ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 6, Issue 1, January 2017

A Survey on Different Techniques of Site Data Distribution with Privacy

M.Tech. Scholar Nisha Sahu, Astt. Prof. Shivendra Dubey

Abstract— **As digital world is increasing day by day so thousands of session are insert, update or delete in each second. So maintains of database at different servers required highly relational model. For each new kind of data relations can be found by functional dependency. So finding new or updating the existent functional dependency is highly reuired. This paper give a detail survey of various techniques developed and implement on different databases. This paper has provide major issues that need to cover in this field as well. This survey would promote a lot of research in the area of mining functional dependencies from data.**

Index Terms— **Conditional Functional Dependency, Data Anonymization, Effective Pruning, Mining, Similarity Constraints.**

I. INTRODUCTION

As in today's era all kind of data are available on servers which include valuable information of companies, government organizations, etc. Some time data include customer details there patterns of purchasing, vendor details, etc. which are very sensitive for company. So possibility of having all data at same server is high, which vulnerable in case of attacks.

So management of such controlled centralized system is desired to be secured, but all such information at one place is not a wise step for storage. For example, if the single site goes down, then everyone is blocked from accessing the databases until the site comes back up again. Also the communications costs from the many far PCs and terminals to the central site can be expensive. One solution to such problems, and an alternative design to the centralized database concept, is known as distributed database.

In data-mining, privacy preserving data-mining is the most innovative fields of research where the algorithms for data-mining are evaluated for the adverse effects occurred in privacy of data. The basic aspect in the privacy-preserving data-mining is dual. In first case, the private raw data such as gender, identifiers, addresses, religion and something similar to these must be altered or removed from the actual database, as per the data receiver, which are not be capable to adjust the security of private data of other person. Whereas in the second case, the private data which may be retrieved from the database through using algorithms for data-mining must also be prohibited, as these patterns of information can be equally well compromise data privacy. To introduce various algorithms for altering the basic data in few fashion, this is the major target in the privacy-preserving data-mining is such that the private data and private-information are still retained as private though after the mining process. As when the private information may be retrieved from the released data by the legal users then this issue have arises, which is also usually termed as the "database inference" issue.

When a person or process any here on the distributed network queries the database, it is not necessary to know where on the network the data being sough t is located. The user just issues the query, and the result is returned. This feature is known as location transparency. This can become rather complex very quickly, and it must be managed by sophisticated software.

As concerned with data placement policies that distribute data in a way that is advantageous for application or workflow execution, for example, by placing data sets near high-performance computing resources so that they can be staged into computations efficiently; by moving data off computational resources quickly when computation is complete; and by replicating data sets for performance and reliability. Effective data placement policies of this type might benefit from knowledge about available resources and their current performance and capacity. Placement services could also make use of hints or information about applications and their access patterns, for example, whether a set of files is likely to be accessed together and therefore should be replicated together on storage systems.

II. Dependencies and Its Types

Dependencies aremetadata that describe relationships among columns. The difficulties of automatically detecting such dependencies in a given dataset are twofold: First, pairs of columns or larger column setsmust be examined, and second, the chance existence of a dependency in the data at hand does not imply that this dependency is meaningful. While much research has been invested in addressing the first challenge and is the focus of this survey, there is less work on semantically interpreting the profiling results.

A common goal of data profiling is to identify suitable keys for a given table. Thus, the discovery of *unique column combinations*, i.e., sets of columns whose values uniquely identify rows, is an important data profiling task [7]. Once unique column combinations have been discovered, a second

Manuscript received Dec, 2016.
Nisha Sahu, *CSE, RadhaRaman Engineering College, RGTU, Bhopal, India, 9893236696*
Shivendra Dubey, *CSE, RadhaRaman Engineering College, RGTU, Bhopal, India, 9713183524*.

step is to identify among them the intended primary key of a relation.

