Efficient Clustering of Transactional Data for Privacy-Preserving Data Publishing

Vartika Puri, Parmeet Kaur, and Shelly Sachdeva

Abstract Transactional data is set-valued data which is generated from retail store, healthcare, etc. The data needs to be published to extract useful information from the data. The data contain some sensitive information about the individual, and if leaked, then it will cause serious implications to the privacy of an individual. Therefore, it is required to protect the user's privacy on the published data while ensuring the data should remain useful for analysis purpose. The paper proposes efficient clustering method using ant-colony-based clustering algorithm to bring similar transactions in same equivalence class/cluster. Finally, we can achieve privacy with minimal information loss. The approach has been tested on INFORMS dataset and compared with the Disassociation. The result shows that the more information is preserved as compared to Disassociation approach.

Keywords Ant colony clustering · Clustering · Privacy-preserving data publishing · Relative error

1 Introduction

Privacy-preserving data publishing (PPDP) is the emerging area where data is published to the third party while preserving privacy of individuals whose data is contained in the published data. Transactional data is the real-world dataset generated by the widely used applications such as retail store and healthcare.

V. Puri (✉) · P. Kaur
Department of Computer Science & Engineering and Information Technology, Jaypee Institute of Information Technology, Noida, India
e-mail: vartika.puri@jiit.ac.in

P. Kaur
e-mail: parmeet.kaur@jiit.ac.in

S. Sachdeva
Department of Computer Science, National Institute of Technology, Delhi, India
e-mail: shellysachdeva@nitdelhi.ac.in

K. Khanna et al. (eds.), *Cyber Security and Digital Forensics*,
Lecture Notes on Data Engineering and Communications Technologies 73,
https://doi.org/10.1007/978-981-16-3961-6_14

The dataset is of the form {id: a_1, $a_2 \ldots a_n$} where id denotes the identity of user and a_1, $a_2 \ldots a_n$ denote the set of attributes belongs to the user.

The primary requirement in PPDP is the protection of identity disclosure [1]. k^m-anonymity [1] is the model which ensures the protection of identity disclosure in transactional data with minimal information loss. K^m-anonymity model ensures every m no. of items should occur in k transactions. There are numerous methods are available to achieve k^m-anonymity. Disassociation [2] is the method which is based on bucketization to achieve k^m-anonymity while incurring less information loss. There are three phases in Disassociation—(i) Horizontal partitioning, (ii) Vertical partitioning, and (iii) Refining.

In first phase, the similar transactions are put into one cluster. In the second phase, the cluster is converted into k^m-anonymous record chunks by placing infrequent item combinations in different record chunks. In any of the privacy-preserving data publishing methods, there are two steps, first, make equivalence classes of similar records, and second, apply anonymization method to anonymize the records in the equivalence class. Therefore, if there are more similar records in a cluster, then there will be less modifications to achieve the desired privacy level and data utility will be maintained. Thus, creation of good equivalence classes/clusters is the main step of PPDP.

Ant colony optimization (ACO) is the technique used by insects existing in adjacent colonies in the search for food. If a source of food is found by any ant team/colony, then some teams of ants follow diverse paths searching this food, leaving behind pheromone trail, a chemical usually excreted by animals, and is of great importance for insects. The pheromone trail directs the other ants, and with its help, other ants follow the way laid down by the ants moving in front of them. Few ant teams will reach the food source prior to the other teams due to the fact that they would have traversed the shortest path, and then they will follow the same path to go back to their colony before the other ant teams. Now this shortest path will have the pheromone trails as the team have traversed this path and came back before other team following another path; therefore, probability of other teams taking the same path over other paths is much higher, lest some other paths (better) are discovered by another teams. Pheromone trail of the shortest path is expected to be more concentrated than the other paths.

Inspiration of the ant-based clustering algorithm comes from the clustering of corpses and larval sorting events found in real ant colonies. Deneubourg et al. [3] have first started the study in this field. He has proposed a basic model in which objects in clusters are randomly moved, picked up, and dropped as per the similarity found in surrounding objects. LF algorithm proposed by Lumer and Faieta [4] which is an extension of basic model, which is applicable for numerical datasets. In this algorithm, ants are considered as agents who travel in a four-sided grid in a random fashion. These agents pick up, transport, and drop the data items scattered within this environment. Operations (picking and dropping) are executed as per the similarity and density of the data items found in the ants' neighborhood: either isolated or data items surrounded by dissimilar ones are likely to be picked up by ants, and ants have a tendency to drop them near the comparable ones. This is how

elements are clustered and sorted in the grid. The ant colony clustering algorithm are more flexible, robust, and decentralized [5–7] than traditional methods.

