ISSN:0975-9646

A Survey on m-Privacy for Collaborative Data Publishing

Sarita D. Kashid,and Prof. Manasi K. Kulkarni

Department of Computer Engineering

Modern College of Engineering, Shivaji Nagar Pune-411004 (M.S.) India

***Abstract*— In this work , we have consider the collaborative data publishing approach in order to anonymize horizontally partitioned data at multiple data providers. Here we can use two different approach of annonymization such as Anonymize-and-Aggregate or Aggregate-and-Anonymize. In this proposed system implementation we are going to implement a data provider-aware anonymization algorithm with adaptive m-privacy checking strategies . This will provide us high utility and m-privacy of anonymized data with higher efficiency. Finally, we are going to propose secure multi-party computation protocols (SMC) for collaborative data publishing with m-privacy. Here we can use either a trusted third-party (TTP) or Secure Multi-party Computation (SMC) protocols. We have also consider a new type of "insider attack" which may be conducting by data providers who may use their own data records to infer the data records contributed by other data providers.**

In this literature survey, we have described previous various approach for data publishing with their various advantages and limitation. Also we have described the possible attacks on each approach. In this paper, we have compared K-Anonymity, L-Diversity and t-Closeness approach .

Keywords: **Anonymization , Insider-attack, m-Privacy, Security, Secure Multi-party Computation (SMC), Trusted Third-Party (TTP) .**

I. INTRODUCTION

Now a days, there is an increasing need for sharing the data and available information which contain personal information stored in distributed databases. Consider any data holder, such as a hospital or a bank, that has a privately held collection

of person-specific information, and other field structured data. In health care , Nationwide Health Information Network (NHIN) has been developed to share information among hospitals and other providers.

Here, we need to understand the various terminology about the data holder,record owner and data recipient . Suppose ,a hospital collects data from patients and publishes the patient records to an external medical center. In this particular example, the hospital is the data publisher, patients are record owners, and the medical center is the data recipient. The data mining or knowledge discovery conducted at the medical center in order to extract the useful information from this raw data. For example we can simply count the total number of men with cancer by using cluster analysis.

In recent years, Privacy preserving data analysis, and data publishing have been become a promising approaches for sharing data to the other people while preserving individual privacy.

When data are gathered from various multiple data providers or data owners, at that time we need to use two main settings for anonymization of the available data.

1. **Anonymize-and-Aggregate**

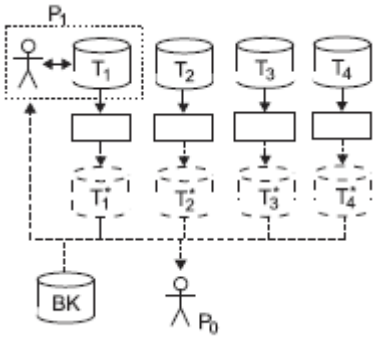

Fig1.1: Anonymize-and-Aggregate

In this approach each data provider or data owner first anonymize the data independently
which results in potential loss of integrated data utility.

2. **Aggregate-and-Anonymize**

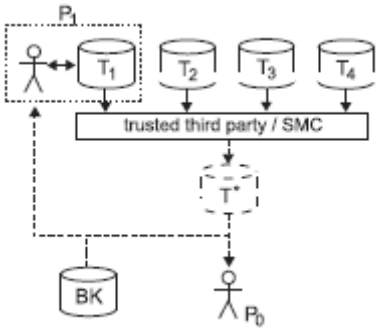

Fig 1.2: Aggregate-and-Anonymize.

Here, one of the most desirable approach is collaborative data publishing which first anonymizes data from all providers as this all data would come from one source. This approach generally uses either a trusted third-party (TTP) or Secure Multi-party Computation (SMC) protocols. Collaborative data publishing can be considered as a multi-party computation problem, in which multiple providers wish to compute an anonymized view of their data without disclosing any private and sensitive information. A trusted third party (TTP) or Secure Multi-Party Computation (SMC)

protocols gives the guarantee that there will be no any disclosure of intermediate information during the anonymization.

If we share and publish the detailed person-specific data in its original form , it may often contains sensitive information about individuals, and publishing such data immediately violates individual privacy. Even if we make policies and guidelines to restrict use and storage of such published sensitive data we can not give the guarantee that sensitive data will not be carelessly misplaced and end up in the wrong hands.