A frequent real-world use-case of multi-column profiling is the discovery of foreign keys [2] with the help of inclusion dependencies. An inclusion dependency states that all values or value combinations from one set of columns also appear in the other set of columns—a prerequisite for a foreign key.

Another form of dependency that is also relevant for data quality is the functional dependency (Fd). A functional dependency states that values in one set of columns functionally determine the value of another column. Again, much research has been performed to automatically detect Fds [1].

Dependencies have many applications: An obvious use case for functional dependencies is schema normalization. Inclusion dependencies can suggest how to join two relations, possibly across data sources. Their conditional counterparts help explore the data by focusing on certain parts of the dataset.

Conditional, partial, and approximate solutions

Real datasets usually contain exceptions to rules. To account for this, dependencies and other constraints detected by data profiling can be relaxed. This work describe two relaxations below: partial and approximate.

Partial dependencies hold for only a subset of the records, for instance, for 95% of the records or for all but 10 records. Such dependencies are especially valuable in data cleansing scenarios: They are patterns that hold for almost all records and thus should probably hold for *all* records if the data were clean. Violating records can be extracted and cleansed [129]. Once a partial dependency has been detected, it is interesting to characterize for which records it holds, i.e., if a condition that selects precisely those records.

Conditional dependencies can specify such conditions. For instance, a conditional unique column combination might state that the column street is unique for all records with city = 'NY.' Conditional inclusion dependencies (Cinds) were proposed by Bravo et al. for data cleaning and contextual schema matching. Conditional functional dependencies (Cfds) were introduced in [3], also for data cleaning.

Approximate dependencies and other constraints are unconditional statements, but are not guaranteed to hold for the entire relation. Such dependencies are often discovered using sampling or other summarization techniques [3]. Their approximate nature is often sufficient for certain tasks, and approximate dependencies can be used as input to the more rigorous task of detecting true dependencies. This survey does not discuss such approximation techniques.

III. Related Work

YKA HUHTALA et. al. [4] presented TANE, an efficient algorithm for finding functional dependencies from large databases. It is based on portioning the set of rows with

Table. 1 Represent Comparison of various techniques.

Title	Approach	Merit	Demerit	Dataset
Privacy-Preserving Mining of Association Rules from Outsourced Transaction Databases [14].	Add Fake transactions	Perturb and De-Perturb Dataset.	As the addition of fake transaction in dataset for perturbation size of dataset increases.	100
A Methodology for Direct and Indirect Discrimination Prevention in Data Mining [15].	Perturbation of non sensitive information	Hide all sensitive information.	Work on binary attributes only. Execution time is high.	99.7
An Efficient Algorithm for Privacy Preserving Data Mining Using Heuristic Approach [16].	Heuristic Approach	Selected Item can be suppressed.	Privacy of all specified item is not sufficient, as some information remain even after perturbation.	89
Enabling Multilevel Trust in Privacy Preserving Data Mining [17].	Noise addition	Provide Privacy for multi level users.	Only numeric data is suppress. No measures taken for textual data.	85.5

All Rights Reserved © 2017 IJARCET

respect to their attribute values, which makes testing the validity of functional dependencies fast even for large number of tuples. The use this approach also focuses on the discovery of approximate functional dependencies easy and efficient. An approximate functional dependency is a functional dependency that almost holds. For example speaking language
is approximately determined by nationality. They also claim this method is erroneous and exceptional rows can be identified easily. This technique shown fast in practice during experimentations. But this work did not mention search criteria during traversal of tuples for discovery of FDs.

Previous work by Yka Huhtala et. al. in [5], explained discovery of functional and approximate dependencies using partitions. They presented this approach for finding functional
dependencies from large databases, based on partitioning mostly similar their above work.

