International Journal of Computer Applications (0975 - 8887)
Volume 178 - No.37, August 2019

Theoretical Model for Detecting Sensitive Data Items of Users in Data Publication

Charles R. Haruna
University of Electronic Science and Technology of China
No. 2006 , Xiyuan Ave, West Hi-Tech Zone
611731, Chengdu, China

MengShu Hou
University of Electronic Science and Technology of China
No. 2006 , Xiyuan Ave, West Hi-Tech Zone
611731, Chengdu, China

Barbie Eghan-Yartel
University of Cape Coast
Cape Coast, Ghana

ABSTRACT

Developments in current information technology are leading to the increased capture and storage of information about people and their activities. This raises serious concerns about the which data items are sensitive and how to detect these sensitive data items. Data privacy has become a very important concern in data publication in this modern era. The protection of data privacy depends on exactly what needs to be kept secret, thus, sensitive data. Protecting data privacy is a complicated task that takes into consideration what needs to be kept confidential. However, current privacy modeling techniques assume sensitive data items.
This paper considers the detection of sensitive data items in data publication for research purposes. We attempt to theoretically formalize a model for detecting sensitive data using a directed graph. We identify transitions that have a lot of sensitive data items published to them; critical transitions. Furthermore, the state that is most risky to the user to traverse in the graph, termed the "Risky State" was ascertained.

General Terms

Data Privacy

Keywords

Sensitive Data items, Data Publication, User Transition

1. INTRODUCTION

Data owners over the years have had problems publishing their data items (examples are medical records, personal photos, telephone numbers, salary) to access services from different states (examples are hospitals, companies, banks). Extraordinary amounts of data on individuals are being collected in this age by the trusted third party. When a data owner seeks a service from a state, their data items are published. The data are published for reasons such as offering services to the user, research purposes and legal cases. Unfortunately, once the data items are published, the owners can no longer count on the third party to publish data items that will not compromise their privacy. The issue of users protecting their privacy is a very vital concern that troubles them the most. A technique to detect sensitive data is needed to control the rate at which third parties publish data items to states that will weaken the data user's privacy. Individuals have different reasons for accessing a service. Publishing certain data items to some states will put the privacy of the data owner at risk. For example, Bob intends to give out his information to both the hospital and a manufacturing industry for medical care and employment respectively. In most cases, the information released to a medical facility via the trusted third party, for Bob to receive a better care, is different from the one released to the industry for employment. The same information on past medical history cannot be released to the manufacturing industry for employment. Some features of the same information released for medical care will not be published to secure an employment. How does the trusted third party be made aware of the data items to be published to different states without compromising the privacy of the user?
To the best of knowledge, there are no existing techniques that detect sensitive items of users, they all assume some data items are sensitive. Most of the techniques in previous works talked about using *k*-anonymity and improvements of *k*-anonymity techniques to protect the user's privacy [1]. [7][8] presented models for protecting user's data privacy based on the adversary's external knowledge. However, their methods of protecting the privacy of the individual's detected sensitive data items were determined only based on assumptions and sanitized techniques.
The goal and contributions of this research is to propose a technique to detect sensitive data and protect the owners in data publishing industry. We used a directed graph depicting the relationships of the user's data publication to explain this work. Furthermore, in the graph Critical transitions and Risky State were identified.

2. PROBLEM FORMULATION

In this section, question about in what situation; a data item published is sensitive, and will become a privacy leakage for a specific user, shall be answered. Most of the existing literature employs sanitized techniques such as data perturbation, de-identification, quasi-identifiers and anonymization. However on what basis or criteria are these identifiers said to be sensitive to the data owner, for them to be removed or sanitized before publishing?

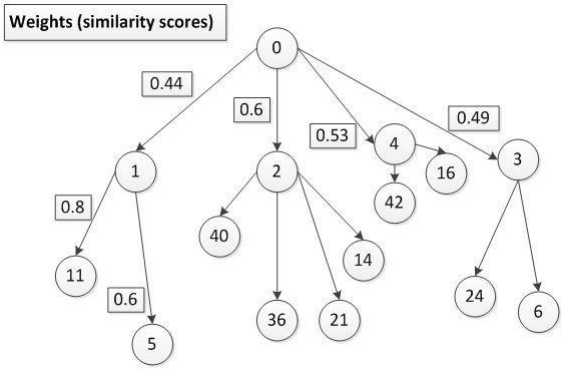

Fig. 1. A subgraph showing a data user's transitions.

