International Journal of Engineering Research and General Science Volume 2, Issue 6, October-November, 2014
ISSN 2091-2730

Privacy-Preserving in Outsourced Transaction Databases from Association Rules Mining

Ms. Deokate Pallavi B.[1], Prof. M.M. Waghmare[2]

[1] Student of ME (Information Technology), DGOFFE, University of Pune, Pune
[2] Asst. Professors, Department of Computer Engineering, DGOIFE, Bhigwan University of Pune, Pune

***Abstract* —** Data mining-as-a-service has been selected as considerable research issue by researchers. 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 data owner encrypts its data, send data and mining queries to the server, and accepts the true patterns from the encrypted patterns received from the server to protect the privacy. The problem of outsourcing transaction database within a corporate privacy framework is studied in this paper. We propose an attack model based on previous knowledge and devise a scheme for privacy preserving outsourced data mining. 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 protect privacy.

Index Terms — Privacy-preserving outsourcing, Association rule mining

Introduction

With the arrival of cloud computing and its model for IT services supported the net and large knowledge centers, the outsourcing of knowledge and computing services is getting a completely unique connation, that is predicted to skyrocket within the close to future. Business intelligence and data discovery services, square measure expected to be among the services amenable to be externalized on the cloud, thanks to their knowledge intensive nature, further because the complexness of knowledge mining algorithms. Thus, the paradigm of mining and management of knowledge as service can presumptively grow as quality of cloud computing grows [1]. This can be the information mining-as-a-service paradigm, geared toward sanctionative organizations with restricted process resources and/or data processing experience to source their data processing has to a 3rd party service supplier [2], [3]. We adopt a conservative frequency-based attack model within which the server is aware of the precise set of things within the owner's knowledge and to boot, it conjointly is aware of the precise support of each item within the original knowledge. In this paper, our goal is to plan encryption scheme that permits formal privacy guarantees to be well-tried, and to validate this model over large-scale real-life dealing databases (TDB)

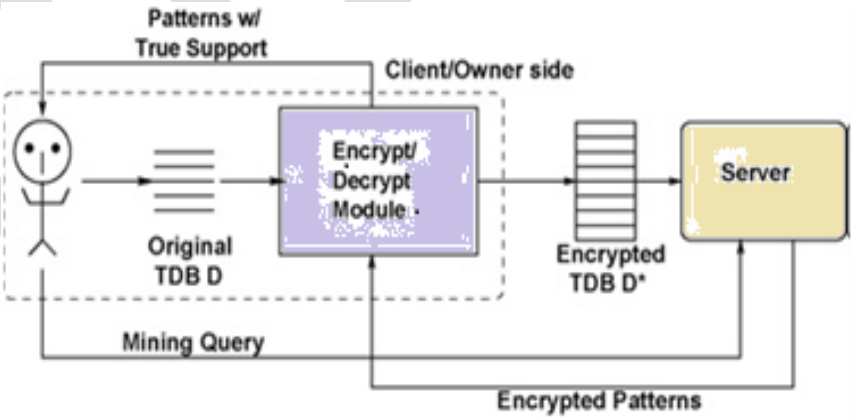

Fig.1. Architecture of mining-as-service paradigm

The design behind our model is illustrated in Fig.1.The client/owner encrypts its information victimisation associate degree encrypt/decrypt (E/D) module. Whereas the main points of this module are going to be explained in Encryption/Decryption Section. The server conducts data processing and sends the (encrypted) patterns to the owner. Our coding theme has the property that came supports aren't true supports. The E/D module recovers truth identity of the came patterns additionally their true supports

Contributions:

First, associate attack model is outlined for attacker and makes the background the attacker might possess precise. Our notion of privacy needs that, for every cipher text item, there area unit a minimum of k−1 distinct cipher things that area unit indistinguishable from the item relating to their supports.

Second, we have developed an encryption scheme, known as RobFrugal. The E/D module will use to remodel consumer information before it's shipped to the server.

Third, to permit the E/D module to recover verity patterns and their correct support, we have a tendency to propose that it creates and keeps a compact structure, referred to as synopsis. We have a tendency to additionally offer the E/D module with an economical strategy for incrementally maintaining the abstract against updates within the type of appends.

