M-Partition Privacy Scheme to Anonymizing Set-Valued Data

D.Vanathi
Associate Professor , Computer Science and Engineering,
Nandha Engineering College, Erode, India.

P. Sengottuvelan
Associate Professor , Information Technology,
Bannari Amman Institute of Technology, Erode, India.

Abstract: In distributed databases there is an increasing need for sharing data that contain personal information. The existing system presented collaborative data publishing problem for anonymizing horizontally partitioned data at multiple data providers. M-privacy guarantees that anonymized data satisfies a given privacy constraint against any group of up to m colluding data providers. The heuristic algorithms exploiting monotonicity of privacy constraints for efficient checking of m-privacy for given group of records. The data provider-aware anonymization algorithm with adaptive m-privacy checking strategies to ensure high utility. But A new type of "insider attack" occurred by colluding data providers.The proposed system m-partition privacy scheme to anonymizing set-valued data. This scales linearly with input size. The Scores well on an information-loss data quality metric applied to anonymize AOL query logs. To Divide-and-conquer techniques used in addressing data with multiple dimensions. The M-partition privacy algorithm is in a top-down manner by recursively separating set-valued data into groups where data in each partition share a generalized representation. The analysis were conducted on different Environment to measure the performance in terms of Loss of information, computational time and no of data provider.

Keywords: Privacy, security, integrity, and protection, distributed databases.

I. INTRODUCTION

Data Mining is also called knowledge discovery in databases (KDD). Data mining is about finding new information in a lot of data. The information obtained from data mining is hopefully both new and useful. The data is saved with a goal. For example, a store wants to save what has been bought. They want to do this to know how much they should buy themselves, to have enough to sell later. Saving this information, makes a lot of data. The data is usually saved in a database. The actual data mining task is the automatic or semi-automatic analysis of large quantities of data to extract previously unknown interesting patterns such as groups of data records (cluster analysis), unusual records (anomaly detection) and dependencies (association rule mining). This usually involves using database techniques such as spatial indices.

Privacy preserving data analysis, and data publishing have received considerable attention in recent years as promising approaches for sharing data while preserving individual privacy. In a non-interactive model, a data provider (e.g., hospital) publishes a "sanitized" version of the data, simultaneously providing utility for data users (e.g., researchers), and privacy protection for the individuals represented in the data (e.g., patients). When data are gathered from multiple data providers or data owners, two main settings are used for anonymization . One approach is for each provider to anonymize the data independently which results in potential loss of integrated data utility. A more desirable approach is *collaborative data publishing* , which anonymizes data from all providers as if they would come from one source , using either a trusted third-party (TTP) or Secure Multi-party Computation (SMC) protocols.

The address the issue of privacy preserving data mining specifically,then consider a scenario in which two parties owning confidential databases wish to run a data mining algorithm on the union of their databases, without revealing any unnecessary information. Our work is motivated by the need to both protect privileged information and enable its use for research or other purposes. The above problem is a specific example of secure multi-party computation

and as such, can be solved using known generic protocols. However, data mining algorithms are typically complex and, furthermore, the input usually consists of massive data sets. The generic protocols in such a case are of no practical use and therefore more efficient protocols are required. This focus on the problem of decision tree learning with the popular ID3 algorithm. Our protocol is considerably more efficient than generic solutions and demands both very few rounds of communication and reasonable bandwidth.

II. RELATED WORK

A simple privacy-preserving reformulation [1] of a linear program whose equality constraint matrix is partitioned into groups of rows. Each group of matrix rows and its corresponding right hand side vector are owned by a distinct private entity that is unwilling to share or make public its row group or right hand side vector. By multiplying each privately held constraint group by an appropriately generated and privately held random matrix, the original linear program is transformed into an equivalent one that does not reveal any of the privately held data or make it public. The solution vector of the transformed secure linear program is publicly generated and is available to all entities. privacy-preserving classification and data mining, wherein the data to be classified or mined is owned by different entities that are unwilling to reveal the data they hold or make it public, has spread to the field of optimization and in particular linear programming. In a number of shortcomings in the privacy-preserving linear programming literature are pointed out. In a method for handling privately held vertical partitions of a linear programming constraint matrix and cost vector is proposed that is based on private random transformations of the corresponding problem variables.