So, we need to develop methods and tools for publishing data in a more hostile environment, so that the published data remains useful for all other user while individual privacy is preserved. This undertaking is called privacy-preserving data publishing (PPDP).

II. PROBLEM STATEMENT

An Let us consider the collaborative data publishing approach with horizontally distributed data across the several data providers, where each data provider contributing a subset of records T_i. Here, we have to protect the identity of data owner.

In collaborative data publishing approach Each record consist of attribute is either an

identifier or a quasi -identifier (QID), which may identify the owner if joined with a publicly known dataset, or a sensitive attribute, we have also protect this sensitive attribute. A data recipient may have access to some background knowledge (BK) . Background knowledge is nothing but the information available about the released data, e.g., Census datasets.

The goal of our proposed approach ,is to publish an anonymized view of the integrated data, T □ , which will provide resistance to attacks.

A.Challenges of Proposed Approach:

Collaborative data publishing aooroach introduces a new attack that has not been studied till now. Here,Each data provider, such as P_1 can use both, anonymized data T □ , as well as its own data T_1

in order to get some additional information about other records. Each data provider has additional data knowledge of its own records, which can help with the attack. This issue can further become worse when multiple data providers collude with each other.

A user may attempt to infer private information about other users with the help of anonymized data by some background knowledge and with his/her own account information. Malicious users may collude or even he/she can create artificial accounts as in a shilling attack.

III. LITERATURE SURVEY

A. K-Anonymity

K-anonymity is a framework [3]which is used for constructing and evaluating algorithms & systems that release information. Suppose , If we want to determine that how many released tuples actually matches with individuals then we need to combine the released data with externally available data and analyzing other possible attacks.

So, Making such a type of determination directly can be very difficult task for the data holder , those who have released the information.

For example ,suppose If you want to identify a particular person and the only information you have is gender and zip code then there should be at least k number of people that meets the above requirement.

Basically,K-anonymity has two major techniques first Generalization and second Suppression .

In order to protect respondents' identity when releasing micro data, data holders are often eliminate orencrypt explicit identifiers, such as names and security numbers. K-anonymity does not provide guarantee of anonymity in de-identifying of data.

I. Definition -k-Anonymity

Let $RT(A_1 ,...,A_n)$ be a table and QI_{RT} be the quasi-identifier associated with it.

RT will satisfy k-anonymity if and only if each sequence of values in $RT[QI_{RT}]$ appears with at least k occurrences in $RT[QI_{RT}]$.

where Notations Represent:

RT =represents the Released Tables.

QI : Quasi -Identifier

$(A_1, A_2,.....,A_n)$ Attributes of the Released Tables

II.Attacks On k-Anonymity

Even if we have taken sufficient care in identification of the quasi-identifier, a solution to k-anonymity can still be vulnerable to attacks. Here, we have described the various attacks on K-Anonymity.

1. Unsorted Matching Attack Against K-Anonymity

Unsorted matching attack is based on the order in which tuples appear in the released table. Generally we never considering the tuple order in relational database, in real world use and this often creates problem. But we can prevent this attack just simply, by randomly sorting the tuples of the solution table. If we can not do this randomly sorting then there may be possibility that , the release of a related table may leak sensitive information.

2.Complementary Release Attack Against K-Anonymity

Sometime , what happen all the attributes my be in the quasi-identifier. But this is not the case always. It is more common that the attributes that constitute the

quasi-identifier are themselves a subset of the attributes released. Therefore

when a table T is released, it should be considered as joining other external information. Therefore, the subsequent releases of the same private data must consider all of the released attributes of T a quasi-identifier that prohibit linking on T, unless the subsequent releases are based on T.

Here , in following section author also has provided the solution on the complementary release attack.

First, We need to Consider all the attributes of previously released tables before releasing the new table.

Second, make base of the subsequent releases on the initially released table.

3.Temporal Attack Against K-Anonymity

Data collection is dynamic process, where tuples can be added, updated, and removed constantly. Therefore , the releases of generalized data over time can be subject to a temporal inference attack.

Let us consider , table T_0 be the original privately held table at time t=0.

