Computational Complexity of Association Rule Hiding Algorithms

Kshitij Pathak
MIT, Ujjain
er.k.pathak@gmail.com

Aruna Tiwari
SGSITS, Indore
atiwari@sgsits.ac.in

Narendra S. Chaudhari
IIT, Indore
nsc183@gmail.com

ABSTRACT

Data mining services require accurate input data for their results to be meaningful, but privacy concerns may influence users to provide spurious information. To preserve client privacy in the data mining process, a variety of techniques based on random perturbation of data records have been proposed recently. One known fact which is very important in data mining is discovering the association rules from database of transactions where each transaction consists of set of items. Two important terms support and confidence are associated with each of the association rule. Actually any rule is called as sensitive if its disclosure risk is above a certain privacy threshold. Sometimes we do not want to disclose a sensitive rule to the public because of confidentiality purposes.

This paper is extension of work done in [1]. In [1] a reduction of 3-SAT problem from optimal sanitization in association rule hiding is presented. This paper proves that optimal sanitization in association rule hiding is NP-Complete. The proofs are based on reduction from 3-SAT.

Keywords

3-SAT; Association Rule; NP-Complete; Data mining;, Data Sanitization

1. INTRODUCTION

Data mining is a technique that helps to extract important data from a large database. It is the process of sorting through large amounts of data and picking out relevant information through the use of certain sophisticated algorithms as shown in Figure 1. As more data is gathered, with the amount of data doubling every twenty months, data mining is becoming an increasingly important tool to transform this data into information.

Data mining can be used to classify data into predefined classes (classification), or to partition a set of patterns into disjoint and homogeneous groups (clustering), or to identify frequent patterns in the data, in the form of dependencies among concepts-attributes (associations). In general, data mining promises to discover unknown information. If the data is personal or corporate data, data mining offers the potential to reveal what others regard as private. This is more apparent as Internet technology gives the opportunity for data users to share or obtain data about individuals or corporations. In some cases, it may be of mutual benefit for two corporations (usually competitors) to share their data for an analysis task. However, they would like to ensure their own data remains private. In other words, there is a need to protect private knowledge during a data mining process. This problem is called Privacy Preserving Data Mining (PPDM). The remainder of this paper is organized as follows. First we review current approaches addressing data mining and security. We then present a formulation of our problem and show that the optimal solution to it is NP-Complete.

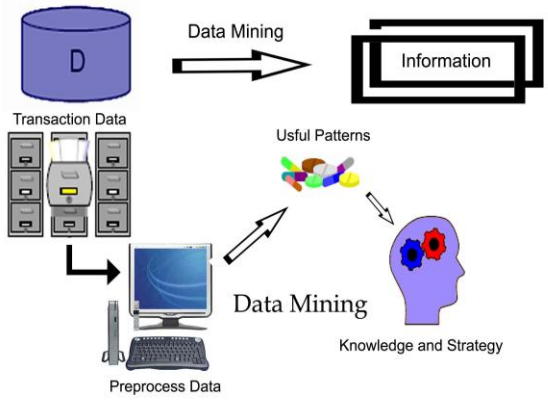

Fig. 1:Data Mining

The problem of association rule hiding was first probed in [2]. After that, many approaches were proposed. Roughly, they can fall into two groups: data sanitization data modification approaches (data modification for short) and knowledge sanitization data reconstruction (data reconstruction) approaches.

2. DATA MODIFICATION & RECONSTRUCTION APPROACHES

Data modification methods hide sensitive association rules by directly modifying original data. Most of the early methods belong to this track.

Data sanitization is a process that is used to promote sharing of transactional databases among organizations while alleviating concerns of individual organizations by preserving confidentiality of their sensitive knowledge in the form of sensitive association rules. It hides the frequent itemsets corresponding to the sensitive association rules that contain sensitive knowledge by modifying the sensitive transactions that

contain those itemsets. This process is guided by the need to minimize the impact on the data utility of the sanitized database by allowing mining as much as possible of the non-sensitive knowledge in the form non-sensitive association rules from the sanitized database.

The basic idea of data modification approaches is the so-called data sanitization. They hide sensitive association rules by directly modifying, or we say, sanitizing the original data D, and get the released database D' directly from D, Most of the existing methods belong to this data modification prosperous track. According to different modification means, it can be further classified into : Data-Distortion techniques and Data-Blocking techniques. However, data modification approaches cannot control the hiding effects intuitively as the sanitization is performed on data level.

