International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Privacy-Preservation of Centralized and Distributed Social Network by Using L-Diversity Algorithm

Shankaranand[1]**, P. Rajasekar**[2]

[1, 2]SRM University, Chennai, India

Abstract: ***Publishing data about individuals without revealing sensitive information about them is an important problem. In recent years, a new definition of privacy called k-anonymity has gained popularity. In a k-anonymized dataset, each record is indistinguishable from at least k−1 other records with respect to certain "identifying" attributes. The problem of Social Network is getting secured data from unauthorized access of database. To consider the distributed configurations in which the network data is split between several data holders. The data is divided between a numbers of data holders. The plan is to get there at an anonymized view of the combined network without informative to any of the data holders. Two variants of an anonymization algorithm which is based in order clustering. High sensitive data has been secured in l-diversity algorithms. Based on the retrieval of data from the database, calculation of data loss has to be done. Also the analyzing of data that how secure the database and also by calculating the data loss. In addition to building a formal foundation for ℓ-diversity, we show in an experimental evaluation that ℓ-diversity is practical and can be implemented efficiently.***

Keywords: Anonymization, Clustring, l-diversity, Social Network

1. Introduction

Networks are structures that describe a set of entities and the relations between them. A social network, for example, provides information on individuals in some population and the links between them ns of friendship, collaboration, correspondence, and so forth. An information network, as another example, may describe scientific publications and their citation links. In their most basic form, networks are modeled by a graph, where the nodes of the graph correspond to the entities, while edges denote relations between them. Real social networks may be more complex or contain additional information. For example, in networks where the described interaction is asymmetric (e.g., a financial transaction network), the graph would be directed; if the interaction involves more than two parties (e.g., a social network that describes co-membership in social clubs) then the network would be modeled as a hypergraph; in case where there are several types of interaction, the edges would be labeled; or the nodes in the graph could be accompanied by attributes that provide demographic information, which may describe relation such as age, gen- der, location, occupation which could enrich and shed light on the structure of the network.

Nowadays, online social media services are growing rapidly day by day and it has given an impact on the way people interact with each other. The Online social networks such as Twitter, Facebook and LinkedIn have become one of the most popular activities on the web. According to the recent study, more than 80% of the university students in America are active members of online social network and spending 30 minutes on average in everyday life. Most of the business owners actively use social network as part of their marketing strategy. These social networks collect huge amount of data about user and their activity and relations. On positive side, this collected data gives great analysis opportunity to data miners/researchers, and on the negative side the data gives a threat to user's data privacy.

The privacy disclosure in a social network can be grouped to three categories: 1) Identity disclosure: the identify of an individual who is associated with a vertex is revealed; 2) Link disclosure: the sensitive relationships between two individuals are disclosed; 3) Sensitive attribute disclosure: the sensitive data associated with each node is compromised e.g., the email message sent/received by the individual in an email communication network. A privacy preservation system over graph and networks should consider all of these issues.

Table 1: Inpatient Microdata

	Zip	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	14853	50	Indian	Cancer
4	14853	55	Russian	Heart Disease
5	14850	47	American	Viral Infection
6	13068	35	American	Cancer
7	13068	36	Japanese	Cancer
8	13068	21	Japanese	Viral Infection

Table 2: ℓ-Diversity: A Practical Privacy Definition

	Non-sensitive		Sensitive	
	Zip Code	Age	Nationality	Condition
1	130**	< 30	*	Heart Disease
2	130**	< 30	*	Heart Disease
3	148**	>40	*	Cancer
4	148**	>40	*	Heart Disease
5	148**	>40	*	Viral
6	130**	3*	*	Cancer
7	130**	2*	*	Cancer
8	130**	3*	*	Viral

2. ℓ-Diversity: A Practical Privacy Definition

In this section we discuss how to overcome the difficulties outlined at the end of the previous section. We derive the ℓ-diversity principle, show how to instantiate it with specific definitions of privacy, outline how to handle multiple

Licensed Under Creative Commons Attribution CC BY

sensitive attributes, and discuss how ℓ-diversity addresses the issues raised in the previous section.

