauckmann et al. [21]	Discovering Conditional Inclusion Dependencies.	CINDERELLA and PLI	CINDERELLA is faster than PLI but consumes more memory.
NuhadShaabani et al. [22]	Scalable Inclusion Dependency Discovery.	S-INDD	S-INDD is efficient and scalable algorithm for larger datasets to identify Unary Inclusion Dependency

III. CONCLUSION

In this paper, we have surveyed some concepts on Relational Databases are Inclusion Dependencies, Foreign Key Identification and Keyword Search. We have compared and described the various approaches for developing the above concepts. The current work on RDBMS is to integrate [23],[24],[25] few more operations [26] on security in RDBMS which is blending the RDBMS and NOSQL.

REFERENCES

[1] Jan Motl, and PavelKordík Foreign Key Constraint Identification in Relational Databases. CEUR Workshop Proceedings Vol. 1885, ISSN 1613-0073, ITAT 2017 Proceedings, pp. 106–111.2017.
[2] Alexandra Rostin, Oliver Albrecht, Jana Bauckmann, Felix Naumann, and Ulf Leser. A machine learning approach to foreign key discovery.12th Int. Work. Web Databases (WebDB), Provid. Rhode Isl., (WebDB):1–6, 2009.
[3] Zhimin Chen, VivekNarasayya, and SurajitChaudhuri. Fast Foreign-Key Detection in Microsoft SQL Server Power- Pivot for Excel. VLDB Endow., 7(13):1417–1428, 2014.
[4] Meihui Zhang, MariosHadjieleftheriou, Beng Chin Ooi, CeciliaM. Procopiuc, and DiveshSrivastava. On multi-columnforeign key discovery. Proc. VLDB Endow., 3(1-2):805–814, 2010.
[5] Charlotte Vilarem ,Approximate key and foreign key discovery in relational Database, A thesis by Charlotte Vilarem, pp. 45-53.
[6] MayssamSayyadian; LeKhac; AnHai Doan; Luis GravanoEfficient Keyword Search Across Heterogeneous Relational Databases IEEE 23rd International Conference on Data Engineeringpp: 346 – 355. 2017.

[7] Fang Liu, Clement Yu, WeiyiMeng, AbdurChowdhury, Effective Keyword Search in Relational Databases, SIGMOD 2006, June 27-29, 2006, Chicago, Illinois, USACopyright 2006 ACM 1-59593-256-9/06/0006.
[8] Deokmin Haam, Ki Yong Lee, Myoung Ho Kim, Keyword search on relational databases using keyword queryinterpretation, 5th International Conference on Computer Sciences and Convergence Information Technology , pp: 957 – 961,2010.
[9] MyintMyintThein and Mie Mie Su Thwin, EFFICIENT SCHEMA BASED KEYWORD SEARCH IN RELATIONAL DATABASES International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol.2, No.6, December 2012.
[10] Wei Wang, Xuemin Lin, Yi Luo, Keyword Search on Relational Databases2007 IFIP International Conference on Network and Parallel Computing Workshops , pp: 7 – 10,2007.
[11] Fabien De Marchi, St´ephane Lopes, and Jean-Marc Petit, Efficient Algorithmsfor Mining Inclusion Dependencies, Springer-Verlag Berlin Heidelberg 2002, LNCS 2287, pp. 464–476, 2002.
[12] J. Bauckmann; U. Leser; F. Naumann, Efficiently Computing Inclusion Dependencies for Schema Discovery22nd International Conference on Data Engineering Workshops (ICDEW'06), pp. 2-10, 2006.
[13] A. Koeller; E. A. Rundensteiner, Discovery of high-dimensional inclusion dependenciesProceedings 19th International Conference on Data Engineering (Cat. No.03CH37405) Pages: 683 – 685, 2003.
[14] S. Agrawal, S. Chaudhuri, and G. Das. DBXplorer: A system for keyword-based search over relational databases. In *ICDE-02*.
[15] V. Hristidis and Y. Papakonstantinou, *Discover: keyword search in relational databases*, Proceedings of the 28th international conference on Very Large Data Bases, p. 670-681, 2002.
[16] G. Bhalotia, A. Hulgery, C. Nakhe, S. Chakrabarti, and S. Sudarshan, *Keyword Searching and Browsing in Databases using BANKS*, 18thInternational Conference on Data Engineering, p. 431, 2002.
[17] V. Hristidis, L. Gravano, and Y. Papakonstantinou, *Efficient IR-style keyword search over relational databases*, Proceedings of the 29th international conference on Very large data bases, p. 850-861, 2003.
[18] Hao He, Haixun Wang, Jun Yang, and Philip S. Yu, BLINKS: ranked keyword searches on graphs, Proceedings of the 2007 ACM SIGMOD international conference on Management of data, p. 305-316, June 11-14, 2007.
[19] Fabian Tschirschnitz, Thorsten Papenbrock, and Felix naumann: Detecting Inclusion Dependencies on Very Many Tables, ACM Transactions on Database Systems, Vol. 42, No. 3, Article 18. July 2017.
[20] Jana Bauckmann, Ulf Leser, FelixNaumann, V´eroniqueTietz: Efficiently Detecting Inclusion Dependencies, IEEE 23rd International Conference on Data Engineering, 04 June 2007.
[21] Jana Bauckmann, ZiawaschAbedjan, Ulf Leser, Heiko Müller, Felix Naumann: Discovering Conditional Inclusion Dependencies, Maui, HI, USA, CIKM'12, October 29–November 2, 2012.
[22] Nuhad Shaabani, Christoph Meinel: Scalable Inclusion Dependency Discovery, Springer International Publishing Switzerland, pp 425-440 ,2015.
[23] AA Chikkamannur, S Handigund: A mediocre approach to syndicate the attributes for a class or relation, International Journal of Software Engineering and Its Applications 5 (4), pp 99-106, 2011.
[24] AA Chikkamannur, SM Handigund: Design of Normalized Relation: An Ameliorated Tool, International Journal of Computing & Information Sciences 9 (1), pp 28-33, 2011.
[25] AA Chikkamannur, SM Handigund: An ameliorated methodology to design normalized relations, IEEE/ACS International Conference on Computer Systems and Applications, AICCSA 2009, pp 861-864.
[26] A Chikkamannur, S Handigund: A Concoct Semiotic for Recursion in SQL: International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), pp 204-210, 2013.