ISSN 2278-3091
Volume 9, No.4, July – August 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse110942020.pdf
https://doi.org/10.30534/ijatcse/2020/110942020

A Novel Method for Privacy Preservation of Health Data Stream

Ganesh Dagadu Puri[1], D. Haritha[2]
[1,2]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India,
[1]puriganeshengg@gmail.com, [2]haritha_donavalli@kluniversity.in

ABSTRACT

Most of the conventional privacy preservation methods focus on static datasets. These methods cannot be applied as it is on real-world datasets; which dynamically modify data. If one tuple in the dataset get modified, statistics of complete dataset gets changed. Privacy preservation measures must be applied after modification of dataset. Such re-anonymization of complete dataset is incompetent when large datasets are often changed. When dynamic stream data is considered, we have to apply different privacy techniques which can apply privacy on each tuple. Although several studies have addressed data privacy for static, dynamic and stream data, they are not adequate for avoiding similarity attack and reducing data loss in privacy preservation. Even in the big streaming data privacy preservation, repetition of L-diverse group can take place. This repetition can cause re-identification of individual. Therefore, we identified limitations of data-privacy preservation for stream data and developed more efficient L-diversity algorithm for preserving privacy of data streams. We used hash values of L-diverse groups to find similarity among the groups. Human disease ontology is used to find synonyms of disease terms. Experimental results demonstrated that, our proposed anonymization algorithm reduced data loss of L-diverse group.

Key words: Ontology, privacy preservation, similarity, stream data.

1. INTRODUCTION

There are several methods used for privacy preservation of static data. These methods can be classified based on the attributes modified in privacy preservation. Privacy can preserved by anonymizing quasi attributes or sensitive attributes in the tuple [1]. L-diversity approach work with the sensitive attribute of tuple [2]. In the L-diversity approach, the (L-1) values are available along with the original value of the tuple. These (L-1) values can be counterfeit values in the group so that attacker cannot understand the sensitive value of individual. This can be done in two ways. In the first method, the tuples in the structured form can be k-anonymized so that attacker cannot find the correct identity of the individual. Second way is to consider equivalent class of the k-anonymized group of tuples and maintain L-diversity in the each equivalent class. Using these approaches we can maintain the privacy of the individual. But these approaches are based on the grouping of the tuples in the table and may not be suitable as it is for large data. Another problem with these approaches is anonymization delay. It is introduced due to the waiting time of arrival of next tuple and arranged in that group. Even if we use the above mentioned methods for dynamic stream data, it will introduce anonymization delay. To make the equivalent classes L-diverse proper clustering of the sensitive values should take place. To avoid this anonymization delay, counterfeit values are inserted in L-diverse group. Due to these counterfeit values in the group; information loss will be available until late validation take place for the counterfeit values. When the counterfeit values get added in the L-diverse group; there is possibility of similar values to be added in the group. These similar values in the L-diverse group can cause the identification of the sensitive value of individual [3]. For example in the 3-diverse group of equivalent class the values are available as gastric ulcer, gastritis and stomach cancer [4]. So from this equivalent class attacker can link the quasi attributes in that equivalent class and come to the conclusion that individual present in that group and suffering from stomach problem. According to the principal of the L-diversity approach, the attacker should not be able to identify the correct sensitive value of individual even if he/she know the quasi attributes of the individual. So this is very important to avoid the similarity attack due to the counterfeit values in the L-diverse group.

Ontology structures are used for presenting domain knowledge and relationship among the concepts in that domain [5]. Semantic similarity is used to find “is-a” relation between the terms. Ontologies can be used to identify semantic similarity between words or concepts [6]. In the human disease ontology, synonyms for human disease are defined [7]. Using the relatedness measure like Wu-palmer, Resnik, Lin; the two disease values can be compared [8]. Relatedness measure between the two values can be found. If the two medical terms are exactly similar, then the relatedness is more. When we work with big streaming data, incoming

data in the form of tuples will be large volume. To handle such large data, we are using the different processing nodes, which can execute in parallel. For such large data, more number of L-diverse groups will be created. In these groups, we have to find similarity. If we use the traditional methods to compare every two terms in the group and find the relatedness of these two terms, it will be time consuming task. So we have to use the similarity method, which can apply on the group.