2.1 The ℓ-Diversity Principle

Let us define a q⋆ -block to be the set of tuples in T⋆ whose non-sensitive attribute values generalize to q⋆ . Consider the case of positive disclosures; i.e., Alice wants to determine that Bob has t[S] = s with very high probability. this can happen only when: ∃ s, ∀ s′ 6= s, n(q⋆ ,s′) f(s′|q) f(s′|q⋆) ≪ n(q⋆ ,s) f(s|q) f(s|q⋆) (2)

The condition in Equation (2) could occur due to a combination of two factors: (i) a lack of diversity in the sensitive attributes in the q⋆ -block, and/or (ii) strong background knowledge. Let us discuss these in turn. Lack of Diversity. Lack of diversity in the sensitive at- tribute manifests itself as follows: ∀ s′ 6= s, n(q⋆ ,s′) ≪ n(q⋆ ,s) (3)

In this case, almost all tuples have the same value s for the sensitive attribute S, and thus β(q,s,T ⋆) ≈ 1. Note that this condition can be easily checked since it only involves counting the values of S in the published table T⋆ . We can ensure diversity by requiring that all the possible values s′ ∈ domain(S) occur in the q⋆ -block with roughly equal proportions. This, however, is likely to cause significant loss of information: if domain(S) is large then the q⋆ -blocks will necessarily be large and so the data will be partitioned into a small number of q⋆ -blocks. Another way to ensure diversity and to guard against Equation 3 is to require that a q⋆ -block has at least ℓ ≥ 2 different sensitive values such that the ℓ most frequent values (in the q⋆ -block) have roughly the same frequency. We say that such a q⋆ - block is well-represented by ℓ sensitive values. Strong Background Knowledge. The other factor that could lead to a positive disclosure (Equation 2) is strong background knowledge. Even though aq⋆ block may haveℓ "well-represented" sensitive values, Alice may still be able to use her background knowledge to eliminate sensitive values when the following is true: ∃ s′, f(s′|q) f(s′|q⋆) ≈ 0 (4)

This equation states that Bob with quasi-identifier t[Q] = q is much less likely to have sensitive value s′ than any other individual in the q⋆ -block. For example, Alice may know that Bob never travels, and thus he is extremely unlikely to have Ebola. It is not possible for a data publisher to guard against attacks employing arbitrary amounts of back- ground knowledge. However, the data publisher can still guard against many attacks even without having access to Alice's background knowledge. In our model, Alice might know the distribution f(q,s) over the sensitive and non- sensitive attributes, in addition to the conditional distribution f(s|q). The most damaging type of such information has the form f(s|q) ≈ 0, e.g., "men do not have breast cancer", or the form of Equation 4, e.g., "among Asians, Japanese have a very low incidence of heart disease". Note that a priori information of the form f(s|q) = 1 is not as harmful since this positive disclosure is independent of the published table T⋆ . Alice can also eliminate sensitive values with instance-level knowledge such as "Bob does not have diabetes". In spite of such background knowledge, if there are ℓ "well represented" sensitive values in a q⋆ -block, then Al- ice needs ℓ − 1 damaging pieces of background knowledge to eliminate ℓ − 1 possible sensitive values and infer a positive disclosure! Thus, by setting the parameter ℓ, the data publisher can determine how much protection is provided against background knowledge — even if this background knowledge is unknown to the publisher. Putting these two arguments together, we arrive at the following principle.

Principle 2 (ℓ-Diversity Principle) A q⋆ -block is ℓ-diverse if contains at least ℓ "well-represented" values for the sensitive attribute S. A table is ℓ-diverse if every q⋆ -block is ℓ-diverse. Returning to our example, consider the inpatient records shown in Figure 1. We present a 3-diverse version of the table in Figure 3. Comparing it with the 4-anonymoustable in Figure 2 we see that the attacks against the 4-anonymoustable are prevented by the 3-diverse table. For example, Alice cannot infer from the 3-diverse table that Bob (a 31 year old American from zip code 13053) has cancer. Even though Umeko (a 21 year old Japanese from zip code 13068) is extremely unlikely to have heart disease, Alice is still unsure whether Umeko has a viral infection or cancer. The ℓ-diversity principle advocates ensuring ℓ "well re-presented" values for the sensitive attribute in every q⋆ -block, but does not clearly state what "well represented" means. Note that we called it a "principle" instead of a theorem — we will use it to give two concrete instantiations of the ℓ-diversity principle and discuss their relative tradeoffs.