Now, we can assume that a k-anonymity solution based on T_0 , let us say released table RT_0 . Now suppose at time t, if we add some additional tuples to the privately held table T_0 , so it becomes time= T_t . Let RT_t be released table which is a k-anonymity solution at time T_t . due to linking of both the released table RT_0 and RT_t may reveal some sensitive information or attribute and therefore we need to compromise k-anonymity protection.

Here, author have also expressed the solution for this type of attack:

1. In order avoid the temporal attack , we can consider all the attributes released in an initial table as quasi identifiers for subsequent releases.
2. all these subsequent releases should be based on the initial releases.

III.Advantages and limitation

1.K-Anonymity approach gives guarantee that individuals cannot be identified by linking attacks.

Limitations:

1.How to identify a set of "Quasi Identifier"?. This is one of the most challenging task in K-Knonymity.

2.For the large number of Quasi Identifiers this approach may not be suitable.

3. in order to protect the data this approach generally suppresses or generalize the quasi identifiers which may reduces the quality of data.

4. It fails protect attack on background knowledge.

B. l-Diversity: Privacy Beyond k-Anonymity

One of the shortcoming of k-anonymity is that it is susceptible to homogeneity attack and background knowledge attacks. So, we need to have a stronger definition of privacy .

Here, in this work [5] author have introduced an ideal notion of privacy called Bayes-optimal for the case that both data provider and the adversary will have full information about background knowledge . But, in practice, the data publisher is unlikely to possess all this information, at the same time the adversary may have more specific background knowledge than the data provider . Therefore, Bayes-optimal privacy gives us a good result as per as theory is concerned , but it is unlikely that it can be guaranteed in practice.

To address this problem, author have described Bayes-optimal privacy which naturally leads to a practical definition called l-diversity.

l-Diversity provides privacy even when the data publisher does not know what kind of knowledge is possessed by the adversary.

The basic idea behind the l-diversity is that we have to well-represent the values of the sensitive attributes in each group.

I.The l-diversity Principle:

An equivalence class is said to have l -diversity if there are at least "well-represented" values for the each sensitive attribute. Here, we can say that a table is in l-diversity if every equivalence class of the table has l-diversity.

1. Distinct l-diversity.

The simplest understanding of "well represented" would be to ensure that there are at least distinct values for the sensitive attribute in each equivalence class.

In Distinct l-diversity Each equivalence class must have at least l well-represented sensitive values

Distinct l -diversity does not prevent any probabilistic inference attacks. An equivalence class may have one value which appears more frequently than other values, taht enables us an adversary to conclude that an entity in the equivalence class is very likely to have that value.

2. Entropy l **-diversity.**

The entropy of an equivalence class E is defined with the help of following formula.

$$Entropy(E) = -\sum_{s \in S} p(E,s) \log p(E,s)$$

in above formula:

S- it is the domain of sensitive attribute,

p(E, s)-it is the fraction of records in E in which sensitive value s is present.

A table is said to have entropy l-diversity if for every equivalence class E, Entropy(E) ≥ log l . Entropy l-diversity is more stronger than distinct l-diversity.

One of the most important charaecterstic of Entropy l-diversity is that here, in each equivalence class not only have an enough different sensitive values, but also it must have enough different sensitive values distributed evenly . It means that the entropy of the distribution of sensitive values in each equivalence class is at least log (l).Sometimes this may be too restrictive.

When some values are very common, then we can say that the entropy of the entire table may be very low.

3.Recursive (c,l)-diversity.

Recursive (c, l)-diversity tells that the most frequent value does not appear too frequently, and at the same time the less frequent values do not appear too rarely. Let m be the number of values in an equivalence class, and r_i , where 1 ≤ i ≤ m be the number of times that the i^{th} most frequent sensitive value appears in an equivalence class E.

Then Equivalence class E is said to be recursive (c, l)-diversity if $r_1 < c(r_l + r_{l+1} + ... + r_m)$. where c is any specified constant.

II.Types of Attack on l-diversity:

There are two types of attack on l-Diversity first Skewness attack and second similarity attack.Below we have briefly presented this two attacks on l -diversity.

1.Skewness Attack:

When the overall distribution is skewed, which satisfy the l-diversity does not prevent attribute disclosure. This type of attack is called skewness attack.

2.Similarity Attack:
When the sensitive attribute values in an equivalence class are distinct but semantically similar, then there may be possibility that an adversary can learn important information.

