Privacy-Maintaining in Outsourced Transaction Databases from Association Rules Mining

Ms. P.B.Deokate
PG Student ME (Information Technology)
Dattakala Group of Institutions Faculty of Engineering, Swami-chincholi(Bhigwan), Tal-Daund,District-Pune,India
E-mail: pallavi.deokate18@gmail.com

Prof.Ms. M.M.Waghmare
Ass.Prof.Department of Computer Engineering
Dattakala Group of Institutions Faculty of Engineering, Swami-Chincholi(Bhigwan) Tal-Daund, District-Pune,India
E-mail: monawaghmare25@gmail.com

***Abstract* — Outsourcing is a trend that is becoming more common in information technology and other industries for services that have usually been regarded as intrinsic to managing a business. An organization (data owner) can outsource its mining needs like resources or expertise to a third party service provider (server). However, both the association rules and the items of the outsourced transaction database are private property of data owner. The client encrypts its data, sends data and mining queries to the server, and accepts the original patterns from the encrypted patterns received from the server to maintain the privacy. The problem of outsourcing transacti on database within a corporate privacy framework is studied in this paper. We generate the synthetic data set called Transaction Database(TDB) and device an encryption decryption scheme to protect privacy in outsourced TDB. Our scheme ensures that each transformed data is different with respect to the attacker's previous information. The experimental results on real transaction database prove that our techniques are scalable, efficient and maintain privacy.**

Index Terms — Privacy-preserving outsourcing, Association rule mining

[I] Introduction

Outsourcing is an arrangement in which one company provides services for another company that could also be or usually have been provided in-house. In some cases, the complete information management of a corporation is outsourced, as well as designing and business analysis yet because the installation, management, and conjugation of the network and workstations. A corporation like IBM manages IT services for a corporation like Xerox to the apply of hiring contractors and temporary workplace employees on a private basis. for instance, associate enterprise may source its IT management as a result of it's cheaper to contract a 3rd party to try to to therefore than it'd be to make its own in-house IT management team. Or a corporation may source all of its knowledge storage desires as a result of it doesn't wish to shop for and maintain its own knowledge storage devices [1]. Most giant organizations solely source some of any given IT operate. Their aim is sanctionative organizations with restricted process resources and/or data processing experience to source their data processing must a 3rd party service supplier [2], [3], [4]. Wong et al. [2] was one in all the first works on defensive against the frequency-based attack within the data processing outsourcing situation. They introduced the thought of fake transactions to defend against the frequency-based attack; but, it absolutely was lacking a proper theoretical analysis of privacy guarantees, and has been shown to be blemished terribly recently in [5], wherever a way for breaking the planned secret writing is given. Therefore, in our previous and preliminary work [6], we have a tendency to planned to resolve this downside by mistreatment k-privacy.

Related work is represented in section II. The architecture of proposed system is described in section III. Then the implemention details such as data set generation, encryption/decryption scheme is given in section IV. Section V discusses the privacy analysis of our scheme over large datasets. Finally, we conclude the paper and discuss directions for future analysis in Section VI.

[II] Related Work

The particular problem attacked in our paper is outsourcing of pattern mining within company privacy. Not solely the underlying data but together the well-mined results do not appear to be meant for sharing. Once the server possesses background and conducts attacks on it basis, it's unable to guess the correct candidate item or itemset like a given cipher item or item set. Another issue is secure multiparty mining over distributed datasets. This body of labor was pioneered by [7] and has been followed up by

several papers since [8]. The divided data cannot be shared and will keep personal but the results of mining on the union of the knowledge square measure shared among the participants, by implies that of multiparty secure protocols [9]–[11]. they do not believe third parties. This approach part implements company privacy; but it's too weak for our outsourcing downside, as a result of the following patterns square measure disclosed to multiple parties. The works that square measure most related to ours square measure [2] and [12]. The success of the attacks in recent paper [5] the most depends on the existence of distinctive, common, and fake things, printed in [2]; our theme does not manufacture any such things. Tai et al. [12] assumed the offender is attentive to actual frequency of single things, equally to u. s. of America. Compared with these two works, our theme can invariably bring home the bacon obvious privacy guarantee with relevance the background of assailant.