The BIRCH algorithm [2] is a well known algorithm for clustering for effectively computing clusters in a large data set. As the data is typically distributed over several sites, clustering over distributed data is an important problem. The data can be distributed in horizontal, vertical or arbitrarily partitioned databases. But, because of privacy issues no party may share its data to other parties. The problem is how the parties can cluster the distributed data without breaching privacy of others data. The solutions in arbitrarily partitioned database setting generally work for both horizontal and vertically partitioned databases. It give a procedure for securely running BIRCH algorithm over arbitrarily partitioned database. Introduce secure protocols for distance metrics and give a procedure for using these metrics in securely computing clusters over arbitrarily partitioned database.

The Privacy preserving [3] Data mining has been a popular research area for more than a decade due to its vast spectrum of applications. The aim of privacy preserving data mining researchers is to develop data mining techniques that could be applied on databases without violating the privacy of individuals. This work propose methods for constructing the dissimilarity matrix of objects from different sites in a privacy preserving manner which can be used for privacy preserving clustering as well as database joins, record linkage and other operations that require pair-wise comparison of individual private data objects horizontally distributed to multiple sites.

ID3 Algorithm [4] describes, Privacy and security concerns can prevent sharing of data, derailing data mining projects. Introduce a generalized privacy preserving variant of the ID3 algorithm for vertically partitioned data distributed over two or more parties. Along with the algorithm, it give a complete proof of security that gives a tight bound on the information revealed. While this has been done for horizontally partitioned data. It present an algorithm for vertically partitioned data: a portion of each instance is present at each site, but no site contains complete information for any instance. This problem has been addressed, but the solution is limited to the case where both parties have the class attribute.

In Bayesian Network construct [5] describes traditionally, many data mining techniques have been designed in the centralized model in which all data is collected and available in one central site. Different parties often wish to benefit from cooperative use of their data, but privacy regulations and other privacy concerns may prevent the parties from sharing their data. Privacy-preserving data mining provides a solution by creating distributed data mining algorithms in which the underlying data need not be revealed. Two parties owning confidential databases wish to learn the Bayesian network on the combination of their databases without revealing anything else about their data to each other. This present an efficient and privacy-preserving protocol to construct a Bayesian network on the parties' joint data.

In Location privacy [6] describes this is an important concern in participatory sensing applications, where users can both contribute valuable information (data reporting) as well as retrieve (location dependent) information (query)

regarding their surroundings. K-anonymity is an important measure for privacy to prevent the disclosure of personal data. It propose a mechanism based on locality-sensitive hashing (LSH) to partition user locations into groups each containing at least K users (called spatial cloaks). The mechanism is shown to preserve both locality and K-anonymity.Then devise an efficient algorithm to answer Knn queries for any point in the spatial cloaks of arbitrary polygonal shape. Extensive simulation study shows that both algorithms have superior performance with moderate computation complexity.

The anonymization of sensitive micro data [7] (e.g. medical health records) is a widely-studied topic in the research community. A still unsolved problem is the limited informative value of anonymized micro data that often rules out further processing (e.g. statistical analysis). Thus, a tradeoff between anonymity and data precision has to be made, resulting in the release of partially anonymized micro data sets that still can contain sensitive information and have to be protected against unrestricted disclosure. Anonymization is often driven by the concept of k-anonymity that allows fine-grained control of the anonymization level.It present an algorithm for creating unique fingerprints of micro data sets that were partially anonymized with k-anonymity techniques.It show that it is possible to create different versions of partially anonymized micro data sets that share very similar levels of anonymity and data precision, but still can be uniquely identified by a robust fingerprint that is based on the anonymization process.

In Privacy Preserving [8] the collaborative data publishing problem for anonymizing horizontally partitioned data at multiple data providers. Consider a new type of "insider attack" by colluding data providers who may use their own data records (a subset of the overall data) in addition to the external background knowledge to infer the data records contributed by other data providers.First, introduce the notion of m-privacy, which guarantees that the anonymized data satisfies a given privacy constraint against any group of up to m colluding data providers. Second,it present heuristic algorithms exploiting the equivalence group monotonicity of privacy constraints and adaptive ordering techniques for efficiently checking m-privacy given a set of records.Finally, present a data provider-aware anonymization algorithm with adaptive m- privacy checking strategies to ensure high utility and m-privacy of anonymized data with efficiency.

In k-anonymity [9] provides a measure of privacy protection by preventing re-identification of data to less than a group of k data items. While algorithms exist for producing k-anonymous data, the model has been that of a single source wanting to publish data. Due to privacy issues, it is common that data from different sites cannot be shared directly. Therefore, this presents a two-party framework along with an application that generates k-anonymous data from two vertically partitioned sources without disclosing data from one site to the other. The framework. is privacy preserving in the sense that it satisfies the secure definition commonly defined in the literature of Secure Multiparty Computation. Many techniques have been proposed to protect privacy, such as data perturbation, query restriction, data swapping, Secure Multiparty Computation (SMC) etc.