Data-Distortion is based on data perturbation or data transformation, and in particular, the procedure is to change a selected set of 1-values to 0-values (delete items) or 0-values to 1- values (add items) if we consider the transaction database as a two-dimensional matrix. Its aim is to reduce the support or confidence of the sensitive rules below the user predefined security threshold. Early data distortion techniques adopt simple heuristic-based sanitization strategies like Algo1a/Algo1b/Algo2a, Algo2b/Algo2c [3], Naive/MinFIA/MaxFIA/IGA [4], RRA/RA and SWA [5,6]. Different heuristics determine different selection strategies on which transactions are to be sanitized and which items are to be victims, which are two core issues affecting the hiding effects in the algorithms. Subsequent techniques like WSDA/PDA [7] and Border-Based [8] advanced the simple heuristics to heuristic greedy (local optimal) strategies trying to greedily select the modifications with minimal side effects on data utility.

Data-Blocking [9] is another data modification approach for association rule hiding. Instead of making data distorted (part of data is altered to false), blocking approach is implemented by replacing certain data items with a question mark "?". The introduction of this special unknown value brings uncertainty to the data, making the support and confidence of an association rule become two uncertain intervals respectively. At the beginning, the lower bounds of the intervals equal to the upper bounds. As the number of "?" in the data increases, the lower and upper bounds begin to separate gradually and the uncertainty of the rules grows accordingly. When either of the lower bounds of a rule's support interval and confidence interval gets below the security threshold, the rule is deemed to be concealed.

Data reconstruction methods put the original data aside and start from sanitizing the so-called "knowledge base". The new released data is then reconstructed from the sanitized knowledge base. This idea is first depicted in [10]. They give a coarse Constraint-based Inverse Itemset Lattice mining procedure (CIILM) for hiding sensitive frequent itemsets.

The main difference is their method aims at hiding frequent itemsets, while this work addresses hiding association rules.

Besides, the authors in [11] recently proposed a reconstruction-based algorithm for classification rules hiding. This work is also worthy of reference. Another dimension to classify existing algorithms is: hiding rules or hiding large (frequent) itemsets. Part of the existing work above chooses to hide association rules, while others choose to hide large itemsets. Relatively, hiding rules is more complicated than hiding itemsets.

The whole approach is divided into three phases:

The first phase is to use frequent itemset mining algorithm to generate all frequent itemsets with their supports and support counts from original database *D*. From FS, they get the set of association rules R. Then in the second phase, they perform sanitization algorithm over FS and get the sanitized frequent itemsets of FS'. In best case, the sanitization algorithm ensures from *FS'* get exactly the set of non-sensitive rules R-Rh, with no normal rules lost and no ghost rules generated.

The third phase is to generate released database *D'* from *FS'* by using inverse frequent set mining algorithm. In their framework, they plan to adopt an inverse frequent set mining algorithm based on FP-tree which comprises the following two steps:

1) The algorithm tries to "guess" a FP-tree that satisfies all the frequent itemsets and their support counts in *FS'*. They call such a FP-tree a compatible FP-tree meaning that from this FP-tree we can mine the same set of frequent itemsets with the same support counts as *FS*'.

2) Then they generate a corresponding database *D'* directly from the compatible FP-tree by outspreading all the paths of the tree.

3. ASSOCIATION RULES AND SANITIZATION

In this section, the notion of association rules is precisely defined and a formulation of the problem is given. It is then proven that the problem of finding an optimal sanitization of the source database is NP-Complete. This is done for a number of (progressively more realistic) notions of what it means to "sanitize". The proofs are based on reductions of the problem addressed in this paper to the 3-SAT problem.

3.1 The Problem

Let

$I = \{i_1, i_2, \ldots, i_n\}$ be a set of *n* binary attributes called *items*.

Let

$D = \{t_1, t_2, \ldots, t_m\}$ be a set of transactions called the *database*.

Each transaction in *D* has a unique transaction ID and contains a subset of the items in *I*. A *rule* is defined as an implication of the form

$$X \Rightarrow Y$$

where

$$X, Y \subseteq I \text{ and } X \cap Y = \emptyset.$$

The sets of items (for short *itemsets*) *X* and *Y* are called *antecedent* (left-hand-side or LHS) and *consequent* (right-hand-side or RHS) of the rule.

The *support* supp(*X*) of an itemset *X* is defined as the proportion of transactions in the data set which contain the itemset

Confidence can be interpreted as an estimate of the probability *P*(*Y* | *X*), the probability of finding the RHS of the rule in transactions under the condition that these transactions also contain the LHS.