There are two possible attacks which can take place in the L-diversity. These are similarity attack and skewness attack. The re-identification of individual is possible due to repetition of the L-diverse group also. The repetition of L-diverse groups is due to the large number of incoming tuples. When number of tuples gets increased, it may happen that the same group of L-diverse values will be created again and again. Such repetition of the same group will cause the re-identification attack. Attacker can easily identify, that the individual present in the group is suffering with particular disease.

2. RELATED WORK

2.1 Static data privacy preservation

Existing privacy models like k-anonymity, L-diversity work well for static data. In the static data no update, delete operations will be performed. In paper [9], authors suggested generalization on quasi identifiers using k-anonymity. The record in equivalent class is not identifiable by (k-1) records in that equivalent class [10]. Re-identification attack can take place on quasi identifiers. In paper [2], authors suggested L-diversity approach where in one equivalent class, sensitive values of that equivalent class can be made L-diverse so that attacker cannot find correct identity of individual. Anatomy approach in [11] uses L-diversity approach; but it does not generalize quasi identifiers. Even though the attacker knows the individual is present in the list, sensitive value of individual cannot be distinguished. In [4], t-closeness method proposed to provide privacy on static data. Authors Ninghui Li et al. tried to address limitations of L-diversity by considering distribution of sensitive values. Though the above methods provide privacy for static data; these methods cannot work on the dynamic data where updation, deletion operations are performed.

2.2 Dynamic data privacy preservation

Privacy preservation methods applied on the static data cannot be used as it is on dynamic data. Several methods like [12], [13] provide privacy to dynamic release by maintaining context or keeping extra information. In [12], author presented method for privacy preservation where insertion operation is considered. Author in [13] proposed m-invariance method which provide insertion and deletion operation on dynamic data. These methods provided privacy to data in the dynamic releases but the methods remained insufficient to provide privacy to tuples in real time.

2.3 Stream data privacy preservation

Privacy preservation techniques in data stream should provide privacy in real time. In [14] authors introduced specialization tree which accumulate data streams and node structures are used for generalization. CASTLE [15] scheme considers clusters to continuously anonymize data streams. In this scheme accumulation of tuples of data stream take place. It works on delay constraint δ to release the tuples in cluster. In [16], probability function is used to release the data streams. It releases the data in cluster when data loss is less but delay is longer. SABRE [17] system uses t-closeness for anonymization of data streams. It uses sliding window as buffer to maintain input until new tuple replace it. Authors in BCASTLE [18] considered the distribution of data in data stream to improve data utility. Authors of SANATOMY [19] extended ANATOMY [11] to use bucketization approach instead of generalization approach to anonymize data streams. Authors in [20] presented algorithm, which is cluster based for anonymization of numerical datastreams. All the data stream privacy preservation techniques presented in this section used accumulation of tuples approach for anonymization. Delay-free anonymization technique presented by authors in [21], reduces anonymization delay due to accumulation of tuples in clusters. In this technique counterfeit values are used to reduce anonymization delay. These counterfeit values will be late validated with incoming tuples. As L-diversity technique is employed in this approach to create counterfeit values in the sensitive value group, this approach suffers from similarity and skewness attack. Also scalable version of this technique is needed to deal with big streaming data.

2.4 Big data privacy preservation

Privacy for big data can be categorized in three parts. Privacy requirements should be applied in big data collection, big data processing and big data storage [22]. Big data collection is pervasively so privacy leakage may take place. In [23], authors explored privacy breaches in big data using four different stages. These stages include data provider, data collector, data miner and decision maker. In [24], different anonymization techniques in privacy protection are discussed. These techniques include generalization, suppression, anatomization, permutation and perturbation**.**

3. PROPOSED METHOD

When first time the tuple is fetched, the L-diverse group for that tuples sensitive value will be created. After that from next tuple, the sensitive value of incoming tuple will be checked to see whether it is available in the sensitive value of earlier published tuples. If it is available in the group of published tuples then the quasi attributes of that tuples will be added in that group. Otherwise new L-diverse group of sensitive values will be created.

3.1 Data stream and disease domain

The aim of the algorithm in Figure 1 is to check every incoming tuple in the batch of tuples. For every tuple t, following conditions hold.