III. Advantgage &Limitations of l-Diversity:
l-Diversity preserve the privacy even if the
data publisher does not aware about what kind of background knowledge is
achieved by the adversary.
Limitation:
l-Diversity doesn't protect the probabilistic inference
attacks which is more intuitive to the
data publisher.

C. t-Closeness: A New Privacy Measure
I.Introduction
as we studied from the literature that l-diversity has a number of limitations , so to overcome this author have proposed a novel privacy notion called t-closeness. The principle of t-closeness state that distribution of a sensitive attribute in any equivalence class is always close to the distribution of the attribute in the overall table .
In this work, author have used the Earth Mover Distance measure for the t-closeness. This provides us the semantic closeness of attribute values.
As we have studied that K-anonymity prevents identity disclosure but it never prevent an attribute disclosure. In order to solve this problem l-diversity requires each equivalence class has at least l different values for each and every sensitive attribute.
Here, Privacy is measured by the information gain of an observer. Before seeing the released table, the observer has some prior belief about the multiple sensitive attribute value of an individual.

In this work[6], author have presentd the base model of t-closeness, which requires that the distribution of a multiple sensitive attribute in any equivalence class is always close to the distribution of the attribute in the overall table , that means the distance between the two distributions must be no more than a threshold value t. t-closeness is more flexible privacy model that that provides higher utility.

II. Advantages and Limitation:
Advantages:
it is more flexible privacy model which
offers higher utility.
Limitations of T-closeness:
1.There is no computational procedure inorder to enforce t-closeness followed in. Generally we use generalizations and suppressions or slicing for the combining of attribute . So there may be possibility that there is loss of co-relation between different attributes. This loss happen because each attribute is generalized separately and due to that we may lose their dependence on each other.
2. There may be chances to damage of data utility if we use very small value of t.

IV. CONCLUSIONS

In this proposed approach we have considered a new type of potential attackers in collaborative data publishing where multiple data providers come together to perform some particular task, called called m-adversary. Privacy threats introduced by the various data providers i.e m-adversaries are modeled by a new privacy notion, called m-privacy.
From the above literature of privacy preserving in data publishing there are many complexity are there in acheiving the privacy. We come to know that K-anonymity, l-diversity has a number of limitations. We can also say that it is not required at all to prevent attribute disclosure. Beyond this a new privacy approach has been used called "(n,t)-closeness. Which helps to preserve the data from attackers.
Here, All algorithms have been implemented in distributed settings with a Tursted Third Party and as SMC protocols. We can also have future scope to implement collabrative data publishing and address the data knowledge of data providers when data are distributed in a vertical fashion or in ad-hoc fashion. This proposed approach further we can implement to other kinds of data such as set-valued data.

REFERENCES

[1].Goryczka Et Al.: "m-privacy For Collaborative Data Publishing", IEEE Transactions On Knowledge And Data Engineering, Vol. 26, No. 10, October 2014.
[2].S. Goryczka, L. Xiong, and B. C. M. Fung, "m-Privacy for collaborative data publishing," in Proc. 7th Int. Conf. CollaborateCom, Orlando, FL, USA, 2011.
[3] L. Sweeney, "k-Anonymity: A model for protecting privacy," Int. J. Uncertain. Fuzz. Knowl. Based Syst., vol. 10, no. 5, pp. 557–570, 2002.
[4] W. Jiang and C. Clifton, "Privacy-preserving distributed k-anonymity," in Proc. 19th Annu. IFIP WG 11.3 Working Conf., vol. 3654. Storrs, CT, USA, 2005, p. 924.
[5] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "L-Diversity: Privacy beyond k-anonymity," in Proc. 22nd ICDE, Atlanta, GA, USA, 2006, p. 24.
[6] N. Li and T. Li, "t-Closeness: Privacy beyond k-anonymity and l-diversity," in Proc. IEEE 23rd ICDE, Istanbul, Turkey, 2007.
[7] W. Jiang and C. Clifton, "A secure distributed framework for achieving k-anonymity," VLDB J., vol. 15, no. 4, pp. 316–333, 2006.
[8] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," ACM CSUR, vol. 42, no. 4, Article 14, Jun. 2010.
[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] Y. Lindell and B. Pinkas, "Secure multiparty computation for privacy-preserving data mining," J. Privacy Confident., vol. 1, no. 1, pp. 59–98, 2009.