The *lift* of a rule is defined as

$$\text{lift}(X \Rightarrow Y) = \frac{\text{supp}(X \cup Y)}{\text{supp}(Y) * \text{supp}(X)} \qquad (1)$$

or the ratio of the observed confidence to that expected by chance.

An association rule is an implication of the form $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$ and $X \cap Y = \Phi$. We say the rule $X \Rightarrow Y$ holds in the database *D* with *confidence c* if $|X \cup Y| / |X| \geq c$. It can also be said that the rule $X \Rightarrow Y$ has *support s* if $|X \cup Y| / |D| \geq s$. Note while the support is a measure of the frequency of a rule, the confidence is a measure of the strength of the relation between sets of items. The well-known association rule mining problem aims to find all significant association rules. A rule is significant if its support and confidence is no less than the user specified minimum support threshold (*MST*) and minimum confidence threshold (*MCT*). To find the significant rules, an association rule mining algorithm first finds all the frequent itemsets and then derives the association rules from them. On the contrary, the association rule hiding problem aims to prevent some of these rules, which is referred as "sensitive rules", from being mined.

Given a database *D* to be released with minimum threshold "*MST*"(Minimum Support Threshold), "*MCT*" (Minimum Confidence Threshold) and a set of association rules *R* mined from *D*,a set of sensitive rules *Rh* subset of *R* to be hidden, a new database *D'* has to be found such that the rules in *Rh* can still be mined from *D'* ,however with predefined modification and the rules in *R-Rh* can be mined as earlier.

3.1.1 3-SAT

Satisfiability is the problem of determining if the variables of a given Boolean formula can be assigned in such a way as to make the formula evaluate to TRUE. Equally important is to determine whether no such assignments exist, which would imply that the function expressed by the formula is identically FALSE for all possible variable assignments. In this latter case, we would say that the function is unsatisfiable; otherwise it is satisfiable. To emphasize the binary nature of this problem, it is frequently referred to as *Boolean* or *propositional satisfiability*. The shorthand "**SAT**" is also commonly used to denote it, with the implicit understanding that the function and its variables are all binary-valued. n complexity theory, the satisfiability problem (SAT) is a decision problem, whose instance is a Boolean expression written using only AND, OR, NOT, variables, and parentheses. The question is: given the expression, is there some assignment of *TRUE* and *FALSE* values to the variables that will make the entire expression true? A formula of propositional logic is said to be *satisfiable* if logical values can be assigned to its variables in a way that makes the formula true. The Boolean satisfiability problem is NP-complete.

The propositional satisfiability problem (PSAT), which decides whether a given propositional formula is satisfiable, is of central importance in various areas of computer science, including theoretical computer science, algorithmics, artificial intelligence, hardware design, electronic design automation, and verification A literal is either a variable or the negation of a variable (the negation of an expression can be reduced to negated variables by De Morgan's laws). For example, x1 is a positive literal and NOT(x1) is a negative literal.

A clause is a disjunction of literals. For example, x1 v NOT(x2) is a clause (read as "x-sub-one or not x-sub-2"). 3-satisfiability is a special case of *k*-satisfiability (*k*-SAT) or simply satisfiability (SAT), when each clause contains exactly $k = 3$ literals. It was one of Karp's 21 NP-complete problems.

3-SAT :-Let S be the set of literals for 3-SAT problem. Let C be the subsets of finite set S of size 3 and containing literal of clauses, find a smallest set S' such that every subset in C contains atleast one element in S' that has to be setted true to make the expression satisfiable.

3.1.2 Reduction from 3-SAT

Let A be the set of large itemsets that are "good" in the sense that we do not wish to make them small. Let B be the set of large itemsets that are "bad", i.e., we want to make them small. These two goals can be incompatible, so the problems we

formulate below are based on the notion that we want to make all of B's itemsets small, while making as few as possible of A's itemsets small. We prove the NP-hardness of these optimization problem based on this notion So Reduction problem can be formulated as

" Given two sets A and B of subsets of a finite set J, such that no element of B is a subset of any element of A and no element of A is a subset of any element of B, find a set of elements R in J such that every subset in B contains at least one of those elements while minimizing the number of subsets of A that contain elements from R".

Let J be the set of all itemsets and A and B be the set of itemsets that are non-sensitive and sensitive respectively. So, we need to hide the itemsets belong to set B and minimize it in A.

