Available online at www.sciencedirect.com

ScienceDirect

AASRI Procedia 5 (2013) 228 – 234

AASRI Procedia

www.elsevier.com/locate/procedia

2013 AASRI Conference on Parallel and Distributed Computing and Systems

Quality Measurements for Association Rules Hiding

Hui Wang*

Department of information technology, National defense information academy, 38 Liberation Park Road ,Wuhan, 430010, China

Abstract

Data mining technologies are widely used and help people to make good decision and good money since appeared. While at the same time, the side effects incurred are concerned by data holders and researchers. Some information extracted from mining can be considered sensitive and needed to be hidden for safe data sharing. A lot of approaches of hiding have been proposed. There is still a lot of room to improve the efficiency and quality. The work focuses on the quality measurements of hiding approaches to guide the sensitive association rule hiding process. Several crucial quality measurements are defined. The requirements of sensitive association rule hiding are described. The transformation strategies of database are proposed. Examples are illustrated and quality measurements are compared.

Keywords: data mining ; privacy preserving ; sensitive rule hiding ; hiding requirement description ; quality measurements

1. Introduction

Data mining technologies are widely used and help people to make good decision and good money since appeared. While at the same time, some knowledge extracted from the data mining can be considered sensitive which may potentially reveal sensitive secrets and cause serious privacy problems [1-3]. This problem is dealt with privacy preserving data mining.

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000 .
E-mail address: author@institute.xxx .

doi:10.1016/j.aasri.2013.10.083

Privacy preserving data mining is a novel research direction in data mining. It analyzes the side effects incurring in data privacy. Privacy preserving data mining hides sensitive knowledge by modifying the original database in a way that sensitive knowledge disappeared with less affecting on non-sensitive knowledge.

A lot of approaches of hiding have been proposed. Usually, based on the discovered association rules and hiding requirements, the hiding of sensitive association rules is executed manually or automatically after data mining process. There is still a lot of room to improve the efficiency and quality. The work focuses on the quality measurements of hiding approaches to guide the sensitive association rule hiding process. Several key quality measurements will be proposed in this paper. The reminder of this paper is organized as follows. Section 2 describes the problem of association rule hiding. Section 3 presents the quality measurements. Association rule Hiding strategies are illustrated in section 4. Conclusions and future works are described in the last section.

2. Problem description

Association rule hiding is a classic knowledge hiding approach which deals with modifying the original database in a way that sensitive association rules disappear while at the same time, without seriously affecting the original data and the non-sensitive rules.

Given a set of association rules which are mined from a specific dataset and are considered to be sensitive, the task of association rule hiding is to properly sanitize the original data so that any association rule mining algorithms that may be applied to the sanitized version of the data will be incapable of discovering the sensitive rules under certain parameter settings, and while at the same time, will be able to mine all the non-sensitive rules that appeared in the original dataset under the same or higher parameter settings and no other rules[4-5]. It is quite challenging.

2.1. Hiding sensitive frequent itemsets

Assume that the original database is denoted as *D-original*, the pre-defined minimum support threshold as α, the set of all the frequent itemsets as *F-original*.

Let *F-sensitive*=$\{s \mid s \in$ *F-original* and *s* is sensitive$\}$.

The goal of sensitive frequent itemsets hiding is to construct a new, sanitized database *D-sanitized* from *D-original*, which makes *F-sensitive*$\cap$*F-sanitized*=Φ, where *F-sanitized* is the set of all the frequent itemsets on *D-sanitized* at the same pre-defined minimum support threshold as α (or higher) , while less affecting the non-sensitive itemsets , i.e., *F-sanitized=F-original-F-sensitive*.

To hide a sensitive itemset, the privacy preserving algorithm has to modify the original database *D-original* in such a way that when the sanitized database *D-sanitized* is mined at the same (or a higher) level of support, the frequent itemsets that are discovered are all non-sensitive.

2.2. Hiding sensitive association rules

Assume that the original database is denoted as *D-original*, the pre-defined minimum support and confidence threshold as α and β, the set of all the association rule that satisfy α and β as *R-origina*l .

Let *R-sensitive*=$\{r \mid r \in$ *R-original* and *r* is sensitive$\}$.

The goal of association rule hiding is to construct a new, sanitized database *D-sanitized* from *D-original*, which makes *R-sensitive*$\cap$*R-sanitized*=Φ, where R-*sanitized* is the set of all the association rules on *D-sanitized* at the same pre-defined minimum support threshold and minimum confidence threshold as α and β (or higher) , while less affecting the non-sensitive association rules , i.e., *F-sanitized=R-original-R-sensitive*.

2.3. Hiding requirement description

Given P: $P \neq \varnothing \wedge P \subseteq I$

The hiding requirement is described with $R_{sensitive}$.