The paper proposed the use of ant colony clustering algorithm on transactional dataset for making optimized clusters of similar transactions.

The rest of the paper is structured as follows: Sect. 2 presents the related work in this domain. Section 3 proposes application of ant colony clustering algorithm for efficient clustering of transactional data. The results of the implementation of the proposed algorithm and its comparison with related approach are discussed in Sect. 4. Lastly, we summarize and conclude the work followed by future work.

2 Related Work

The application of ACO to solve the clustering problem was introduced by Shelokar et al. [8]. Firstly, the sample data is represented by each string element, and its content signify the cluster number which the sample data allotted to. Each ant in the ACO at that time builds a solution on the basis of string representation. As per [9], the ant algorithm can be segregated into two sets to achieve clustering, ant-based sorting, and ACO based clustering. Ant-based sorting algorithm uses 2D grid. As per that algorithm, foremost the objects are scattered randomly. Afterward, objects dissimilar to its neighborhood are picked up by artificial ants and transfer it to the cluster containing similar objects. The proposed solution was also used in the studies [10–13]. Though a defined cluster number is not required in the beginning by ant-based sorting, the processing time will be high as it requires post-processing to recognize the generated clusters [9]. This was proven in few prior studies where the analysis of the cluster number should be done visually once the clustering is completed [12]. ACO based clustering is another ant algorithm for clustering which uses the same idea of solution string to denote the clustering solution. The solution string is built on each iteration and assessed by the objective function to discover the most optimal one. Although a defined cluster number is a prerequisite, ACO based clustering is more efficient in computation than ant-based sorting. Also, once the clustering is done, it does not require post-processing [9]. Apart from ACO, some of the proposed clustering algorithms also practice the same concept of solution string as ACO based clustering [14–16]. ACOC [17] is the first implementation of ACO based clustering. After that, ACOC has been enhanced in some studies such as [18] which revised the original ACOC by keeping the identified best solution as the initial solution for the next iteration and adding the ability to determine the optimal cluster number automatically using Jaccard index. The study has demonstrated that the algorithm takes more time to run. The research [17] have adopted another methodology by combining the ACOC with k-means algorithm. In this, the ACO explores the initial solution generated by k-means. However, the algorithm was only tested on financial services data processing. Besides, in research [19], ACO based clustering concept was used; however, it builds the classification model according to the training dataset which is clustered using ACO. The fast ant

colony optimization for clustering (FACOC) improves the efficacy of computation in ACOC [20]. In FACOC, the threshold value is used to define whether a cluster number turn out to be common for an object once it is being selected for multiple times. If a cluster number for an object turns out to be common, then that cluster number will be selected without computing the probability in the next iteration for that particular object. With this, the redundant computations can be reduced, enhancing the execution time. In addition, local search will not affect the object with common cluster number. However, the result indicates that FACOC outputs have inferior clustering quality than ACOC.

3 Proposed Approach: Application of Ant Colony Clustering Algorithm for Efficient Clustering of Transactional Data

We propose an algorithm in which efficient partitions are created using ant colony clustering algorithm and then utilizes VERPART algorithm [21] to finally achieve k^{m}-anonymity. The clusters are initialized using HORPART algorithm [21]. HORPART algorithm selects the most frequent item "a" in the dataset and splits the dataset into two partitions, the records which contain "a" come in one partition and the rest of the records come in another partition. The process of splitting like this will continue till we get the partitions of predefined size, say P_1, $P_2 \ldots P_n$.

At the beginning, an ant m is associated to a partition P_I, and during the iterations, the ant will select the most dissimilar transaction d_i of partition P_I, and another partition P_J is selected at random using a roulette-wheel with probability p_{IJ}, where p_{IJ} depends on the pheromone trail and a local heuristic. Ant will assign d_i to the partition P_J. The value of the pheromone trail is modified according to the rule.

$$\tau_{xy} = (1 - \rho)\tau_{xy} + \Delta\tau_{xy}^{k} \tag{1}$$

where τ_{xy} is the amount of pheromone deposited for a state transition from partition x to partition y, ρ is the pheromone evaporation coefficient where $\rho \in [0,\ 1]$ and $\Delta\tau_{xy}^{k}$ is the amount of pheromone deposited by k^{th} ant.