[III] System Architecture

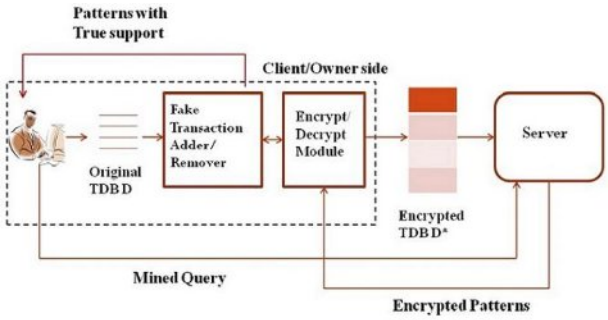


Fig.1. Proposed System Architecture.

In this paper, our goal is to develop an encryption scheme that permits formal privacy guarantees, and to validate this model over big-scale real-life transaction databases (TDB). The proposed system architecture is shown in Fig.1.The client/owner encrypts its data using encrypt module. Before encryption the fake transactions are added to the original data by client. The details of encrypt/decrypt (E/D) module will be explained in following sections. The server conducts data mining with the association rules mining and sends the (encrypted) data to the owner. The encrypted patterns are send back to the client where it get decrypted by E/D module. Then the added fake transactions are removed by fake transaction adder/remover module. Our encryption scheme has the property that can identify the true supports. The E/D module obtains the true identity of the returned patterns as well as their true supports.

Contributions:

1] First, our first task is to generate synthetic data set (TDB). The transaction database (TDB) will include number of items that are purchased by the client in particular period. Depending on this data, an attack model is defined for an attacker that will make the background knowledge in precise. Our notion of privacy requires that, for each ciphertext item, there are at least k−1 distinct cipher items that are indistinguishable from the item regarding their supports.

2] Second, we have developed an encryption scheme, known as advanced encryption standards (AES). It includes the different method called Frugal, RobFrugal, generate noise table & hash table. The fake transactions are added to the data before it's encryption. The encrypted data is upload to the server for mining.

3] Third, to allow the server to conduct data mining using association rule mining and then server sends mined results to the owner.

4] Fourth, the returned encrypted data is decrypted using decrypt module. To recover the true patterns and their correct support by E/D module, we propose that it creates and keeps a compact structure, called synopsis. We also provide the E/D module with an efficient strategy for incrementally maintaining the synopsis against updates in the form of appends.

[IV] Implementation

Problem studied: Given a plain database D, construct a k-private cipher database D * by using substitution ciphers and adding fake transactions such that from the set of frequent cipher patterns and their support in D* sent to the owner by the server, the owner can reconstruct the true frequent patterns of D and their exact support. Additionally, we would like to minimize the space and time incurred by the owner in the process and the mining overhead incurred by the server.

A. Synthetic Data Set Generation (TDB)

Let I = i1, ..., in be the set of items and D = t1, ..., tm a TDB of transactions, each of this is a set of items. We denote the support of an itemset S ⊆ I as suppD(S) and the frequency by freqD(S). Recall that freqD(S) = suppD(S)/|D|. For each item i, suppD(i) and freqD(i) denote, respectively, the individual support and frequency of i. The function suppD(.), projected

over items, is called the item support table of D represented in tabular form i.e. support table in Fig. 2(b). The well-known frequent pattern mining problem [13] is: given a TDB D and a support threshold σ, find all itemsets whose support in D is at least σ.

TDB
Apple
Orange Apple
Apple Orange
Banana Orange
Apple Milk
Apple Chocolate
Banana

(a) TDB

Item	Support
Apple	5
Orange	3
Banana	2
Milk	1
Chocolate	1

(b) Item Support Table

Fig.2. Example of TDB and its support table. (a) TDB. (b) Item support table.

B. Encryption

We let D denote the first TDB that the owner has. to safeguard the identification of individual things, the owner applies associate encoding operate to D and transforms it to D*, the encrypted information.

1. Frugal

The sparing technique consists of grouping along cipher things into teams of k adjacent things within the item support table in decreasing order of support, ranging from the foremost frequent item e1. Assume e1, e2, . . . , en is that the list of cipher things in descendent order of support (with relation to D), the teams created by sparing square measure , , and so on. The last cluster, if but k in size, is incorporate with its previous cluster. we tend to denote the grouping obtained victimisation the on top of definition as Gfrug. for instance, take into account the instance TDB and its associated (cipher) item support shown in Fig. 2. For k = 2, Gfrug has 2 groups: and . This corresponds to the partitioning teams shown in Table I(a). Thus, in D*, the support of e4 are going to be dropped at that of e2; and also the support of e1 and e3 dropped at that of e5.