In k-Anonymity protection [10] Consider a data holder, such as a hospital or a bank, that has a privately held collection of person-specific, field structured data. Suppose the data holder wants to share a version of the data with researchers. How can a data holder release a version of its private data with scientific guarantees that the individuals who are the subjects of the data cannot be re-identified while the data remain practically useful? The solution provided that includes a formal protection model named *k*-anonymity and a set of accompanying policies for deployment. A release provides *k*-anonymity protection if the information for each person contained in the release cannot be distinguished from at least *k*-1 individuals whose information also appears in the release. This also examines re-identification attacks that can be realized on releases that adhere to *k* anonymity unless accompanying policies are respected. The *k*-anonymity protection model is important because it forms the basis on which the real-world systems known as Data fly, m-Argus and *k*-Similar provide guarantees of privacy protection.

III. ARCHITECTURE DIAGRAM OF M-PARTITION PRIVACY SCHEME TO ANONYMIZING SET-VALUED DATA

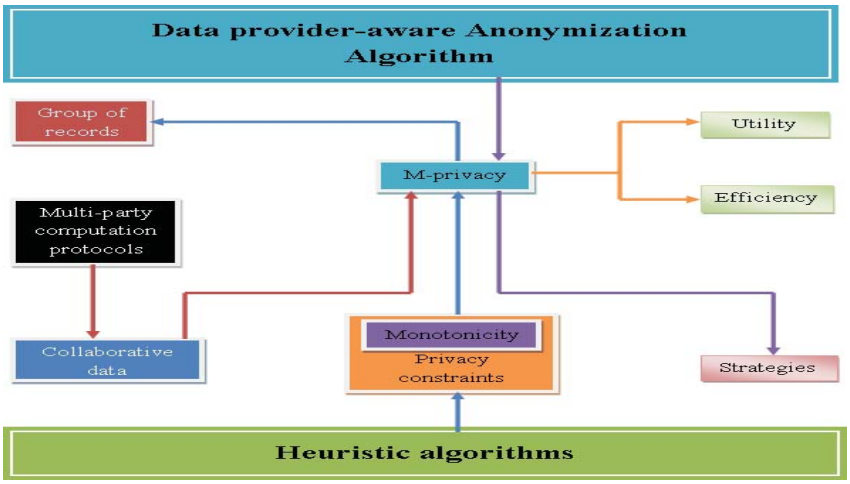

Figure 1. Architecture diagram of m-partition privacy scheme to anonymizing set-valued data

The phases involved in the proposed schemes are

- Anonymization for M-Privacy
- K-Anonymity in Set Valued Data
- Partition based Anonymization

A.Anonymization for M-Privacy

The baseline algorithm utilizes a data provider-aware algorithm with adaptive verification strategies to ensure high utility and *m*-privacy for anonymized data. The SMC implements the m privacy anonymization in a distributed environment while preserving security. For a privacy constraint C that is generalization monotonic m-privacy with respect to. C is generalization monotonic**.** Most existing generalization-based anonymization algorithms are modified to guarantee m-privacy with respect to. C.

The Adoption is straightforward every time a set of records is tested for privacy fulfillment check m-privacy with respect to. C. The Binary Space Partitioning (BSP) recursively chooses an attribute to split data points in multidimensional domain space until data cannot be split any further without breaching *m*-privacy with respect to. C.

The Features of BSP takes into account the data provider as an additional dimension for splitting uses privacy fitness score as a general scoring metric for selecting the split point. It adapts its *m*-privacy checking strategy for efficient verification.

B. K- Anonymity is Set Valued Data

The K- Anonymity is set valued data privacy model consider Let I = {I1, I2... IlIl} be the set of items from which elements of the sets are drawn and Let D = {t1, t2... tlDl} be a transactional database over I where each transaction ti within D is a non-empty subset of I. The Equivalence class in transactional database D consists of a multi set of transactions. An equivalence class for D is the set of all transactions with identical sets of items S.

The k-anonymity in set-valued data transactional database D is k-anonymous if every transaction in D occurs at least k times, or equivalently the size of each equivalence class in D is at least k. The Transactional database is k-anonymous if each transaction is identical to at least k − 1 others. The states that given any m or fewer items chosen from any transaction there are at least k−1 other transactions containing same set of m items.