$R_{sensitive} = \{r \mid \exists i - left \in P \wedge i - left \text{ is considered sensitive and contained in } r\text{'s left hand}\}$

The goal is to hide rules in *R-sensitive* by low the confidence down to below the pre-defined minimum confidence threshold β. According to the definition of confidence, it can be done by increasing the support of the left hand or decreasing the support of the right hand to decrease the confidence of the rule.

3. Quality measurements for hiding

The quality of hiding is important. Quality measurements measure the hidden effects, side effects and database effects. Let $C(r)$ and $C'(r)$ the confidence of association rules r in *R-original* and *R-sanitized*, respectively. Define hidden-rate, lost-rate, false-rate and altered-rate as following.

3.1. Hidden rate

Hidden rate is to measure the quality of sensitive association rule hiding. It measures the percentage of hidden association rules among all of the sensitive rules.

$$hidden - rate = \frac{|\{r \mid r \in R_{sensitive} \wedge C(r) \geq \beta \wedge C'(r) < \beta\}|}{|\{r \mid r \in R_{sensitive} \wedge C(r) \geq \beta\}|} * 100\%$$

The lower hidden-rate is the better. The best situation is hidden rate is 100%, that means that all the sensitive rules can be hidden at the same (or higher) thresholds α and β.

3.2. Lost rate

Lost rate is to measure the side effect of hiding. It measures the percentage of lost association rules among all non-sensitive rules.

$$lost - rate = \frac{|\{r \mid r \notin R_{sensitive} \wedge C(r) \geq \beta \wedge C'(r) < \beta\}|}{|\{r \mid r \notin R_{sensitive} \wedge C(r) \geq \beta\}|} * 100\%$$

The lower lost rate is the better. There should be no lost rules in the sanitized database which are arrived support and confidence in the original database at the same (or higher) thresholds α and β.

3.3. False rate

False rate is to measure the side effect, either. It measures the percentage of false association rules among all of rules whose confidence is below the pre-defined minimum confidence threshold.

$$false - rate = \frac{|\{r \mid C(r) < \beta \wedge C'(r) \geq \beta\}|}{|\{r \mid C(r) < \beta\}|} * 100\%$$

The lower false rate is the better. No false rules should be produced when mining the sanitized database at the same (or higher) thresholds α and β.

3.4. Altered rate

Altered rate is to measure the workload of the database transformation. It measures the percentage of altered transactions among the entire database.

$$altered-rate = \frac{|\{r \mid r \in D-original \wedge r\ is\ altered\ in\ D-sanitized\}|}{|D-original|} * 100\%$$

The lower altered rate is the better. When more transactions are modified, the more processing time would be required.

4. Transformation strategies

Assuming some predicting items contained in the left hand of an association rule and it is so crucial to infer a conclusion which is the right hand of the association rule. There are might multiple items contained in the right hand of association rule. It is only need to hide rules with one item in the right hand instead of multi-items' ones. We have theorem 1 as follow.

4.1. Theorem 1

Given

$$A \cup \{i-left\} \rightarrow \{i-right\} \in R\ ;\ A \cup \{i-left\} \rightarrow B \cup \{i-right\} \in R\ ;$$
$$A \neq \varnothing\ and\ B \neq \varnothing\ and\ A \cap B = \varnothing\ ;\ i-left \in I\ and\ i-right \in I\ ;\ i-left \neq i-right$$

Following conclusion holds:
confidence(A ∪{i-left} →B ∪{i-right})<=confidence(A ∪{i-left} →{i-right})
Proof:
According to the definition of rule's confidence:

$$confidence(A \rightarrow B) = \frac{support(A \cup B)}{support(A)}$$

$$confidence(A \cup \{i-left\} \rightarrow \{i-right\}) = \frac{support(A \cup \{i-left\} \cup \{i-right\})}{support(A)}$$

$$confidence(A \cup \{i-left\} \rightarrow B \cup \{i-right\}) = \frac{support(A \cup \{i-left\} \cup B \cup \{i-right\})}{support(A)}$$

According to the definition of rule's support:

$$support(A) = |\{(tid,T) \mid (tid,T) \in D \wedge T \subseteq A \wedge A \subseteq I\}|$$

support(A ∪{i-left} ∪{i-right})>= support(A ∪{i-left} ∪B ∪{i-right})

So, the conclusion above holds.