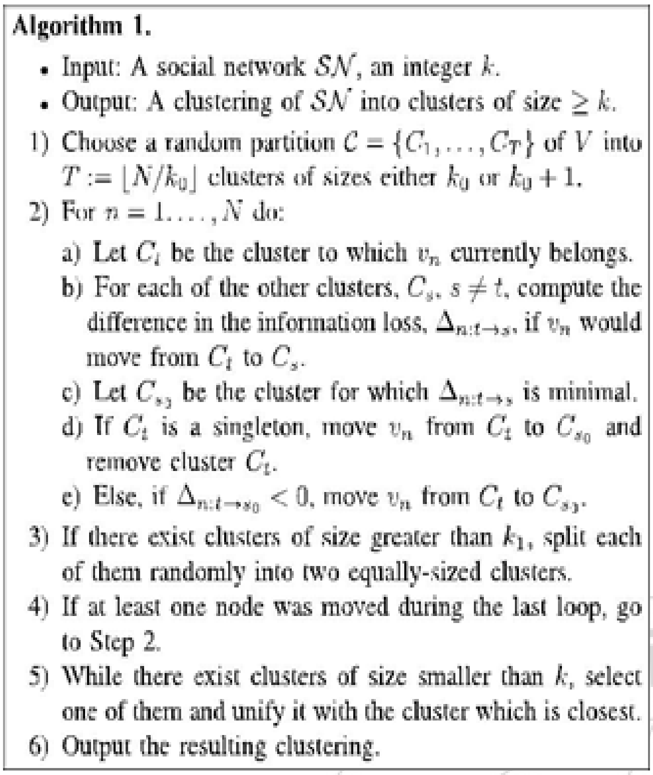
Figure3.3: Diverse Inpatient Microdata

	Non-sensitive		Sensitive	
	Zip Code	Age	Nationality	Condition
1	130**	< 40	*	Heart Disease
6	130**	< 40	*	Cancer
7	130**	< 40	*	Cancer
3	148**	>40	*	Cancer
4	148**	>40	*	Heart Disease
5	148**	>40	*	Viral Infection
2	130**	<=40	*	Heart Disease
8	130**	<=40	*	Viral Infection

3. Anonymization by Sequence Clustering

The sequential clustering algorithm for k-anonymizing tables was presented. It was shown there to be a very efficient algorithm in terms of runtime as well as in terms of the utility of the output anonymization. We proceed to describe an adaptation of it for anonymizing social networks. Algorithm 1 starts with a random partitioning of the network nodes into clusters. The initial number of clusters in the random partition is set to bN=k0c and the initial clusters are chosen so that all of them are of size k0 or k0 þ 1, where k0 ¼ k is an integer and is some parameter that needs to be determined.

Licensed Under Creative Commons Attribution CC BY



```
Algorithm 1.
  • Input: A social network $SN$, an integer $k$.
  • Output: A clustering of $SN$ into clusters of size $\geq k$.
1) Choose a random partition $C = \{C_1, \ldots, C_T\}$ of $V$ into
   $T := \lfloor N/k_0 \rfloor$ clusters of sizes either $k_0$ or $k_0 + 1$.
2) For $n = 1, \ldots, N$ do:
   a) Let $C_t$ be the cluster to which $v_n$ currently belongs.
   b) For each of the other clusters, $C_s$, $s \neq t$, compute the
      difference in the information loss, $\Delta_{n:t\to s}$, if $v_n$ would
      move from $C_t$ to $C_s$.
   c) Let $C_{s_3}$ be the cluster for which $\Delta_{n:t\to s}$ is minimal.
   d) If $C_t$ is a singleton, move $v_n$ from $C_t$ to $C_{s_0}$ and
      remove cluster $C_t$.
   e) Else, if $\Delta_{n:t\to s_0} < 0$, move $v_n$ from $C_t$ to $C_{s_3}$.
3) If there exist clusters of size greater than $k_1$, split each
   of them randomly into two equally-sized clusters.
4) If at least one node was moved during the last loop, go
   to Step 2.
5) While there exist clusters of size smaller than $k$, select
   one of them and unify it with the cluster which is closest.
6) Output the resulting clustering.
```

The Sequential clustering achieves significantly better results than SanGreeA, in terms of information loss, as we demonstrate later on in Section 6. One reason is that greedy algorithms, such as SaNGreeA, do not have a mechanism of correcting bad clustering decisions that were made in an earlier stage; sequential clustering, on the other hand, constantly allows the correction of previous clustering decisions. Another advantage of sequential clustering over SaNGreeA is that it may evaluate at each stage during its operation the actual measure of information loss, since at each stage it has a full clustering of all nodes. The latter advantage in terms of utility translates to a disadvantage in terms of runtime. While SaNGreeA requires evaluations of the cost function, the number of cost function evaluations in the sequential clustering depends on N3. (The algorithm scans all N nodes and for each one it considers alternative cluster allocations; the computation of the cost function for each such candidate alternative clustering requires to update the inter cluster costs for all pairs of clusters that involve either the cluster of origin or the cluster of destination in that contemplated move.) Hence, we proceed to describe a relaxed variant of sequential clustering which requires only evaluations of the cost function.