Example 1:-

J = {1,2,3,4,5,6,7,8,9,10}

A = { {1,2} , {2,5 } , {6,7 } }

B = { {1,9} , {2,8} , {6,8 } }

So, R = { 8,9 }

Reduction

Given an instance of 3-SAT, here is how to create an instance of optimization problem such that polynomial time solution to latter implies a polynomial time solution to the former.

Let S = {1,2,3,4,….n} for 3-SAT problem Then for optimization problem here is what A,B,J look like in terms of C and S of 3-SAT problem instance.

J = S union { n+1,n+2 } i.e J = { 1,2,3…..n,n+1,n+2}

A = { {1,n+1,n+2} , {2,n+1,n+2} ,…….{n,n+1,n+2}}

B = C (hence n+1, n+2 does not appear anywhere in B).

The R that solves the instance of optimization problem is equal to the S' that solves the instance of 3-SAT.

Proof by example:-

Let S = { 1,2,3, ……10 }

J = { 1,2,3,…10,11,12}

A = { {1,9,10} , {2,9,10} , {3,9,10} ……{8,9,10}}

B can be { {1,2,5} , {3,6,8} , {4,5,7} } = C

R = S' = { 5,6}

Hence optimal sanitization is NP-Complete.

4. CONCLUDING REMARKS

The work reported in this paper deals with the time complexity and space complexity of the 3-SAT and optimal sanitization in association rule hiding. In this paper, Optimal sanitization problem is reduced to 3-SAT. We conclude that Optimal sanitization problem is NP-Complete. In this paper we introduce a new polynomial reduction from 3SAT to optimal sanitization in association rule hiding and demonstrate that this framework has advantages over the standard representation. More specifically, after presenting the reduction we conclude that many hard 3SAT instances can be solved using optimal sanitization.

5. REFERENCES

[1]. Kshitij Pathak, Aruna Tiwari, Narendra S Chaudhari A Reduction of 3-SAT problem from optimal sanitization in association rule hiding In: Proc. Of IEEE Int'l conference ETNCC, 2011, 43-46

[2]. Atallah, M., Bertino, E., Elmagarmid, A., Ibrahim, M., and Verykios, V.S. Disclosure limitation of sensitive rules. In: Scheuermann P, ed. Proc. of the IEEE Knowledge and Data Exchange Workshop (KDEX'99). IEEE Computer society, 1999. 45-52.

[3]. Verykios, V.S., Elmagarmid, A., Bertino, E., Saygin, Y., and Dasseni, E. Association rule hiding. IEEE Transactions on Knowledge and Data Engineering, 2004, 16(4):434-447.

[4]. Oliveira, S.R.M. and Zaïane, O.R. Privacy preserving frequent itemset mining. In: Proc. of the 2 nd IEEE ICDM Workshop on Privacy, Security and Data Mining. Australian Computer Society, 2002. 43-54.

[5]. Oliveira, S.R.M. and Zaïane, O.R. Protecting sensitive knowledge by data sanitization. In: Proc. of the 3Prd P IEEE Int'l Conf. on Data Mining (ICDM'03). IEEE Computer Society, USA, 2003. 613-616.

[6]. Oliveira, S.R.M. and Zaïane, O.R. A unified framework for protecting sensitive association rules in business collaboration. Int'l Journal of Business Intelligence and Data Mining, 2006, 1(3):247-287.

[7]. Shariq J. Rizvi Jayant R. Haritsa Maintaining Data Privacy in Association Rule Mining Proceedings of the 28th VLDB Conference,Hong Kong, China, 2002

[8]. Sun, X. and Yu, P.S. A border-based approach for hiding sensitive frequent itemsets. In: Proc. of the 5th IEEE Int'l

Conf. on Data Mining (ICDM'05). IEEE Computer Society, 2005. 426-433.

[9]. Saygin, Y., Verykios, V.S., and Clifton, C. Using unknowns to prevent discovery of association rules. SIGMOD Record,2001, 30(4):45-54.

[10]. Chen, X., Orlowska, M., and Li, X. A new framework for privacy preserving data sharing. In: Proc. of the 4th IEEE ICDM Workshop: Privacy and Security Aspects of Data Mining. IEEE Computer Society, 2004. 47-56.

[11]. Natwichai, J., Li, X., and Orlowska, M. A reconstruction based algorithm for classification rules hiding. In: Dobbie G, Bailey J, eds. Proc. of the Seventeenth Australasian Database Conf. (ADC'06). Australian Computer Society,2006. 49-58.