1) $\exists\, t \in T$ such that each tuple contains quasi attributes and sensitive attribute.
2) For each attribute in tuple, type of attribute type $\in$ {quasi, sensitive}
3) Domain of disease contains set of disease values, from which L-diverse group will be constructed. D={D1,D2,D3….., Dn}

Incoming stream data is divided in quasi attributes and sensitive attribute. For anonymization, two schema are considered as QIT and ST.

1) QIT is (groupid, qi) and ST is (groupid, Si, count)
2) Sv is defined as Sv (Si, count) in which disease values are selected randomly from disease domain to make the L-diverse group. Sensitive values in the schema ST are selected from the Sv, where count of each sensitive value to form L-diverse group is maintained.

3.2 Similarity in L-diverse group

The similarity attack is handled at the time of creation of the L-diverse group. The L-diverse group should be created in such way that there should not be similar values in the group. At the time of L-diverse group creation following conditions hold.

1) Sg=L-diverse(Sv) and Sg $\subseteq$ Sv.

Sg is formed by using L-diverse function on the sensitive value domain. So Sg group is subset of the Sv.

2) hash(Sort (Sg)) $\notin$ hashlist

After that L-diverse value group should be sorted and its hash code should be calculated. Sort operation on the group is important because when the l-diverse groups are formed, randomly disease values are selected from sensitive value domain. In this selection of sensitive values it may happen that same groups are created for two different sensitive values. Even though the sequence of values in the group is not same, the two group values can be similar. For same set of values but ordering is different, the hash values for such groups are different. To avoid this condition first every group will be sorted and then the hash value of the group will be found. If this hash code is available in list of previously calculated hash codes, then the group is repeated. To avoid this repetition, randomly one disease value will be selected and synonym using ontology will be found. Again the hash code of the group will be calculated. If hash code does not match with previous group hash codes, then that group will be selected as L-diverse group.

The algorithm handles the similarity in the group by comparing one value with the other disease values on semantic level. If the similarity is found in the group, as algorithm shows synonym for that value will be selected. If the synonym for that value is already present in the group, then another synonym will be selected. Once the groups are formed without any similar values in the group, then it can be given for privacy preservation. Each such group contains the published sensitive values and unpublished sensitive values. The data loss of the group will be calculated as

3) Dataloss=1-(releasedcount - incomingcount).

In this equation released count shows the late validated records in the group. If all incoming records get released and late validated, then data loss is zero. Incoming count of the group is showing available counterfeit values to make group L-diverse.

Algorithm

Input: Data stream t, number of tuples processing

Output: Anonymized tuples

Schema for QIT is (groupid, q_i) and ST is (groupid, s_i, count)

```
WHILE (true)
    Read a new tuple t from T
    Divide tuple in quasi identifier q_i and sensitive identifier S_i, where q_i = {q_1,q_2,....,q_m}
    IF (S_i available in published ST)
        Add quasi identifier q_i in the group QIT
    Else
        Create L-diverse group of sensitive attribute with counterfeit values
        Check for similarity among sensitive values of that group
        IF (similarity found)
            Select random sensitive value in that group
            Find synonym Sy for that value
            IF (synonym Sy already contain in the group)
                Select next synonym Sy of that value
                Replace original value with synonym Sy
            END IF
        END IF
    Publish the tuple
    Calculate data loss, publish ratio for the tuple
    END IF
END WHILE
```

Figure 1: Algorithm for privacy preservation using proposed method to avoid similarity attack.

4. RESULT

Figure 2 shows the data loss using ontology for synonyms of the sensitive values in the group. Data loss is decreased by using synonyms of the sensitive value in the group. Using counterfeit values in L-diverse sensitive group, record get published without delay. Delay in the anonymization is reduced using such L-diverse group. The counterfeit values in the L-diverse group are late validated with incoming tuples. In initial stage data loss of that group is more. When more number of sensitive values in the group gets validated,

data loss in that group becomes less. If all values in the group get validated with incoming tuple then there is no data loss in the L-diverse group. To avoid similarity attack, we are using synonyms of that term. If the terms are similar in the group, we can replace the similar term with the synonym using ontology. For few disease terms, more than one synonymous term are available. For disease ITM2B-related cerebral amyloid angiopathy 2; the synonym terms are as HOOE, Familial Danish Dementia, Heredopathia Ophthalmootoencephalica, FDD, and Cerebellar Ataxia. There are five synonym term for ITM2B-related cerebral amyloid angiopathy 2 disease.