2.1 Formal Definitions

Let the set of records be R = $(r_1, r_2, \ldots, r_n)$. A data owner, *U*, has a set of data items $D = \{d_1, d_2, \ldots, d_n\}$, where each item $d_i \epsilon$ R represents information to be published to different set of states, $S = \{s_1, s_2, \ldots, s_n\}$. But among these data items are sensitive ones that when published will compromise the user's privacy. Examples of a data publisher's states can be staying at home, going to the office, going to the bar or restaurant, to the shopping mall, to the transport yard, home of a friend or a relative, a holiday resort, to the sports complex or to the hospital. We call each trajectory by the user between states, a transition.

A graph $G = (V, E)$ consists of a set of vertices (nodes) $V = \{v_1, v_2, \ldots, v_n\}$ representing the set of records/states. A set of edges $E \subseteq V \times V$, i.e. the edge set is a subset of ordered pairs of distinct nodes representing the user's transition from a state to another. An edge $e(j, k) \subseteq E$ is called directed if $(j, k) \subseteq E$. A state *k* is said to be accessible from state *j* written as $j \rightarrow k$, if the system started in state *j* and has a non-zero probability of transitioning into state *k*. If there is a directed edge $j \rightarrow k$, state *j* is said to be an ancestor of state *k*, and *k* the descendant of *j*. Each edge e_g in set *E*, is associated with a non-negative weight, $w(e_g)$. A similarity based technique [17][18][19] is used on a pair of vertices to calculate a similarity value as the weight of the edges. In this work the similarity-based technique used is the Euclidean distance, D_E, defined [20] as:

$$D_E\left(\overrightarrow{t_a}, \overrightarrow{t_b}\right) = \left(\sum_{t=1}^{m} |w_{t,a} - w_{t,b}|^2\right)^{1/2} \quad (1)$$

Fig. 1, shows a subgraph with weights of user's transitions to states accessing different types of services .

2.1.1 Detection of Sensitive Data Items. A user *U* sends numerous requests for their data items to be published to different states for services. Some data items published on the transitions to the states may lessen or compromise the user's privacy depicting these data items as sensitive information relative to the transition being published on. Randomly assigned weights of publishing a data item in a database is $w(d_i)$ and ranges [0,1].

The pseudocode for detecting sensitive data items is shown in **Algorithm** 1. It inputs a directed graph $G = (V, E)$ and data items d_i from the dataset. Similarity values are then calculated for each transition as weights of the graph $w(S_j \rightarrow S_k)$. Random weights

Algorithm 1 Detecting Sensitive Data Items

Require: A directed graph $G = (V, E)$, data items d_i.
Ensure: sensitive data items d_i on a transition $(S_j \rightarrow S_k)$.
1: **generate** $w(S_j \rightarrow S_k)$ by using Euclidean distance similarity score.
2: **generate** random $w(d_i)$.
3: **calculate** set of $value(d_i, (S_j \rightarrow S_k))$ = $w(S_j \rightarrow S_k) \times w(d_i)$
4: **select** distinct $value(d_i, (S_j \rightarrow S_k))$.
5: **order** $value(d_i, (S_j \rightarrow S_k))$ desc.
6: **select** top 20% $value(d_i, (S_j \rightarrow S_k))$.
7: **set** *SD* $\leftarrow$ select random $value(d_i, (S_j \rightarrow S_k))$.
8: **while** $d_i \leq d_n$
9: **select** $(S_j \rightarrow S_k)$ to determine sensitive data items on.
10: **select** d_i.
11: **if** $value(d_i, (S_j \rightarrow S_k)) >=$ *SD*.
12: **save** d_i on $(S_j \rightarrow S_k)$ as sensitive data item.
13: **else** increase d_i by 1.
14: **end if**
15: **end while**
16: **return** d_i on $(S_j \rightarrow S_k)$ as sensitive data item.

$w(d_i)$ are assigned to the data items. Lines 3 computes the set of values of data items on a transition, $value(d_i, (S_j \rightarrow S_k))$ as $w(S_j \rightarrow S_k) \times w(d_i)$. The set of $value(d_i, (S_j \rightarrow S_k))$ is sorted in a descending order from largest to smallest. The set of top twenty (20)% from the sorted $value(d_i, (S_j \rightarrow S_k))$ is selected. To set the threshold value, the algorithm randomly selects a value from the set containing the top twenty (20)% $value(d_i, (S_j \rightarrow S_k))$. Finally, it compares $value(d_i, (S_j \rightarrow S_k))$ with the set value, *SD*. If $value(d_i, (S_j \rightarrow S_k)) >=$ *SD*, the data items d_i are returned as sensitive data on the relative transition $(S_j \rightarrow S_k)$. Otherwise, d_i is not sensitive and when published on $(S_j \rightarrow S_k)$ to access services will not compromise privacy of the user.

2.1.2 Critical Transition of a user. In data publication, the critical path of a user from a start node to its descendant nodes is defined as the transition with possibly the most sensitive data items detected on. Thus, sensitive data items on each transition are detected, then the transition with the highest number of sensitive data items is identified as the critical transition.

