A New Scheme on Privacy Preserving Association Rule Mining*

Nan Zhang, Shengquan Wang, and Wei Zhao

Department of Computer Science, Texas A&M University
College Station, TX 77843, USA
{nzhang,swang,zhao}@cs.tamu.edu

Abstract. We address the privacy preserving association rule mining problem in a system with one data miner and multiple data providers, each holds one transaction. The literature has tacitly assumed that randomization is the only effective approach to preserve privacy in such circumstances. We challenge this assumption by introducing an algebraic techniques based scheme. Compared to previous approaches, our new scheme can identify association rules more accurately but disclose less private information. Furthermore, our new scheme can be readily integrated as a middleware with existing systems.

1 Introduction

In this paper, we address issues related to production of accurate data mining results, while preserving the private information in the data being mined. We will focus on association rule mining. Since Agrawal, Imielinski, and Swami addressed this problem in [1], association rule mining has been an active research area due to its wide applications and the challenges it presents. Many algorithms have been proposed and analyzed [2–4]. However, few of them have addressed the issue of privacy protection.

Borrowing terms from e-business, we can classify privacy preserving association rule mining systems into two classes: business to business (B2B) and business to customer (B2C), respectively. In the first category (B2B), transactions are distributed across several sites (businesses) [5, 6]. Each of them holds a private database that contains numerous transactions. The sites collaborate with each other to identify association rules spanning multiple databases. Since usually only a few sites are involved in a system (e.g., less than 10), the problem here can be modelled as a variation of secured multiparty computation [7]. In the second category (B2C), a system consists of one data miner (business) and multiple data providers (customers) [8, 9]. Each data provider holds only one transaction. Association rule mining is performed by the data miner on the aggregated transactions provided by data providers. On-line survey is a typical example of this type of system, as the system can be modelled as one data miner (i.e., the

* This work was supported in part by the National Science Foundation under Contracts 0081761, 0324988, 0329181, by the Defense Advanced Research Projects Agency under Contract F30602-99-1-0531, and by Texas A&M University under its Telecommunication and Information Task Force Program. Any opinions, findings, conclusions, and/or recommendations expressed in this material, either expressed or implied, are those of the authors and do not necessarily reflect the views of the sponsors listed above.

J.-F. Boulicaut et al. (Eds.): PKDD 2004, LNAI 3202, pp. 484–495, 2004.

survey collector and analyzer) and millions of data providers (i.e., the survey providers). Privacy is of particular concern in this type of system; in fact there has been wide media coverage of the public debate of protecting privacy in on-line surveys [10]. Both B2B and B2C have wide applications. Nevertheless, in this paper, we will focus on studying B2C systems.

Several studies have been carried out on privacy preserving association rule mining in B2C systems. Most of them have tacitly assumed that randomization is an effective approach to preserving privacy. We challenge this assumption by introducing a new scheme that integrates algebraic techniques with random noise perturbation. Our new method has the following important features that distinguish it from previous approaches:

- Our system can identify association rules more accurately but disclosing less private information. Our simulation data show that at the same accuracy level, our system discloses private transaction information about five times less than previous approaches.
- Our solution is easy to implement and flexible. Our privacy preserving mechanism does not need a support recovery component, and thus is transparent to the data mining process. It can be readily integrated as a middleware with existing systems.
- We allow explicit negotiation between data providers and the data miner in terms of tradeoff between accuracy and privacy. Instead of obeying the rules set by the data miner, a data provider may choose its own level of privacy. This feature should help the data miner to collaborate with both hard-core privacy protectionists and persons comfortable with a small probability of privacy divulgence.

The rest of this paper is organized as follows: In Sect. 2, we present our models, review previous approaches, and introduce our new scheme. The communication protocol of our system and related components are discussed in Sect. 3. A performance evaluation of our system is provided in Sect. 4. Implementation and overhead are discussed in Sect. 5, followed by a final remark in Sect. 6.