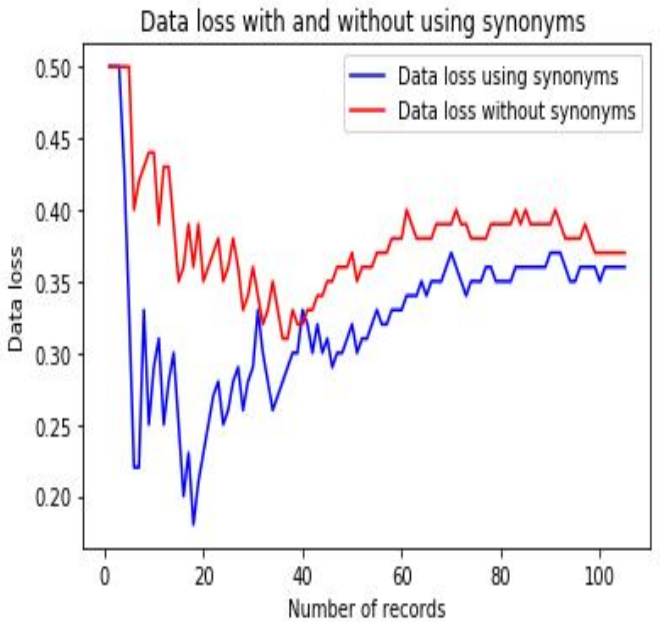

Figure 2: Data loss with and without synonyms.

Such synonyms are used to late validate the counterfeit values in the group, then publish ratio of the group get increased. Data loss of the group is minimized if the synonyms are used for late validation. Experimental accuracy of algorithm is checked by constructing the stream data with the quasi attributes age, weight, low blood pressure, high blood pressure and temperature. The sensitive attribute in the data stream is taken as disease. For the disease attribute, the values are taken from human disease ontology.

5. CONCLUSION

Privacy preservation of stream data using counterfeit values is done. Similarity attack is avoided in l-diverse group using hash values of groups. Late validation of the counterfeit values is performed in less time using synonyms of sensitive values. As the validation of the values in group is increased, data loss of the group due to counterfeit values gets reduced. This method is useful to avoid similarity attack in privacy preservation of big data in stream format.

REFERENCES

1. Ganesh D Puri, D Haritha. **Survey big data analytics, applications and privacy concerns**, *Indian Journal of Science and Technology*, Vol. 9, no 17, pp. 1-8, May 2016. https://doi.org/10.17485/ijst/2016/v9i17/93028
2. A Machanavajjhala, J Gehrke, D Kifer. **l-diversity: Privacy beyond k-anonymity,** *ACM Transactions on Knowledge Discovery from Data (TKDD)*, Vol.1, no 1 2007
3. Ganesh Dagadu Puri, D. Haritha. **Framework to Avoid Similarity Attack in Big Streaming Data,** *International Journal of Electrical and Computer Engineering (IJECE)* Vol. 8, No. 5, pp. 2920-2925, October 2018.
4. L. Ninghui, L. Tiancheng, S. Venkatasubramanian. **t-Closeness: Privacy beyond k-anonymity and ℓ-diversity**, *Proceedings - International Conference on Data Engineering,IEEE* 2007, pp. 106-115.
5. Valeriy Nikolaevich Dolzhenkov, Isa Daudovich Maltzagov, Aleksandra Igorevna Makarova, Nagbdu Sultahsikhkyzy Kamarova, Petr Viktorovich Kukhtin. **Software Tools for Ontology Development Valeriy**, *International Journal of Advanced Trends in Computer Science and Engineering*, Vol.9, no.2, pp. 935-941, March-April 2020. https://doi.org/10.30534/ijatcse/2020/05922020
6. Ali Muttaleb Hasan, Noorhuzaimi Mohd Noor, Taha. H. Rassem, Ahmed Muttaleb Hasan. **Knowledge-Based Semantic Relatedness measure using Semantic features**, *International Journal of Advanced Trends in Computer Science and Engineering*, Vol.9, no.2, pp. 914-924, March-April 2020. https://doi.org/10.30534/ijatcse/2020/02922020
7. Schriml, Lynn Marie, et al. **Disease Ontology: a backbone for disease semantic integration,** *Nucleic acids research*, Vol 40, D1, pp. D940-D946.
8. Catia Pesquita, Daniel Faria, Andre´ O. Falca˜o, Phillip Lord, Francisco M. Couto. **Semantic similarity in biomedical ontologies,** *PLoS Computational Biology*, Vol 5, no. 7, pp. 1-12, July 2007.
9. L. Sweeney. **k-anonymity: a model for protecting privacy**. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, Vol.10, no 5, pp. 557-570, 2002.
10. C Liu, S Chen, S Zhou, J Guan, Y Ma. **A novel privacy preserving method for data publication**. *Information Sciences* 501 pp.421-435, 2019. https://doi.org/10.1016/j.ins.2019.06.022
11. X Xiao, Y Tao. **Anatomy: Simple and effective privacy preservation.** *Proceedings of the 32nd international conference on Very large data bases*. VLDB Endowment, 2006. pp. 139-150.
12. JW Byun, Y Sohn, E Bertino, N Li. **Secure anonymization for incremental datasets.** *Workshop on*