2.1.3 Risky State. From the graph a "Risky State" of the user's transitions shall be identified. The "Risky State" is assumed to be the most dangerous state and the weak link of the graph because a lot of sensitive data items are published to it, thus it is vulnerable to attacks by adversaries and hence, will compromise the privacy of the user.

2.1.4 Complexity Analysis. Our algorithm can detect a user?s sensitive data items only with a certain level of confidence. Confidence is a number between 0 and 1 that denotes the probability that an item is detected as sensitive. In this work, it could be assumed that capturing confidence is a property. Together with the accuracy (efficiency), the algorithm determines an item?s sensitivity with respect to privacy ? that is, how detectable a data item is captured with a particular level of confidence and at a given level of accuracy. For the data items, the higher their sensitivity, the more detectable they are. Assume a data item d_i, has an accuracy that corresponds to a discrete subset of size *n* data items and its detection confidence is *p*. We can define a data item?s sensitivity as the

Table 1. Setting of Threshold Values.

Experiments	*SD*
Experiment 1	0.19
Experiment 2	0.07
Experiment 3	0.05
Experiment 4	0.14

reduction in uncertainty (entropy) after the data item?s detection confidence for a given accuracy is known:

$$sensitivity\,(d_i) = -\sum_{1}^{n} log\frac{1}{n} - \left[(-plogp) - \sum_{1}^{n-1}\frac{1-p}{n-1}log\frac{1-p}{n-1}\right] \quad (2)$$

3. EXPERIMENTS AND DISCUSSIONS

According to research, there are no suitable methods for identifying sensitive data items on transitions that this work can be compared with.
The experiments setup was as follows:
Data used were from a Directed Gnutella P2P network from August 6, 2002 and constructed a directed graph with 8717 nodes and 31525 edges representing states where a user can access a service and the transitions respectively. We also used 2 datasets from the UCI machine learning; Adult Data Set and Census Income Data Set with 16281 and 32561 records respectively. We pruned the number of records in the datasets to match the number of nodes in the directed graph. The attributes (data items) from the datasets were $D = \{$age, workclass, fnlwgt, education, education-num, marital-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week, native-country$\}$. The binary attribute salary class (salary above or below 50,000) was not retained. The graphs were scaled using natural log scale (base e logarithmic scale).

3.1 Sensitive Data Items

In this work, to analyze the efficiency, four (4) experiments were performed and sensitive data items on each transition were identified. Table **??** shows the threshold (*SD*) values returned in each of the experiments. Experiments 1 and 4 returned huge *SD* values compared to experiments 2 and 3. Figs. 2 and 3 show the number of sensitive data items detected on the user's transitions. The transitions $S_j \rightarrow S_k$ were represented with numbers on the horizontal axis. In experiments 2 and 3 on both data sets, a lot of sensitive data items were detected on numerous transitions as compared to experiments 1 and 4 on both datasets. This implies that the lower the threshold value *SD*, the more sensitive data items will be detected on many transitions.

3.2 Critical Transitions

The critical transition of each start state to its reachable states were determined. Sensitive data items on all transitions were detected.
Table **??** shows the sets of sensitive data items detected on some few selected transitions using Census Income Data set. For example, in identifying critical transition, transitions $S_{17} \rightarrow S_{938}$, $S_{17} \rightarrow S_{1608}$, $S_{17} \rightarrow S_{2786}$ and $S_{17} \rightarrow S_{3597}$ had no sensitive data items detected on them in all four experiments, but $S_{17} \rightarrow S_{4038}$ had sensitive data items detected on, only in experiment 3. It implies that in experiments 1, 2 and 4, there were no critical transitions identified but in experiment 3, the critical transition was $S_{17} \rightarrow S_{4038}$.

3.3 Risky State

The number of sensitive data items published to each state from all of its ancestors, were accumulated and the state with the highest number identified as the "Risky State".
Figs. 4 and 5 show graphs of accumulated sensitive data items published to states. Table **??** shows Risky states identified in all experiments under both data sets. On both datasets, the graphs are heavy-tailed at the lower bottom. Using Adult data set, S_{176}, S_{356}, S_{176} and S_{356} were identified as Risky states in experiments 1, 2, 3 and 4 respectively. Using Census Income data set, S_{176}, S_{67}, S_{356} and S_{176} were also identified as Risky states in experiments 1, 2, 3 and 4 respectively. In experiments 2 and 3 using both data sets, a lot of sensitive data items were detected as compared to experiments 1 and 4. We can thus conclude that the lower the threshold value *SD*, the more sensitive data items will be detected on a transition to be identified as risky.