2 Approaches

In this section, we will first introduce our models of data, transactions, and data miners. Based on these models, we review the randomization approach – a method that has been widely used in privacy preserving data mining. We will point out the problems associated with the randomization approach which motivates us to design a new privacy preserving method, based on algebraic techniques.

2.1 Model of Data and Transactions

Let I be a set of n items: $I = \{a_1, \ldots, a_n\}$. Assume that the dataset consists of m transactions $t_1, \ldots, t_m$, where each transaction t_i is represented by a subset of I. Thus, we may represent the dataset by an $m \times n$ matrix $T = [a_1, \ldots, a_n] = [t_1, \ldots, t_m]'$ [1]. Let $\langle T \rangle_{ij}$ denote the element of T with indices i and j. Correspondingly, for a vector v,

[1] We denote the transpose of matrix T as T'.

Table 1. Transaction Matrix

$$\begin{array}{c|cccc} & a_1 & a_2 & \cdots & a_n \\ \hline t_1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_m & 1 & 0 & \cdots & 1 \end{array}$$

its ith element is represented by $\langle v \rangle_i$. An example of matrix T is shown in Table 1. The elements of the matrix depict whether an item appears in a transaction. For example, suppose the first transaction contains items a_{20} and a_{47}. Then the first row of the matrix has $\langle T \rangle_{1,20} = \langle T \rangle_{1,47} = 1$ and all other elements equal to 0.

An itemset $B \subseteq I$ is k-itemset if B contains k items (i.e., $|B| = k$). The *support* of B is defined as

$$supp(B) = \frac{|\{t \in T | B \subseteq t\}|}{m} \tag{1}$$

A k-itemset B is frequent if $supp(B) \geq min\_supp_k$, where min_supp_k is a predefined minimum threshold of support. The set of frequent k-itemsets is denoted by L_k. Technically speaking, the main task of association rule mining is to identify frequent itemsets.

2.2 Model of Data Miners

There are two classes of data miners in our system. One is *legal data miners*. These miners always act legally in that they perform regular data mining tasks and would never intentionally breach the privacy of the data. On the other hand, *illegal data miners* would purposely discover the privacy in the data being mined. Illegal data miners come in many forms. In this paper, we focus on a particular sub-class of illegal miners. That is, in our system, illegal data miners are *honest but curious*: they follow proper protocol (i.e., they are honest), but they may keep track of all intermediate communications and received transactions to perform some analysis (i.e., they are *curious*) to discover private information [11].

Even though it is a relaxation from Byzantine behavior, this kind of honest but curious (nevertheless illegal) behavior is most common and has been widely adopted as an adversary model in the literatures. This is because, in reality, a workable system must benefit both the data miner and the data providers. For example, an online bookstore (the data miner) may use the association rules of purchase records to make recommendations to its customers (data providers). The data miner, as a long-term agent, requires large numbers of data providers to collaborate with. In other words, even an illegal data miner desires to build a reputation for trustworthiness. Thus, honest but curious behavior is an appropriate choice for many illegal data miners.

2.3 Randomization Approach

To prevent the privacy breach due to the illegal data miners, countermeasures must be implemented in data mining systems. Randomization has been the most common approach for countermeasures. We briefly view this method below.