secure data management. *Springer, Berlin, Heidelberg*, pp. 48-63, 2006.

13. X Xiao, Y Tao. **M-invariance: Towards privacy preserving re-publication of dynamic datasets**. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, June 2007, pp. 689-700.
14. J Li, BC Ooi, W Wang. **Anonymizing streaming data for privacy protection.** In *2008 IEEE 24th International Conference on Data Engineering*, April 2008, pp. 1367-1369.
15. J Cao, B Carminati, E Ferrari. **CASTLE: Continuously anonymizing data streams.** *IEEE Transactions on Dependable and Secure Computing*, Vol.8, no.3, pp. 337-352, 2010.
16. B Zhou, Y Han, J Pei, B Jiang, Y Tao, Y Jia. **Continuous privacy preserving publishing of data streams.** In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, March 2009, pp. 648-659.
17. J Cao, P Karras, P Kalnis, KL Tan. **SABRE: A Sensitive Attribute Bucketization and REdistribution framework for t-closeness.** *The VLDB Journal*, Vol. 20, no. 1, pp. 59-81, 2011. https://doi.org/10.1007/s00778-010-0191-9
18. P Wang, J Lu, L Zhao, J Yang. **B-CASTLE: An efficient publishing algorithm for k-anonymizing data streams.** In *2010 Second WRI Global Congress on Intelligent Systems*. Vol. 2, pp. 132-136. Dec 2010.
19. P Wang, L Zhao, J Lu, J Yang. **SANATOMY: Privacy preserving publishing of data streams via anatomy.** In *2010 Third International Symposium on Information Processing IEEE*, October 2010, pp. 54-57.
20. H Zakerzadeh, SL Osborn. **Faanst: fast anonymizing algorithm for numerical streaming data.** *Data privacy management and autonomous spontaneous security*. Springer, Berlin, Heidelberg, pp.36-50, 2010.
21. S Kim, MK Sung, YD Chung. **A framework to preserve the privacy of electronic health data streams.** *Journal of biomedical informatics*, Vol.50, pp.95-106. April 2014.
22. R Lu, H Zhu, X Liu, JK Liu, J Shao. **Toward efficient and privacy-preserving computing in big data era.** *IEEE Network* Vol.28, no.4, pp. 46-50, 2014.
23. L Xu, C Jiang, J Wang, J Yuan, Y Ren. **Information security in big data: Privacy and data mining**. *Ieee Access* Vol.2, pp.1149-1176, 2014.
24. BCM Fung, K Wang, R Chen, PS Yu. **Privacy-preserving data publishing: A survey of recent developments.** *ACM Computing Surveys (Csur)*, Vol.*42*, *no.*4, pp.1-53, 2010. https://doi.org/10.1145/1749603.1749605