3.4 Occurrences of Data Items as Sensitive

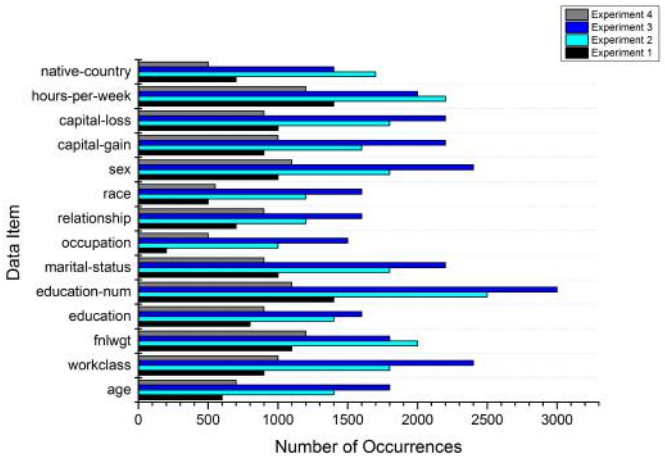

Fig. 6. Number of occurrences of data items as sensitive: adult data set.

Figs. 6 and 7 show the number of occurrences of data items as sensitive using adult data set and census income data set respectively. Showed in both figures, experiments 2 and 3 had data items detected as sensitive on transitions the highest. Experiments 1 and 4 have the lowest occurrences of data items as sensitive. This implies that the lower *SD* value, the higher the number of data items being detected as sensitive on both data sets and vice versa.

3.5 Efficiency of the technique

Efficiency of the algorithm are measured in terms of the running time or speed of detecting sensitive data items and identifying risky states.
In Fig. 8, it can clearly be seen that detecting sensitive data items (first graph) and identifying risky states (second graph) are faster in experiments 1 and 4 than in 2 and 3. This is due to the fact that the

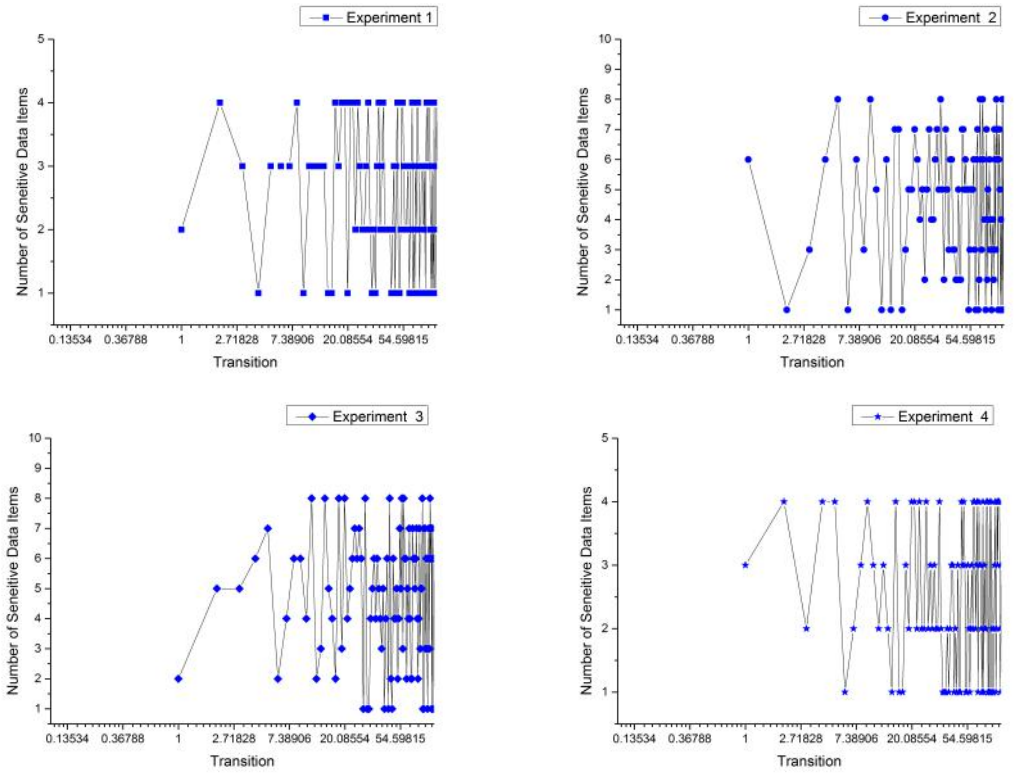

Fig. 2. Number of sensitive data items on transitions using Adult data set.

Table 2. Sets of sensitive data items on selected transitions using Census Income data set.