2. RobFrugal

To fix the privacy vulnerabilities of sparing, we tend to introduce the RobFrugal grouping technique.
Given a TDB D and its sparing grouping Gfrug = (G1, ...,Gm), the grouping technique RobFrugal consists in modifying the teams of Gfrug by repetition the subsequent operations, till no cluster of things is supported in D:

1) Choose the smallest j ≥ one such suppD(Gj) > 0;

2) Notice the item i ∈ Gj such, for the smallest amount frequent item i of Gj we tend to have: suppD(Gj \ ∪) = 0; and

3) Swap i with i' within the grouping.

For example, given the item support table in Fig. 2, the grouping illustrated in Table I(b), obtained by exchanging e4 and e5 within the 2 teams of sparing, is currently strong.

3. Noise Table

In the *RobFrugal* encryption scheme, the output of grouping can be represented as the noise table. It extends the item support table with an extra column "Noise" indicating, for each cipher item *e*, the difference among the support of the most frequent cipher item in *e*'s group and the support of *e* itself, as reported in the item support table. the noise table obtained with *RobFrugal* is reported in Table II(a).

4. Hash Table

In order to implement the synopsis efficiently, we use a hash table generated with a minimal perfect hash function [14].The hash tables for the items of nonzero noise in Table II(a) are shown in TableII(b).

5. Generate Fake Transactions

Given a noise table specifying the noise $N(e)$ needed for each cipher item *e*, we generate the fake transactions as follows. First, we drop the rows with zero noise, corresponding to the most frequent items of each group or to other items with support equal to the maximum support of a group. Second, we sort the remaining rows in descending order of noise.

TABLE I
Grouping with k=2
(a) Frugal

Item	Support
e2	5
e4	3
e5	2
e1	1
e3	1

(b) RobFrugal

Item	Support
e2	5
e5	2
e4	3
e1	1
e3	1

TABLE II
Noise Table & It's Hash Table
(a) Noise Table for k=2

Item	Support	Noise
e2	5	0
e5	2	3
e4	3	0
e1	1	2
e3	1	2

(b) Hash tables of items of nonzero noise in (a)

	Table 1
0	<e5,1,2>
1	<e3,2,0>

	Table 2
0	<e1,2,0>

C. Decryption

The client receives frequent patterns mined over D*. Synopsis allows computing the actual support of every pattern.

Item	Support	Noise
e_2	5	0
e_5	2	3
e_4	3	0
e_1	1	2
e_3	1	2

	Table1
0	$\langle e_5, 1, 2\rangle$
1	$\langle e_3, 2, 0\rangle$

	Table2
0	$\langle e_1, 2, 0\rangle$

Fake Transaction

{e5}

{e1} {e1}

The Real Support is calculated as:

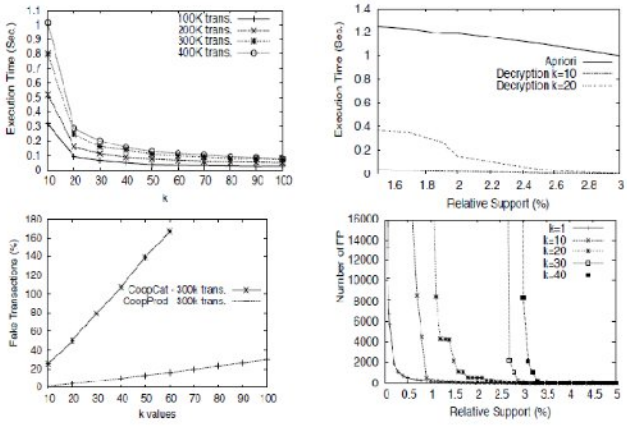
$RS(\{\mathbf{e5}\}) = suppD^* - suppD^* \setminus D = 5 - (1 + 2) = 2$

$RS(\{\mathbf{e5, e3}\}) = suppD^* - suppD^* \setminus D = 2 - (2 + 0) = 0$

[V] Result Analysis

- Item-based attack
  - **RobFrugal** guarantees the k-privacy against the item-based attack (***prob(e)*** <= ***1/k***)
- Itemset-based attack
  - **RobFrugal** guarantees the k-privacy against the itemset-based attack (***prob(E)*** <= ***1/k***)

Client and Server Overhead: Coop Data



On Coop dataset for k=10 we have:

- 5% of transactions have exactly a crack probability 1/10
- 95% of transactions have a probability strictly smaller than 1/10
- 90% have a probability strictly smaller than 1/100
- sNo single transaction contains any pattern consisting exactly of the items in a group created by RobFrugal.