The author concept in this paper is Privacy Preserving mining of frequent patterns on encrypted outsourced Transaction Database (TDB) [18]. They proposed a encryption scheme and adding fake transaction in the original dataset. Their method proposed a strategy for incremental appends and dropping of old transaction batches and decrypt dataset. They also analyze the crack probability for transactions and patterns. The Encryption/Decryption (E/D) module encrypts the TDB once

which is sent to the server. Mining is conducted repeatedly at the server side and decrypted every time by the E/D [18] module. Thus, we need to compare the decryption time with the time of directly executing a priori over the original database.

A classification of privacy preserving techniques is presented and major algorithms in each class is surveyed. The merits and demerits of different techniques were pointed out [19]. The algorithms for hiding sensitive association rules like privacy preserving rule mining using genetic algorithm.

This authors [20] presents a survey of different association rule mining techniques for market basket analysis, highlighting strengths of different association rule mining techniques. As well as challenging issues need to be addressed by an association rule mining technique. The results of this evaluation will help decision maker for making important decisions for association analysis.

Hajian [21] present dithered B-tree, a B-tree index structure that can serve as a building block for realizing efficient system implementations in the area of secure and private database outsourcing. The dithered tree insert algorithm [21] can be further optimized to incur only one traversal from the root to the leaf, instead of two. The index structure from learning whether or not the search term (i.e., key) is present in the database and check the data for secure and private database outsourcing.

IV. Secure Multiparty Computation

The security and cryptography communities have set standards for what it means to provably maintain privacy and security. One concept that is particularly relevant to this proposal is Secure Multiparty Computation, introduced in [4]. The basic idea of Secure Multiparty Computation is that a computation is secure if at the end of the computation, no party knows anything except its own input and the results. One way to
view this is to imagine a trusted third party – everyone gives their input to the trusted party, who performs the computation and sends the results to the participants. Now imagine we can achieve the same result without having a trusted party. Obviously, some communication between the parties is required for any interesting computation – how do we ensure that this communication doesn't disclose anything? The answer is to allow non-determinism in the exact values sent in the intermediate communication (e.g., encrypt with a randomly chosen key), and prove that a party using its own input and the result can generate a "predicted" intermediate computation that is as likely as the actual values.

There has been work in cooperative computation between entities that mutually distrust one another. This computation may be of any sort: scientific, data processing or even secret sharing. Secure two party computation was first investigated by Yao [7] and was later generalized to multiparty computation. The seminal paper by Goldreich proves that there exists a secure solution for any functionality[11]. The approach used is as follows: the function F to be computed is first represented as a combinatorial circuit, and then the parties run a short protocol for every gate in the circuit. Every participant gets random shares of the input and output wires for every gate. This approach, though appealing in its generality and simplicity, means that the number of rounds of the protocol grow with the size of the circuit. This grows with the size of the input. This is highly inefficient for large inputs, as in data mining. Although this proves secure solutions
exist, achieving efficient secure solutions for distributed data mining is still open.

Secure Multiparty Computation makes two key contributions to the proposed work:
1. Methods for securely computing functions with small inputs (e.g., secure comparison).
2. Definitions and proof techniques for private and secure computations in a distributed environment.

V. Privacy Preserving Techniques

Use of personal data for any activity without any information to the concern is term as Public concern. In order to understand this thing consider a user never want to share there personal information with other without his permission, but it is done by the information holding organization, then this is called as public concern. Mostly it is divide into few category such as bank, health care departments are most trustful for customer information privacy. While in case of credit card company, some of social site they are least trust companies.

Data Swapping

Data swapping techniques mainly appeal of the method was it keeps all original values in the data set, at the same time the record re-identification is very difficult [1]. Data swapping means replaces the original data set by another one.