Transition	Sensitive Data Items			
	Experiment 1	Experiment 2	Experiment 3	Experiment 4
$S_{17} \rightarrow S_{938}$	{None}	{None}	{None}	{None}
$S_{17} \rightarrow S_{1608}$	{None}	{None}	{None}	{None}
$S_{17} \rightarrow S_{2786}$	{None}	{None}	{None}	{None}
$S_{17} \rightarrow S_{3597}$	{None}	{None}	{None}	{None}
$S_{17} \rightarrow S_{4038}$	{None}	{None}	{age, workclass, fnlwgt}	{None}
$S_{1111} \rightarrow S_{4067}$	{occupation, relationship}	{relationship, capital-gain}	{age, workclass, capital-gain}	{workclass, fnlwgt}
$S_{7421} \rightarrow S_{7949}$	{None}	{marital-status, occupation, relationship}	{age, marital-status, occupation, relationship}	{None}

threshold values in 1 and 4 are higher than in 2 and 3. Therefore, the higher the threshold values, the faster the proposed algorithm in this work executes.

4. RELATED WORK

From intensive research done, there are no works that provide a model to detect sensitive information, that when released will compromise the user's privacy. In most of the works, ascertaining sensitive information was based on assumption and these sets of sensitive data satisfied all of the user's transitions in all possible states. Therefore, some works on data privacy that made assumptions of sensitive items are briefly discussed.

Sweeney et al. [1] proposed a formal model called *k*-anonymity and a set of accompanying policies for protecting the user's data during data publishing. Pei, Jian, et al. [3] suggested a model to generate only one anonymized version that will satisfy multiple users reducing problems with the multiple quasi-identifiers. Agrawal et al. [6] stated that data from clients is randomized in order to preserve privacy. The authors designed a model for privacy-preserving computation of multidimensional aggregates on data partitioned across multiple clients before it is integrated at the server side. In the work of Bhat et al [12], a graph theoretical approach based on *k*-partitioning of graphs, which paves way to creation of a complex decision tree classifier, organised in a prioritised hierarchy, was proposed to address problems causing a big privacy breach. Navarro-Arribas et al [13] addressed the problems of protecting sensitive data items in query logs by ensuring the anonymity of the users in the logs. They presented the anonymization of query logs using microaggregation. Their proposed method ensured the *k*-anonymity of the users in the query log, while preserving its utility. In [16] Babu, K.S et al stated methods proposed to enforce *k*-

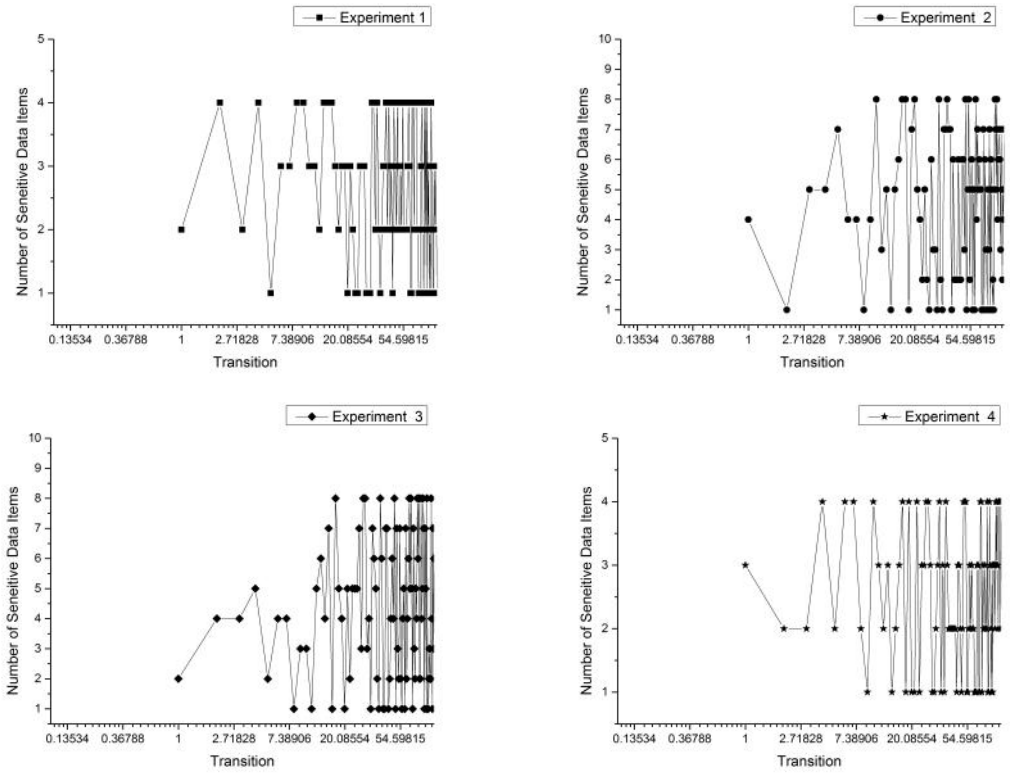

Fig. 3. Number of sensitive data items on transitions using Census Income data set.