[VI] Conclusion

In this paper, we have a studied the problem of (corporate) privacy-maintaining mining of frequent patterns on an encrypted outsourced transaction information. we've got thought of that the wrongdoer is aware of the domain things|of things} and their precise frequency and may use this data to spot cipher items and cipher itemsets. AN secret writing theme, known as RobFrugal, is planned that's supported 1–1 substitution ciphers for things and adding fake transactions. It makes use of a compact abstract of the fake transactions from that truth support of strip-mined patterns from the server is expeditiously recovered. we have a tendency to conjointly planned a method for progressive maintenance of the abstract against updates. The preliminary experiments on massive real information are given. Currently, our privacy analysis is predicated on the belief of equal probability of

candidates. It might be fascinating to reinforce the framework and therefore the analysis by appealing to scientific discipline notions like excellent secrecy [14]. Moreover, our work considers the ciphertext-only attack model, within which the wrongdoer has access solely to the encrypted things. We are going to investigate secret writing schemes which will resist such privacy vulnerabilities. we have a tendency to are fascinated by exploring the way to improve the RobFrugal rule to attenuate the amount of spurious patterns.

ACKNOWLEDGMENT

I wish to express my sincere thanks to our Principal, HOD and Professors and staff members of Computer Engineering Department at Dattakala faculty of Engineering, Swami Chincholi, Bhigawan. Last but not the least, I would like to thank all my Friends and Family members who have always been there to support and helped me to complete this research work.

References

[1] Fosca Giannotti, Laks V. S. Lakshmanan, Anna Monreale, Dino Pedreschi, and Hui (Wendy) Wang, "Privacy-Preserving Mining of Association Rules From Outsourced Transaction Databases," in IEEE SYSTEMS JOURNAL, VOL. 7, NO. 3, SEPTEMBER 2013.

[2] W. K. Wong, D. W. Cheung, E. Hung, B. Kao, and N. Mamoulis, "Security in outsourcing of association rule mining," in *Proc. Int. Conf. Very Large Data Bases*, 2007, pp. 111–122.

[3] C. Clifton, M. Kantarcioglu, and J. Vaidya, "Defining privacy for data mining," in *Proc. Nat. Sci. Found. Workshop Next Generation Data Mining*, 2002, pp. 126–133.

[4] L. Qiu, Y. Li, and X. Wu, "Protecting business intelligence and customer privacy while outsourcing data mining tasks," *Knowledge Inform. Syst.*, vol. 17, no. 1, pp. 99–120, 2008.

[5] I. Molloy, N. Li, and T. Li, "On the (in)security and (im)practicality of outsourcing precise association rule mining," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2009, pp. 872–877.

[6] M. Kantarcioglu and C. Clifton, "Privacy-preserving distributed mining of association rules on horizontally partitioned data," *IEEE Trans. Knowledge Data Eng.*, vol. 16, no. 9, pp. 1026–1037, Sep. 2004.

[7] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 439–450.

[8] S. J. Rizvi and J. R. Haritsa, "Maintaining data privacy in association rule mining," in *Proc. Int. Conf. Very Large Data Bases*, 2002, pp. 682–693.

[9] F. Giannotti, L. V. Lakshmanan, A. Monreale, D. Pedreschi, and H. Wang, "Privacy-preserving data mining from outsourced databases," in *Proc. SPCC2010 Conjunction with CPDP*, 2010, pp. 411–426.

[10] B. Gilburd, A. Schuster, and R. Wolff, "k-ttp: A new privacy model for large scale distributed environments," in *Proc. Int. Conf. Very Large Data Bases*, 2005, pp. 563–568.

[11] P. K. Prasad and C. P. Rangan, "Privacy preserving birch algorithm for clustering over arbitrarily partitioned databases," in *Proc. Adv. Data Mining Appl.*, 2007, pp. 146–157.

[12] C. Tai, P. S. Yu, and M. Chen, "K-support anonymity based on pseudo taxonomy for outsourcing of frequent itemset mining," in *Proc. Int. Knowledge Discovery Data Mining*, 2010, pp. 473–482.

[13] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. Int. Conf. Very Large Data Bases*, 1994, pp. 487–499.

[14] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. Cambridge, MA: MIT Press, 2001.