Here some original values belonging to a sensitive attribute are exchanged between them. This swapping can be done in a way so that the t-order statistics of the original data set are preserved. A t-order statistic is a statistic that can be generated from exactly t attributes. A new concept called approximate data swap was introduced for practical data swapping. It computes the t-order frequency table from the original data set, and finds a new data set with approximately the same t-order frequency. The elements of the new data set are generated one at a time from a probability distribution constructed through the frequency table. The frequency of already created elements and a possible new element is used in the construction of the probability distribution. Inspired by existing data swapping techniques used for statistical databases a new data swapping technique has been introduced for privacy preserving data mining, where the requirement of preserving t-order statistics has been relaxed. The technique emphasizes the pattern preservation instead of obtaining unbiased statistical parameters [2]. It preserves the most classification rules and also obtained different classification algorithms. As the class is typically a categorical attribute containing just two different values, the noise is added by changing the class in a small number of records. It can be achieved by randomly shuffling the class attributes values belonging to heterogeneous leaves of a decision tree.

Horizontally partitioned Dataset

In the data-sets which are horizontally-partitioned data, have several set of the records along with the same type of sets of attributes that are in used for the purposes of mining. A case of horizontally-partitioned is discussed, where the privacy-preserving classification is done in fully shared setting, in which every individual have secure accesses to their private record only. A host system of the other applications of data-mining has been concluded to issue of the horizontally-partitioned data-sets. Many applications of data-mining may be done that is clustering, filtering and the association-rule mining.

Vertical partitioned Dataset

The vertically-partitioned dataset [9] have several operations like calculating the scalar product or intersection of safe set size may be useful in the calculation of results of the algorithms of data-mining. Vertically-partitioned data performed linear regressions without sharing their data values. The method of the vertically-partitioned data may be enhanced to the variety of the applications of data-mining i.e. decision trees, k means clustering, Naïve Bayes Classifier and SVM Classification.

VI. Conclusions

As researchers are working on different field out of which finding an effective functional dependency is measure issue with this growing digital world. This paper has give a detailed discussion of the various approaches done by different researchers. Various measure issue are still need to be cover in finding functional dependencies which is open challenge for researcher. One of the major issue is finding an automatic dependency in encrypted data. This survey would promote a lot of research in the area of mining functional dependencies from data.

Highlight a section that you want to designate with a certain style, and then select the appropriate name on the style menu. The style will adjust your fonts and line spacing. **Do not change the font sizes or line spacing to squeeze more text into a limited number of pages.** Use italics for emphasis; do not underline.

REFERENCES

1] Abedjan, Z., Grütze, T., Jentzsch, A., Naumann, F.: Mining and profiling RDF data with ProLOD++. In: Proceedings of the International Conference on Data Engineering (ICDE), pp. 1198–1201(2014).

[2] Rostin, A., Albrecht, O., Bauckmann, J., Naumann, F., Leser,U.: A machine learning approach to foreign key discovery. In: Proceedings of the ACM SIGMOD Workshop on the Web and Databases (WebDB) (2009), pp: 40-49.

[3] Thorsten Papenbrock, Jens Ehrlich, Jannik Marten, Tommy Neubert, Jan-Peer Rudolph, Martin Schonberg, Jakob Zwiener and Felix Naumann, "Functional Dependency Discovery: An Experimental Evaluation of Seven Algorithms", Proceedings of VLDB 2015, pp: 1082-1093.

[4] Huhtala, Y., Karkkainen, J., Porkka, P., and Toivonen, H., (1999), TANE: An Efficient Algorithm for discovering Functional and Approximate Dependencies, The Computer Journal, V.42, No.20, pp.100-107.

[5] Huhtala, Y., Karkkainen, J., Porkka, P., and Toivonen, Dependencies Using Partitions, IEEE ICDE 1998, pp: 392-401.

[6] Shyue-liang Wang, Jenn-Shing Tsai and Been-Chian Chien, "Mining Approximate Dependencies Using Partitions on Similarity-relation-based Fuzzy Databases", IEEE International Conference on Systems, Man and Cybernetics(SMC) 1999, pp: 1-8.

[7] Yao, H., Hamilton, H., and Butz, C., FD_Mine: Discovering Functional dependencies in a Database Using Equivalences, Canada, IEEE ICDM 2002, pp: 1-15.