Related work is represented within the next section. The pattern mining task is reviewed then. Our privacy model is given in next section. Then next section develops the encryption/decryption theme we tend to use. Finally, we conclude this paper and discuss directions for future analysis in last Section.

Related Work

The particular drawback attacked in our paper is outsourcing of pattern mining inside company privacy. Not only the underlying knowledge however conjointly the mined results don't seem to be meant for sharing. Once the server possesses background and conducts attacks thereon basis, it is unable to guess the proper candidate item or itemset equivalent to a given cipher item or item set.

Another issue is secure multiparty mining over distributed datasets. Knowledge on that mining is to be performed is partitioned and distributed among many parties. This body of labor was pioneered by [7] and has been followed up by many papers since [8]. The partitioned off knowledge can't be shared and should stay personal however the results of mining on the union of the information are shared among the participants, by means that of multiparty secure protocols [9]–[11]. They don't think about third parties. This approach partly implements company privacy; however it's too weak for our outsourcing problem, because the ensuing patterns are disclosed to multiple parties.

The works that are most associated with ours are [2] and [12]. A recent paper [5] has formally evidenced that the encryption system in [2] are often broken while not victimization context-specific info. The success of the attacks in recent paper [5] the main depends on the existence of distinctive, common, and faux things, outlined in [2]; our theme doesn't produce any such things. Tai et al. [12] assumed the wrongdoer is aware of actual frequency of single things, equally to United States of America. Compared with these 2 works, our theme will invariably bring home the bacon obvious privacy guarantee with regard to the background of attacker.

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

(a)

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

(b)

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

Pattern Mining Task

We let I = i1,…, in be the set of things and D = t1, ...,tm a TDB of transactions, every of that could be a set of things. We tend to denote the support of associate itemset S ⊆ I as suppD(S) and also the frequency by freqD(S). Recall that freqD(S) =

suppD(S)/|D|. For every item i, suppD(i) and freqD(i) denote, severally, the individual support and frequency of i. The perform suppD(.), projected over things, is additionally known as the item support table of D described in tabular kind [Fig. 2(b)]. The well-known frequent pattern mining downside [13] is: given a TDB D and a support threshold σ, notice all itemsets whose support in D is a minimum of σ.

Privacy Model

We let D denote the initial TDB that the owner has. to safeguard the identification of individual things, the owner applies AN cryptography perform to D and transforms it to D*, the encrypted info. We tend to discuss with things in D as plain things the encrypted info. We tend to discuss with things in D as plain things and things in D* as cipher things.

A. Adversary Knowledge

The server in Nursing trespasser who gains access to that might possess some information victimization that they'll conduct attacks on the encrypted information D*. We tend to generically sit down with Associate in Nursingy of those agents as a wrongdoer. We tend to adopt a conservative model and assume that the wrongdoer is aware of precisely the set of (plain) things I within the original TDB D and their true supports in D, i.e., suppD(i), ∀ i ∈ I. The wrongdoer might have access to similar information from a competitory company, might scan printed reports, etc.

B. Attack Model

The data owner (i.e., the corporate) considers actuality identity of: 1) each cipher item; 2) each cipher transaction; and 3) each cipher frequent pattern because the material possession that ought to be protected. We take into account the subsequent attack model.

1) **Item-based attack:** ∀ cipher item e ∈ E, the offender constructs a collection of candidate plain things Cand(e) ⊂ I. The likelihood that the cipher item e may be broken prob(e) = 1/|Cand(e)|.
2) **Set-based attack:** Given a cipher itemset E, the offender constructs a collection of candidate plain itemsets Cand(E), wherever ∀X ∈ Cand(E), X ⊂ I, and |X| = |E|. The likelihood that the cipher itemset E may be broken prob(E) = 1/|Cand(E)|.