4.2. Strategies

The strategies are based on theorem 1. According to theorem 1, if association rule such as $A \cup \{i\text{-}left\} \rightarrow \{i\text{-}right\}$ is hidden, then all the association rules such as $A \cup \{i\text{-}left\} \rightarrow B \cup \{i\text{-}right\}$ ($B \neq \Phi$) are hidden. To hide rules such as $A \cup \{i\text{-}left\} \rightarrow \{i\text{-}right\}$, it is needed to check the transactions contained $\{i\text{-}left, i\text{-}right\}$ and increase the appearances of item *i-left* in the transactions to increase the support of the left hand or decrease the appearances of item *i-right* in the transactions to decrease the support of the right hand. The transformation of database is repeated until good quality measurements are arrived.

4.3. Examples

Assuming $F=\{a,b,c\}$; $P=\{c\}$; $\alpha=2$; $\beta=0.75$.

The goal is to hide rules containing item *c* in the left hand. We consider the hidden effect and side effects at a transformation of one transaction of the database.

Let database is represented as a binary vector where 1 means an item contained in a transaction and 0 means the item not contained in the transaction.

(1) Decrease the support of the right hand

Let *D-original*={011,111,111,110,100,001}

We decrease the appearance of item *b* to hide rules with the right hand contained item *b* while the left hand contained item *c*.

For example, we make item *b* disappear in transaction 1 in *D-sanitized*.

D-sanitized={001,111,111,110,100,001}

The results in this step are illustrated in Table 1, where *hidden-rate*=50%, l*ost-rate*=25%, *false-rate*=1/7.

Table 1. Hiding by decreasing the support of item *b*

Rule set	Rule	C(D-original)	C'(D-sanitized)	Status
To hide	*c→a*	2/4	2/4	hold
	c→b	3/4	2/4	hidden
	c→ab	2/4	2/4	hold
	ac→b	2/2	2/2	fail
	bc→a	2/3	2/2	false
Maybe false	*a→b*	3/4	3/4	hold
	a→c	2/4	2/4	hold
	a→bc	2/4	2/4	hold
Maybe lost	*ab→c*	2/3	2/3	hold
	b→a	3/4	3/3	hold
	b→c	3/4	2/3	lost
	b→ac	2/4	2/3	hold

(2) Increase the support of the left hand

Let *D-original*={111,111,111,110,100,101}

We increase the appearance of item *c* to hide rules with the right hand contained item *a* while the left hand contained item *c*.

For example, we make item *c* appear in transaction 5 in *D-sanitized*.

*D-sanitize*d={111,111,111,110,101,101}

The results in this step are illustrated in Table 2, where *hidden-rate*=60%, l*ost-rate*=0, *false-rate*=1/3.

Table 2. Hiding by increasing the support of item *c*

Rule set	Rule	C(D-original)	C'(D-sanitized)	Status
To hide	*c→a*	4/4	5/5	fail
	c→b	3/4	3/5	hidden
	c→ab	3/4	3/5	hidden
	ac→b	3/4	3/5	hidden
	bc→a	3/3	3/3	fail
Maybe false	*a→b*	4/6	4/6	hold
	a→c	4/6	5/6	false
	a→bc	3/6	3/6	hold
Maybe lost	*ab→c*	3/4	3/4	hold
	b→a	4/4	4/4	hold
	b→c	3/4	3/4	hold
	b→ac	3/4	3/4	hold

5. Conclusions

Currently, approaches for privacy preserving of association rules are basically based on the same fundamental theory of adjusting supports and confidences. The work focuses on the quality measurements of hiding approaches to guide the sensitive association rule hiding process. Several crucial quality measurements are defined. The requirements of sensitive association rule hiding are described. The transformation strategies of database are proposed. Examples are illustrated and quality measurements are compared. Lots of algorithms are proposed by researchers and a lot of room for improvement of the current hiding methodologies. It is quite challenging to arrive at each of the best quality measurement.

References

[1]VS Verykios and A Gkoulalas-Divanis, "A survey of association rule hiding methods for privacy", in Privacy Preserving Data Mining: Models and Algorithms, Springer Berlin Heidelberg, 2008, pp. 267-289.
[2]A Gkoulalas-Divanis and VS Verykios, "Privacy preserving data mining: how far can we go?", in Handbook of Research on Data Mining in Public and Private Sectors: Organizational and Governmental Applications, IGI Global, 2009, pp. 1-21.
[3]A Agrawal, U Thakar, R Soni, and BK Chaurasia, "Efficiency enhanced association rule mining technique", in D Nagamalai, E Renault, and M Dhanushkodi (eds.) PDCTA 2011, CCIS, vol. 203, 2011, pp.

375-384.
[4]A Gkoulalas-Divanis and VS Verykios, “Association Rule Hiding for Data Mining”, Advances in Database Systems 41, Springer Science Business Media, LLC 2010.
[5]Shyue-Liang Wang , Bhavesh Parikh, Ayat Jafari, “Hiding informative association rule sets”, Expert Systems with Applications 33 (2007) , pp.316-323.