[8] Wyss. C., Giannella, C., and Robertson, E. (2001), FastFDs: A Heuristic-Driven, Depth-First Algorithm for Mining Functional Dependencies from Relation Instances, Springer Berlin Heidelberg 2001, pp:1-22.

[9] Russell, Stuart J. and Norvig, Peter. Arti cial Intelligence: A ModernApproach. Prentice Hall, 1995, pp: 260-372.

[10] A. C. Yao, "How to generate and exchange secrets," in Proceedings of the 27th IEEE Symposium on Foundations of Computer Science. IEEE, 1986, pp. 162–167.

[11] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game - a completeness theorem for protocols with honest majority," in 19th ACM Symposium on the Theory of Computing, 1987, pp. 218–229.

[12] Heikki Mannila and Kari-Jouko R¨aih¨a. Design by example: An application of Armstrong relations. Journal of Computer and System Sciences, 33(2): 2015, pp:126-141.

[13] Wenfei Fan, Jianzhong Li, Nan Tang, And Wenyuan Y. "Incremental Detection Of Inconsistencies In Distributed Data". Ieee Transactions On Knowledge And Data Engineering, Vol. 26, No. 6, June 2014 pp: 1367-1378.

[14] Sara Hajian and Josep Domingo-Ferrer. "A Methodology for Direct and Indirect Discrimination Prevention in Data Mining". *IEEE transaction on knowledge data engineering, VOL. 25, NO. 7,* JULY 2013, pp.1445-1459.

[15] Sara Hajian and Josep Domingo-Ferrer "A Methodology for Direct and Indirect Discrimination Prevention in Data Mining" *IEEE Transactions On Knowledge And Data Engineering*, VOL. 25, NO. 7, JULY 2013, pp. 1-16.

All Rights Reserved © 2017 IJARCET

[16] Yaping Li, Minghua Chen, Qiwei Li, and Wei Zhang. "Enabling Multilevel Trust in Privacy Preserving Data Mining" IEEE transaction on knowledge data engineering, VOL. 24, NO. 9, September 2012, , pp. 1598-1612.

[17] D. Pedreschi, S. Ruggieri, and F. Turini, "Discrimination-Aware Data Mining," *Proc. 14th ACM Int'l Conf. Knowledge Discovery and Data Mining (KDD '08)*, 2008, pp. 560-568.

[18] Pedreschi, D., Ruggieri, S. & Turini, F. (2008). Discrimination-aware data mining. Proc. of the 14th ACM International Conference on Knowledge Discovery and Data Mining (KDD 2008), pp. 560-568.

[19] Hajian, S., Domingo-Ferrer, J. & Martinez-Ballesté, A. (2011a). Discrimination prevention in data mining for intrusion and crime detection. Proc. of the IEEE Symposium on Computational Intelligence in Cyber Security (CICS 2011), pp. 47-54.

[20] Verykios, V. & Gkoulalas-Divanis, A. (2008). A survey of association rule hiding methods for privacy. In C. C. Aggarwal and P. S. Yu (Eds.), Privacy- Preserving Data Mining: Models and Algorithms. Springer.

[21] M.Mahendran, Dr.R.Sugumar, K.Anbazhagan, R.Natarajan. "An Efficient Algorithm for Privacy Preserving Data Mining Using Heuristic Approach". International Journal of Advanced Research in Computer and Communication Engineering Vol. 1, Issue 9, November 2012, page 737-744.

[22] I. Erlich, G. K. Venayagamoorthy, and W. Nakawiro, " A mean-variance optimization algorithm," In Proc.IEEE World Congress on Computational Intelligence, Barcelona, Spain. July 2010, pp. 18-23.

[23] D. Pedreschi, S. Ruggieri and F. Turini, "Measuring discrimination in socially-sensitive decision records," *SDM* 2009, *SIAM*, 2009, pp. 581-592.