Encryption/Decryption Scheme

A. Encryption

In this section, we have introduced the cryptography theme, referred to as RobFrugal that transforms a TDB D into its encrypted version D*. Our theme is constant with reference to k > zero and consists of 3 main steps: 1) Using 1–1 substitution ciphers for every plain item; 2) Using a selected item kgrouping method; and 3)Using a way for adding new pretend transactions for achieving k-privacy. The made pretend transactions area unit else to D to make D*, and transmitted to the server. A record of pretend transactions, i.e., DF = D*\D, is hold on by the E/D module within the variety of a compact precis, as mentioned in Sections C and D.

B. Decryption

When the consumer requests the execution of a pattern mining question to the server, specifying a minimum support threshold σ, the server returns the computed frequent patterns from D *. Clearly, for each itemset S and its corresponding cipher itemset E, we've that suppD(S) ≤ suppD* (E). For every cipher pattern E came back by the server beside suppD* (E), the E/D module recovers the corresponding plain pattern S. It must reconstruct the precise support of S in D and choose on this basis if S may be a frequent pattern. to attain this goal, the E/D module adjusts the support of E by removing the result of the faux transactions. suppD(S) = suppD* (E) − suppD*\D(E). This follows from the very fact that support of Associate in Nursing itemset is additive over a disjoint union of dealing sets. Finally, the pattern S with adjusted support is unbroken within the output if suppD(S) ≥ σ. The calculation of suppD*\D(E) is performed by the E/D module victimisation the precis of the faux transactions in D* \D.

C. *Grouping Items for k-Privacy*

Given the things support table, many ways will be adopted to cluster the things into teams of size k. We tend to begin from an easy grouping methodology known as sparing. The item support table is sorted in descendant order of support and check with cipher things during this order as e1, e2, etc. Assume e1, e2. . . en is that the list of cipher things in descendant order of support (with reference to D), the teams created by sparing square measure. The last cluster, if but k in size, is united with its previous cluster. We tend to denote the grouping obtained mistreatment the on top of definition as Gfrug. For instance, contemplate the instance TDB and its associated (cipher) item support shown in Fig. 2. For k = 2, Gfrug has 2 groups. This corresponds to the partitioning teams shown in Table I(a). Thus, in D* the support of e4 are delivered to that of e2; and also the support of e1 and e3 delivered to that of e5.

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 stinting, and is currently robust: none of the 2 teams, thought of as itemsets, is supported by any dealings in D.

TABLE I
Groping 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 and Its 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

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

Table 2
< *e1,2,0*>

D. Constructing Fake Transactions

Given a noise table specifying the noise N(e) required for every cipher item e, we have a tendency to generate the pretend transactions as follows. First, we have a tendency to drop the rows with zero noise, admire the foremost frequent things of every cluster or to different things with support capable the most support of a bunch. Second, we have a tendency to kind the remaining rows in descendent order of noise.Continuing the instance, think about cipher things of nonzero noise in Table II(a). The subsequent 2 pretend dealings square measure generated: 2 instances of the transaction and one instance of the dealing.

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.

CONCLUSION

In this paper, we studied the problem of (corporate) privacy-preserving mining of frequent patterns on an encrypted outsourced transaction database. We have considered that the attacker knows the domain of items and their exact frequency and can use this knowledge to identify cipher items and cipher itemsets. An encryption scheme, called *RobFrugal*, is proposed that is based on 1–1 substitution ciphers for items and adding fake transactions. It makes use of a compact synopsis of the fake transactions from which the true support of mined patterns from the server can be efficiently recovered. We also proposed a strategy for incremental maintenance of the synopsis against updates. This method is robust against an adversarial attack based on the original items and their exact support.

Currently, our privacy analysis is based on the assumption of equal likelihood of candidates. It would be interesting to enhance the framework and the analysis by appealing to cryptographic notions such as perfect secrecy [14]. Moreover, our work considers the ciphertext-only attack model, in which the attacker has access only to the encrypted items. It could be interesting to consider other attack models where the attacker knows some pairs of items and their cipher values. We will investigate encryption

schemes that can resist such privacy vulnerabilities. We are also interested in exploring how to improve the *RobFrugal* algorithm to minimize the number of spurious patterns.

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] 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.

[4] 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.

[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] 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.
[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] 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.

[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] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst.Tech. J.*, vol. 28, pp. 656–715, 1948