$$\Delta\tau_{xy}^{k} = \begin{cases} 1 & \text{if ant } k \text{ transfer a transaction from } x \text{ to } y \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The local heuristic or short-term visibility is defined as.

$$\eta_{IJ} = \frac{A \cap B}{A \cup B} \tag{3}$$

where A and B are the transactions. If the transaction A is more similar to the transaction B of particular partition, then it gives a big value in order to influence in the probability of assigning it to the partition. If ant m is at partition I, partition J is chosen with probability

$$p_{XY}^{k} = \frac{\left(\tau_{XY}^{\alpha}\right)\left(\eta_{XY}^{\beta}\right)}{\sum_{z \in \text{allowed}_X}\left(\tau_{XZ}^{\alpha}\right)\left(\eta_{XZ}^{\beta}\right)} \tag{4}$$

To find whether the obtained solution is better than previous or not, the Jaccard similarity is calculated.

$$B(P) = \sum_{I=1}^{n} JS(P_I) \tag{5}$$

Then, dissimilar transaction d of partition I is assigned to the partition J.

```
Algorithm: Proposed algorithm

Input:  number of iterations t_max, the number of ants M; the initial
value of pheromone τ_0; α; β; ρ; Original dataset D.
Output: Anonymized dataset D'.
Initialize τ_IJ= τ_0;
Calculate η according to (3)
Initialize the probabilities p
Initialize partitions P_1,…,P_m using HORPART
For n=1 to t_max:
 For t=1 to M do:
   For each p in P:
     Find the most dissimilar transaction d_i in the partition p
     Select a partition P_J according to (4)
     Assign d_i to the partition P_J
   End For
   Calculate the cost B(P)according to (5),
   if B(P) > B(P_I-1) then
      keep B(P)
   else
       keep B(P_I-1)
   Update τ according to (1) and (2)
 End For
End For
For each partition p ∈ P do:
Apply VERPART[21] on p.
```

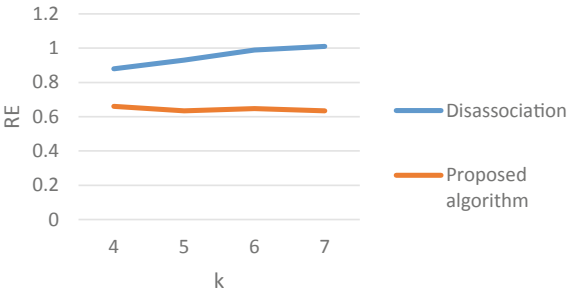

Fig. 1 Relative error

4 Implementation and Results

The proposed approach is implemented in Python and tested on INFORMS dataset.[1] In ACO, there are five parameters that needs to be fixed. In the literature [22], following conditions have been specified on selecting these values:

- β has to be larger than α; so that destination cluster should be chosen based on local heuristic, i.e., similarity with the destination cluster instead of deposited pheromone.
- α, $\beta <= 1$ is better than α, $\beta > 1$;
- $\rho = 0.8$ is better than $\rho = 0.7$ and $\rho = 0.9$ decided by set of experiments.

Considering the above conditions, the following values are considered:

- M = number of clusters
- $t_{max} = 100$
- $\alpha = 0.8$
- $\beta = 1$
- $\rho = 0.8$
- $\tau_0 = 0.001$

The results are analyzed in terms of information loss while achieving privacy-preserving data publishing. To evaluate information loss, we have used relative error measure [21] which measure the loss in the association of items occurred while anonymization shown in Fig. 1. The relative error is calculated for different values of k (= 4, 5, 6, 7) and for the anonymized data using proposed algorithm and Disassociation algorithm, respectively. The result shows that if data is anonymized using the proposed algorithm, it gives lower values of relative error for each k than Disassociation algorithm. The lower value of relative error shows that more items are still associated in the anonymized data and, thus, preserves data utility.

[1] https://sites.google.com/site/informsdataminingcontest/data.

5 Conclusion

The paper has clearly shown that if we can create the equivalence classes/clusters which have similar records result in less information loss due to anonymization process. Thus, data utility maintained. The proposed algorithm uses ant colony optimization to further refine the equivalence classes/clusters; it shows the significant improvements in equivalence class and, thus, reduces the information loss cause by anonymization. The proposed approach has been tested on INFORMS dataset, and it gives lower relative error than Disassociation algorithm. In future, the applicability of other nature-inspired algorithm can be tested and compared.