Table 3. Risky States and number of detected data items.

Experiment	Adult Data Items		Census Income Data Items	
	Risky State	Detected Items	Risky State	Detected Items
1	176	167	176	167
2	356	323	67	296
3	176	280	356	301
4	356	177	176	164

anonymity notably Samarati?s algorithm and Sweeney?s Datafly, which both adhere to full domain generalisation, require a trade off between computing time and information loss. Thus they proposed an improved greedy heuristic for enforcing *k*-anonymity with full domain generalisation. Iyengar [14] addressed the importance of preserving the anonymity of the individuals or entities during the data dissemination process. Gkoulalas-Divanis et al [15] presented a survey on algorithms that have been proposed for publishing structured patient data, which protect the disseminated data against several privacy threats, while the data remained useful for subsequent analysis tasks.

In works [2][7][8], the authors presented models for defining data privacy based on the adversary's external knowledge. These privacy frameworks allow data owners to safeguard their data against attackers called realistic adversaries when publishing their data. Kapur et al. [9] based their work on modeling as the basis of data management of uncertainty, stating that the uncertainty of the data in *k*-anonymity is caused by generalization. Jiang et al. [5] described a theoretical model for privacy control in context aware-systems, which has the ability to infer revealing information from loosely related personal data has even more troubling implications for individual privacy.

5. CONCLUSION

In this paper, a theoretical model for detecting the sensitive data items of a user's set of data on various transitions to different states was proposed. It showed that if users publishing their set of data to access services at different states can determine the sensitive data items, then they can securely publish the non-sensitive data items without privacy being invaded or compromised. A directed graph was constructed and the critical transitions were determined and the "Risky State" of the entire graph was identified as well. The results from the experiments and security analysis proved to some extent that the proposed algorithm for detecting sensitive data is efficient when implemented by users in publishing their sets of data. With the detection of sensitive data items modeled, modeling privacy definition in future shall be addressed.

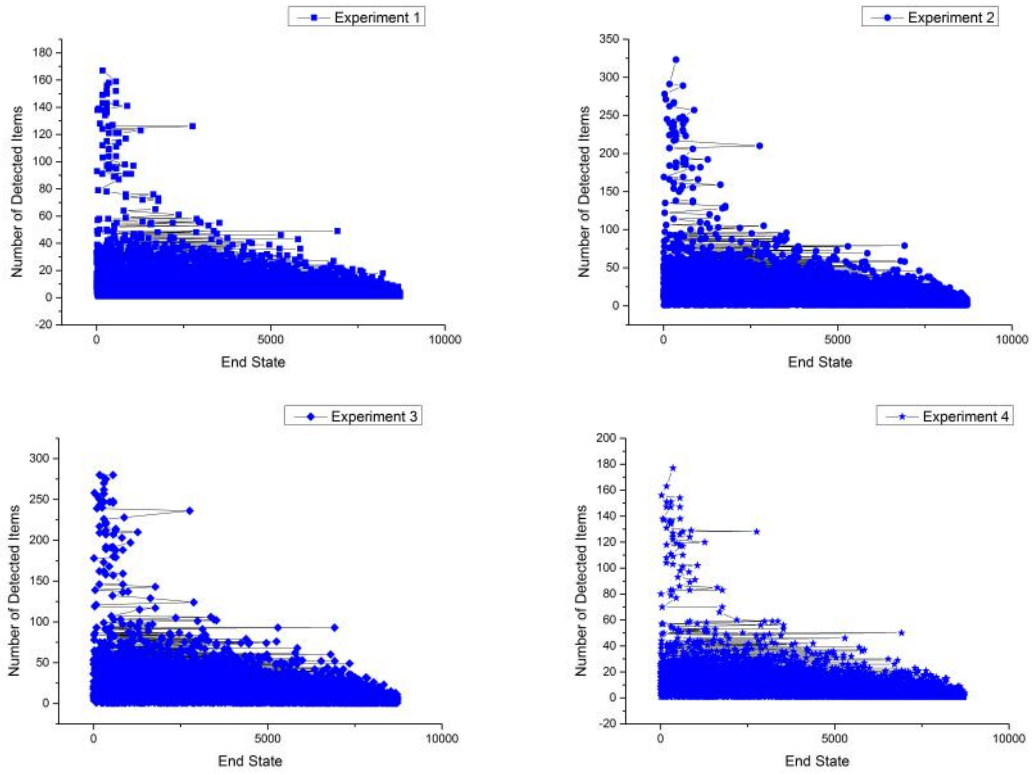

Fig. 4. Risky state using Adult data set.