The km-anonymity only protects individuals' privacy when adversary knows m or fewer items whereas k-anonymity, with the absence of parameter m, requires no limit on number of items adversary can know smaller the m in km-anonymity and weaker privacy km-anonymity provides When m = Mmax. Mmax is the maximum length of transaction.

C.Partition based Anonymization

The Partition based anonymization recursively separating set-valued data into groups where data in each partition share a generalized representation. In Mondrian anonymization algorithm generalization hierarchy has to be used in deciding which transactions are similar be grouped together.

The partition based anonymization algorithm starts by generalizing all transactions to root level in the hierarchy. The starting point always produces a trivial anonymization with one partition, as long as there are at least k transactions in the database**.** All transactions share same representation ("ALL") after being generalized to the root

The Pass the initial partition to the anonymize routine which splits the current partition into sub-partitions recursively invokes anonymize on all resulting sub partitions. The partitioning process terminates when no further split is possible

IV. PERFORMANCE EVALUATION

In this section , evaluate performance of M-partition privacy scheme to anonymizing set-valued data through J2EE. To confirm the analytical results, we implemented M- Partition privacy scheme in the sensor different analysis and evaluated M-PP the performance of technique. The performance of is evaluated by the following metrics.

- Loss of Information
- Computational Time
- No of Data Provider

Table 1.Loss of Information

Number of records	Loss of information in Existing System	Loss of Information in Proposed System
5	56	49
10	65	59
15	76	70
20	82	75
25	89	83

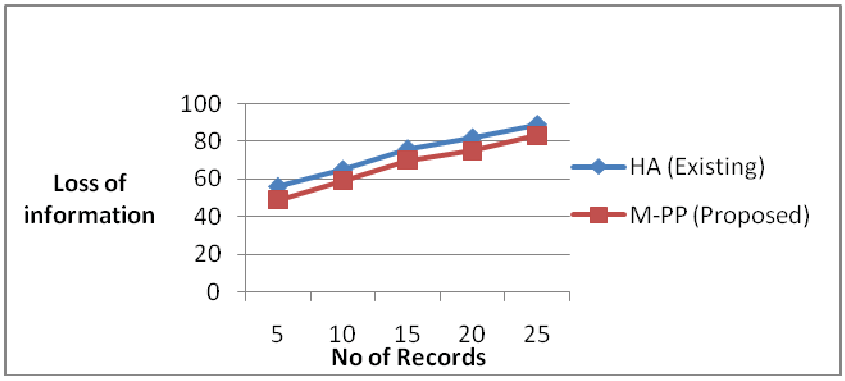

Figure 4.1. Loss of Information

Figure 4.1 demonstrates the Loss of information. X axis represents the number of Records whereas Y axis denotes the Loss of information using both the proposed M- Partition Privacy scheme. When the number of records increased, Loss of information gets decreases accordingly. The rate of Loss of information is illustrated using the existing Heuristic and proposed M-Partition Privacy scheme. Figure 4.1shows better performance of proposed M-PP in terms of No of records than existing and proposed M-PP. M- partition Privacy achieves 15 to 25% less Loss of information rate variation when compared with existing system.

Table 2. Computational Time

Number of records	Computational Time in Existing System	Computational Time in Proposed System
5	42	36
10	46	38
15	48	40
20	52	44
25	54	49

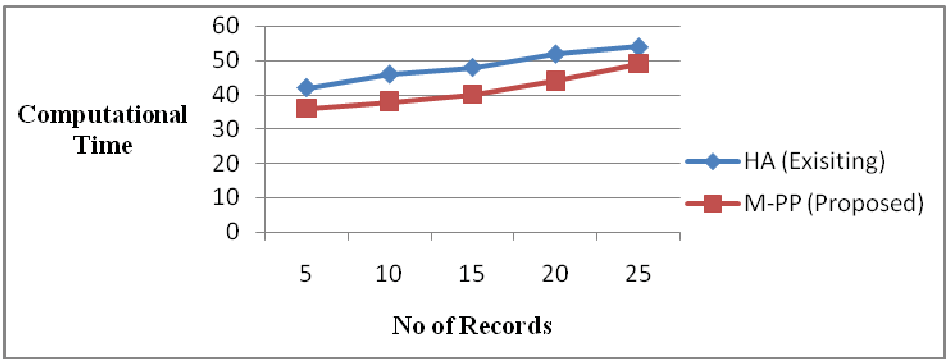

Figure 4.2. Computational Time

Figure 4.2 demonstrates the Computational Time. X axis represents the number of records whereas Y axis denotes Computational time the using both the HA and our proposed M-PP. When the number of records increased, Computational Time also gets increases accordingly. The Computational Time is illustrated using the existing HA and our proposed M- Partition privacy. Figure 4.2 shows better performance of Proposed M-Partition privacy in terms of records than existing HA and our proposed M-PP. M- Partition Privacy 20 to 35% less Computational Time variation when compared with existing system.