A **Modified Structural Information Loss Measure**

The proposed SaNGreeA algorithm uses a measure of structural information loss We proceed to define it. Let B be the N ×N adjacency matrix of the graph G =(V,E), i.e.,B(n,n')=1 if{v,v'}ε E and B(n,n')=0 otherwise. Then, a Hamming-like distance is defined on V as follows:

$$D(n,n') := \frac{\left|\left\{\ell \neq n,n' : B(n,\ell) \neq B(n',\ell)\right\}\right|}{N-2}.$$

This definition of distance induces the following measure of structural information loss per cluster and a corresponding overall structural information loss.

$$I'_S(C) = \frac{1}{N}\sum_{t=1}^{T}|C_t| \cdot I'_S(C_t) = \sum_{t=1}^{T} x(C_t),$$

Where

$$x(C_t) = \frac{2}{N(|C_t|-1)}\sum_{v_n,v_{n'}\in C_t} \mathrm{D}(n,n').$$

In other words, I0 S of a given cluster is the average distance between all pairs of nodes in that cluster, and I0 S of the whole clustering is the corresponding weighted average of structural information losses over all clusters. The corresponding weighted measure of information loss is then.

$$I'(C) = w \cdot I_D(C) + (1-w) \cdot I'_S(C),$$

4. Conclusion

We have designed and implemented the Privacy Preservation of Social Networks to facilitate the preservation of databases of social networks by Sequential Clustering algorithm. Initially the databases of Social Networks are divided based on the columns and by applying the algorithms we have to secure that separated databases. By using the k- Anonymity algorithm, creation of random key have done for securing the data in the tables from the hackers. The Sensitive data like age and marital status are secured by using the l-diversity algorithm. Generations of duplicate random keys for each data when the user registered every time has done. Finally the data's from the different databases are grouped by using the Sequential Clustering Algorithm and the analyzing of data loss in the database.

5. Scope for Future Extension

In this we projected on not only securing the data in the tables by adding the keywords for the every data that are already in the tables. The keywords that are created are randomized and it will change for every user when the new users are registered. This will make the time when the data are extracted and when new user are registered the random key are generated for every registration. This result in the performance slows down. This can be overcome by future extension Also this can be implemented in all sorts of networks and securable databases.

References

[1] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu, "Anonymizing Tables," Proc. 10th Int'l Conf Database Theory (ICDT), vol. 3363, pp. 246-258, 2005.

[2] A. Campan and T.M. Truta, "Data and Structural k-Anonymity in Social Networks," Proc. Second ACM SIGKDD Int'l Workshop Privacy, Security, and Trust in KDD (PinKDD), pp. 33-54, 2008.

Licensed Under Creative Commons Attribution CC BY

[3] M. Hay, G. Miklau, D. Jensen, P. Weis, and S. Srivastava, “Anonymizing Social Networks,” Univ. of Massachusetts, technical report, vol. 7, no. 19, 2007.

[4] V. Iyengar, “Transforming Data to Satisfy Privacy Constraints,” Proc. Eighth ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD), pp. 279-288, 2002.

[5] W. Jiang and C. Clifton, “A Secure Distributed Framework for Achieving k-Anonymity,” The Int’l J. Very Large Data Bases, vol. 15, pp. 316-333, 2006.

[6] M. Kantarcioglu and C. Clifton, “Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data,” IEEE Trans.

[7] A. Schuster, R. Wolff, and B. Gilburd, “Privacy-Preserving Association Rule Mining in Large-Scale Distributed Systems,” Proc. IEEE Int’l Symp. Cluster Computing and the Grid (CCGRID), pp. 411-418, 2004.

[8] X. Ying and X. Wu, “Graph Generation with Prescribed Feature Constraints,” Proc. SIAM Conf. Data Mining (SDM), pp. 966-977, 2009.

Author Profile

Shankaranand received the B.E. degrees in Information Technology Engineering from ECE, Bilaspur. He is currently pursuing master’s degree program in Information technology in SRM University, Chennai , India

www.ijsr.net IJSR ISSN (Online): 2319-7064

Licensed Under Creative Commons Attribution CC BY