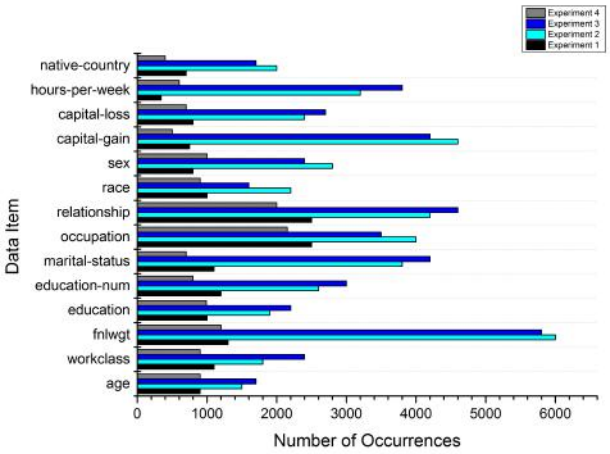

Fig. 7. Number of occurrences of data items as sensitive: census income data set.

Acknowledgment

We thank Anthony Padua Bangdome for providing insightful comments that helped improve the paper. We are most grateful.

6. REFERENCES

[1] Sweeney, Latanya. "k-anonymity: A model for protecting privacy." International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10.05 (2002): 557-570.

[2] Machanavajjhala, Ashwin, Johannes Gehrke, and Michaela Gtz. "Data publishing against realistic adversaries." Proceedings of the VLDB Endowment2.1 (2009): 790-801.

[3] Pei, J., Tao, Y., Li, J. and Xiao, X., 2009, March. Privacy preserving publishing on multiple quasi-identifiers. In Data Engineering, 2009. ICDE'09. IEEE 25th International Conference on (pp. 1132-1135). IEEE.

[4] Wu, Jiawei, and Guohua Liu. "Modeling the Uncertain Data in the K-anonymity Privacy Protection Model." Computational Intelligence and Security (CIS), 2011 Seventh International Conference on. IEEE, 2011.

[5] Jiang, Xiaodong, and James Landay. "Modeling privacy control in context-aware systems." Pervasive Computing, IEEE 1.3 (2002): 59-63.

[6] Agrawal, Rakesh, Ramakrishnan Srikant, and Dilys Thomas. "Privacy preserving OLAP." Proceedings of the 2005 ACM SIGMOD international conference on Management of data. ACM, 2005.

[7] Chen, Bee-Chung, Kristen LeFevre, and Raghu Ramakrishnan. "Privacy skyline: Privacy with multidimensional adversarial knowledge." Proceedings of the 33rd international conference on Very large data bases. VLDB Endowment, 2007..

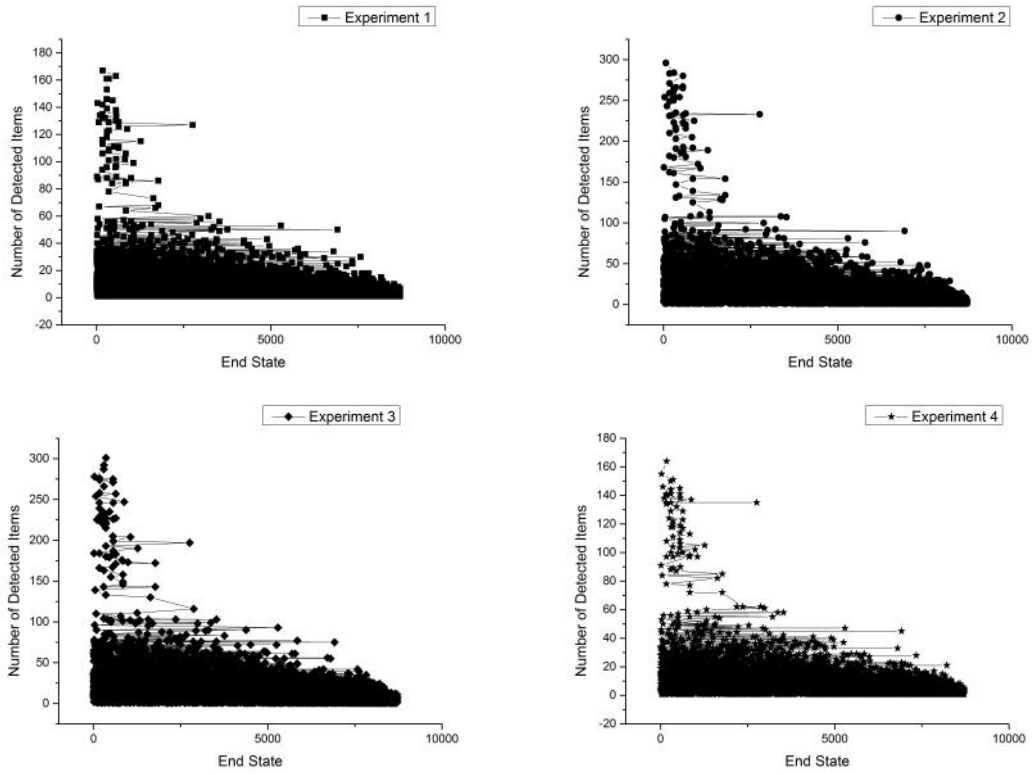

Fig. 5. Risky state using Census Income data set.