References

1. Terrovitis, M., Mamoulis, N., Kalnis, P.: Privacy-preserving anonymization of set-valued data. VLDB Endowment, 115–125 (2008)
2. Loukides, G., Liagouris, J., Gkoulalas-Divanis, A., Terrovitis, M.: Disassociation for electronic health record privacy. J. Biomed. Inform. **50**, 46–61 (2014)
3. Deneubourg, J.L., Goss S., Franks, N., Sendova-Franks A., Detrain, C., Chretien, L.: The dynamics of collective sorting: robot-like ants and ant-like robots. In: Meyer, J.A., Wilson, S. (eds) Proceedings of the 1st International Conference on Simulation of Adaptive Behaviour: From Animals to Animals, pp. 356–365, MIT Press, Cambridge, Mass, USA (1991)
4. Lumer, E., Faieta, B.: Diversity and adaptation in populations of clustering ants. In: Proceedings of the Third International Conference on Simulation of Adaptive Behaviour: From Animals to Animals, pp. 501–508, MIT Press, Cambridge, Mass, USA (1994)
5. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm intelligence: from natural to artificial system. Oxford University Press, New York, NY, USA (1999)
6. Ghosh, A., Halder, A., Kothari, M., Ghosh, S.: Aggregation pheromone density based data clustering. Inf. Sci. **178**(13), 2816–2831 (2008)
7. Gao, W., Yin, Z.X.: Modern intelligent bionics algorithm and its applications. Science Press, Beijing, China (2011)
8. Shelokar, P., Jayaraman, V.K., Kulkarni, B.D.: An ant colony approach for clustering. Anal. Chim. Acta **509**(2), 187–195 (2004)
9. Jabbar, A.M., Ku-Mahamud, K.R., Sagban, R.: Ant-based sorting and ACO-based clustering approaches: a review. In: 2018 IEEE Symposium on Computer Applications and Industrial Electronics (ISCAIE) (2018)
10. Gao, W.: Improved Ant Colony Clustering Algorithm and Its Performance Study. Computational Intelligence and Neuroscience **2016**(19) (2016)
11. Yang, Y., Kamel, M.S.: An aggregated clustering approach using multi-ant colonies algorithms. Pattern Recogn. **39**(7), 1278–1289 (2006)
12. Kuo, R.J., Wang, H.S., Hu, T.L., Chou, S.H.: Application of ant K-means on clustering analysis. Comput. Math. Appl. **50**(10–12), 1709–1724 (2005)
13. Korurek, M., Nizam, A.: A new arrhythmia clustering: technique based on ant colony optimization. J. Biomed. Inform. **41**(6), 874–881 (2008)
14. Tao, W.A., Ma, Y., Tian, J.H., Li, M.Y., Duan, W.S., Liang, Y.Y.: An improved ant colony clustering algorithm. In: Zhu, R., Ma, Y. (eds) Information Engineering and Applications, vol. 154. Lecture Notes in Electrical Engineering, pp. 1515–1521, Springer, London, UK (2012)
15. Inkaya, T., Kayaligil, S., Ozdemirel, N.E.: Ant colony optimization based clustering methodology. Appl. Soft Comput. J. **28**, 301–311 (2015)

16. Chaturvedi, A., Green, P.E., Carroll, J.D.: k-Modes clustering. J. Classif. **18**(1), 35–55 (2001)
17. Handl, J., Knowles, J., Dorigo, M.: Ant-based clustering and topographic mapping. Artif. Life **12**(1), 35–62 (2006)
18. Wu, B., Zheng, Y., Liu, S., Shi, Z.Z.: CSIM: a document clustering algorithm based on swarm intelligence. In: Proceedings of the Congress on Evolutionary Computation (CEC '02), pp. 477–482, Honolulu, Hawaii, USA (2002)
19. Yang, Y., Kamel, M.S., Jin, F.: Topic discovery from document using ant-based clustering combination. In: Web Technologies Research and Development—APWeb of Lecture Notes in Computer Science, pp. 100–108, Springer, Berlin, Germany (2005)
20. Ramos, G.N., Hatakeyama, Y., Dong, F., Hirota, K.: Hyperbox clustering with ant colony optimization (HACO) method and its application to medical risk profile recognition. Appl. Soft Comput. J. **9**(2), 632–640 (2009)
21. Terrovitis, M., Liagouris, J., Mamoulis, N., Skiadopoulos, S.: Privacy preservation by disassociation. VLDB Endowment **5**, 944–955 (2012)
22. Trejos, J., Murillo, A., Piza, E.: Clustering by Ant Colony Optimization. Classification, Clustering, and Data Mining Applications (2004)