Table 3. No of Data Providers

Number of records	No of data providers in Existing System	No of data providers in Proposed System
5	98	86
10	87	75
15	76	63
20	65	53
25	54	42

Figure 4.3 demonstrates the No of data provider. X axis represents number of records whereas Y axis denotes the No data provider using both the HA and our proposed M-PP Technique. When the number of records increases the no of data provider also gets increased.

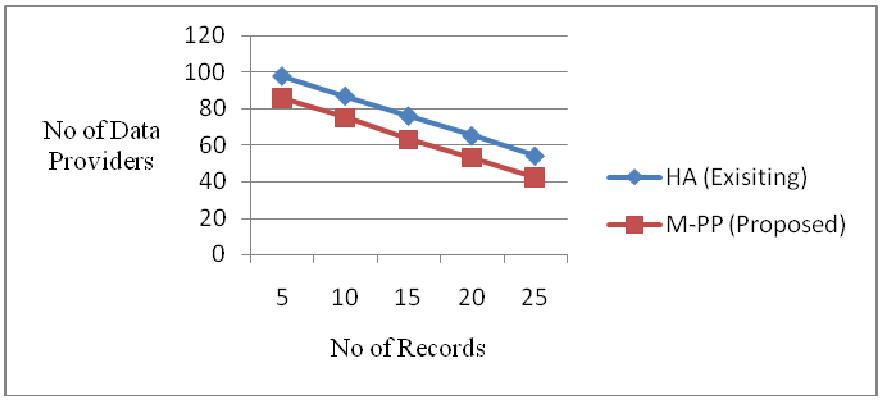

Figure 4.3. No of Data Provider

Figure 4.3 shows the effectiveness of No of Data provider over different number of records than existing HA and our proposed M-PP. M- Partition Privacy achieves 30% to 50% more No of Data provider when compared with existing schemes.

V. CONCULSION

This performed a tradeoff analysis of system methodologies utilizing M- Partition Privacy to answer user queries. Finally, applied our analysis results to the design of a M – Partition Privacy algorithm to identify and apply the best design parameter settings in J2EE. Then implemented the proposed scheme, and conducted comprehensive performance analysis and evaluation, which showed its efficiency and advantages over existing schemes.

REFERENCES

[1] Olvi L. Mangasarian, "Privacy-Preserving Horizontally Partitioned Linear Programs".2003
[2] P. Krishna Prasad and C. Pandu Rangan "Privacy Preserving BIRCH Algorithm for Clustering over Arbitrarily Partitioned Databases".
[3] Ali Inan, Yücel Saygın, Erkay Sava, Ayça Azgın Hinto lu, Albert Levi "Privacy Preserving Clustering on Horizontally Partitioned Data".
[4] Jaideep Vaidya and Chris Clifton "Privacy-Preserving Decision Trees over Vertically Partitioned Data"
[5] Zhiqiang Yang and Rebecca N. Wright, "Privacy-Preserving Computation of Bayesian Networks on Vertically Partitioned Data" IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 18, NO. 9, SEPTEMBER 2006
[6] Khuong Vu and Rong Zheng Jie Gao "Efficient Algorithms for K-Anonymous Location Privacy in Participatory Sensing" 2012 Proceedings IEEE INFOCOM
[7] Sebastian Schrittwieser, Peter Kieseberg, Isao Echizen, Sven Wohlgemuth, Noboru Sonehara, and Edgar Weippl "An Algorithm for k-anonymity-based Fingerprinting"
[8] S. Goryczka, L. Xiong, and B. C. M. Fung, "m-Privacy for collaborative data publishing," in Proc. of the 7th Intl. Conf. on Collaborative Computing: Networking, Applications and Work sharing, 2011.
[9] W. Jiang and C. Clifton, "A secure distributed framework for achieving k-anonymity," VLDB J., vol. 15, no. 4, pp. 316–333, 2006
[10] L. Sweeney, "k-Anonymity: a model for protecting privacy," Int. J. Uncertain. Fuzziness Knowl.-Based Syst., vol. 10, no. 5, pp. 557–570, 2002.