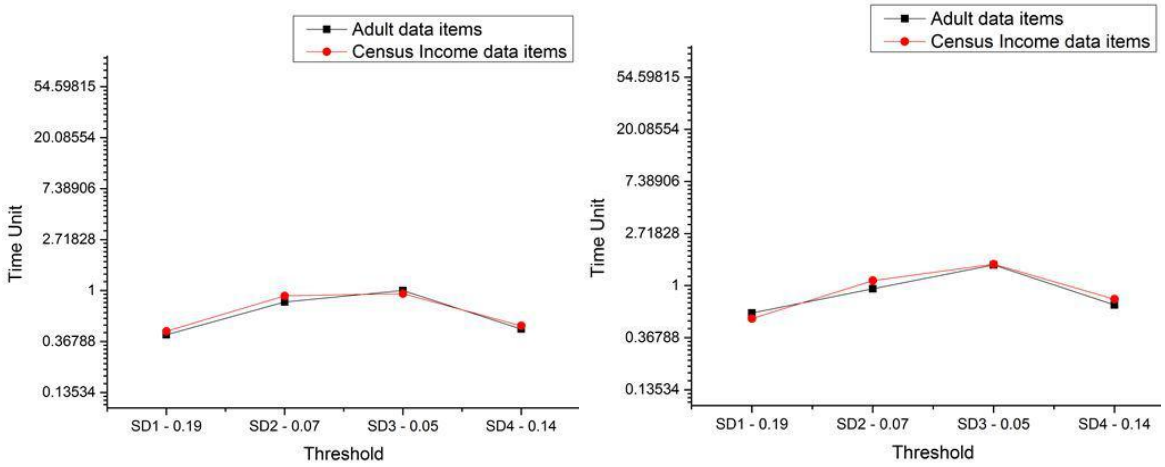

Fig. 8. Efficiency: Detecting data items as sensitive and Identifying risky state.

[8] Martin, D.J., Kifer, D., Machanavajjhala, A., Gehrke, J. and Halpern, J.Y., 2007, April. Worst-case background knowledge for privacy-preserving data publishing. In Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on (pp. 126-135). IEEE.

[9] Kapur, Eshan, Parveen Kumar, and Sahil Gupta. "Proposal of a two way sorting algorithm and performance comparison with existing algorithms." International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol 2 (2012).

[10] Shokri, R., Theodorakopoulos, G., Le Boudec, J.Y. and Hubaux, J.P., 2011, May. Quantifying location privacy. In Security and privacy (sp), 2011 ieee symposium on (pp. 247-262). IEEE.

[11] Xiao, Yonghui, and Li Xiong. "Protecting locations with dif-

ferential privacy under temporal correlations." Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. ACM, 2015.

[12] Bhat, T.P., Karthik, C. and Chandrasekaran, K., 2015. A Privacy Preserved Data Mining Approach Based on k-Partite Graph Theory. Procedia Computer Science, 54, pp.422-430.

[13] Navarro-Arribas, G., Torra, V., Erola, A. and Castell-Roca, J., 2012. User k-anonymity for privacy preserving data mining of query logs. Information Processing & Management, 48(3), pp.476-487.

[14] Iyengar, V.S., 2002, July. Transforming data to satisfy privacy constraints. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 279-288). ACM.

[15] Gkoulalas-Divanis, A., Loukides, G. and Sun, J., 2014. Publishing data from electronic health records while preserving privacy: a survey of algorithms. Journal of biomedical informatics, 50, pp.4-19.

[16] Babu, K.S., Reddy, N., Kumar, N., Elliot, M. and Jena, S.K., 2013. Achieving k-anonymity Using Improved Greedy Heuristics for Very Large Relational Databases. Transactions on Data Privacy, 6(1), pp.1-17.

[17] R. J. Bayardo, Y. Ma, and R. Srikant. Scaling up all pairs similarity search. In WWW, pages 131?140, 2007.

[18] S. Chaudhuri, V. Ganti, and R. Kaushik. A primitive operator for similarity joins in data cleaning. In ICDE, page 5, 2006.

[19] Xiao, C., Wang, W., Lin, X., Yu, J.X. and Wang, G., 2011. Efficient similarity joins for near-duplicate detection. ACM Transactions on Database Systems (TODS), 36(3), p.15.

[20] Huang, A., 2008, April. Similarity measures for text document clustering. In Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand (pp. 49-56).