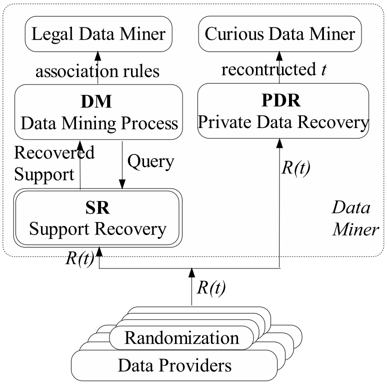

Fig. 1. Randomization approach

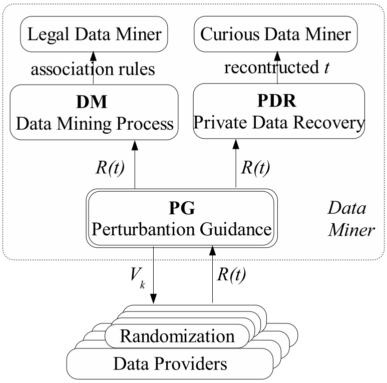

Fig. 2. Our new scheme

We consider the entire mining process to be an iterative one. In each stage, the data miner obtains a perturbed transaction from a different data provider. With the randomization approach, each data provider employs a randomization operator $R(\cdot)$ and applies it to one transaction t which the data provider holds. Fig. 1 depicts this kind of system.

Upon receiving transactions from the data providers, the legal data miner must first perform an operation called *support recovery* which intends to filter out the noise injected in the data due to randomization, and then carry out the data mining tasks. At the same time, an illegal (honest but curious) data miner may perform a particular privacy recovery algorithm in order to discover private data from that supplied by the data providers.

Clearly, the system should be measured by its capability in terms of supporting the legal miner to discover accurate association rules, while preventing illegal miner from discovering private data.

2.4 Problems of Randomization Approach

Researchers have discovered some problems with the randomization approach. For example, as pointed in [8], when the randomization is implemented by a so called *cut-and-paste* method, if a transaction contains 10 items or more, it is difficult, if not impossible, to provide effective information for association rule mining while at the same time preserving privacy. Furthermore, large itemsets have exceedingly high variances on recovered support values. Similar problems would exist with other randomization methods (e.g., MASK system [9]) as they all use random variables to distort the original transactions.

Now, we will explore the reasons behind these problems.

- First, we note that previous randomization approaches are *transaction-invariant*. In other words, the same perturbation algorithm is applied to all data providers. Thus, transactions of a large size (e.g., $|t| > 10$) are doomed to failure in privacy protection by the large numbers of the real items divulged to the data miner. The solution proposed in [8] has ignored all transactions with a size larger than 10. However, a real dataset may have about 5% such transactions. Even if the average transaction size is relatively small, this solution still prevents many frequent itemsets (e.g., with size of 4 or more) from being discovered.

- Second, previous approaches are *item-invariant*. All items in the original transaction t have the same probability of being included in the perturbed transaction $R(t)$. No specific operation is performed to preserve the correlation between different items. Thus, a lot of real items in the perturbed transactions may never appear in any frequent itemset. In other words, the divulgence of these items does not contribute to the mining of association rules.

Note that invariance of transactions and items is inherent in the randomization approach. This is because in this kind of system, the communication is one-way: from data providers to the data miner. As such, a data provider cannot obtain any specific guidance on the perturbation of its transaction from the (legal) data miner. Consequently, lack of communication between data providers prevents a data provider from learning the correlation between different items. Thus, a data provider has no choice but to employ a *transaction-invariant* and *item-invariant* mechanism.

This observation motivates us to develop a new approach that allows two-way communication between the data miner and data provider. We describe the new approach in the next subsection.

2.5 Our New Approach

Fig. 2 shows the infrastructure of our system. The (legal) data miner S contains two components: DM (data mining process) and PG (perturbation guidance). When a data provider C_i initializes a communication session, PG first dispatches a reference V_k to C_i. Based on the received V_k, the data perturbation component of C_i transforms the transaction t to a perturbed one $R(t)$ and transmits $R(t)$ to PG. PG then updates V_k based on the recently received $R(t)$ and forwards $R(t)$ to the data mining process DM.

The key here is to properly design V_k so that correct guidance can be provided to the data providers on how to distort the data transactions. In our system, we let V_k be an algebraic quantity derived from T. As we will see, with this kind of V_k, our system can effectively maintain accuracy of data mining while significantly reduce the leakage of private information.

3 Communication Protocol and Related Components

In this section, we will present the communication protocol and the associated components in our system. Recall that in our system, there is a two-way communication between data providers and the data miner. While only little overhead is involved, this two-way communication substantially improves performance of privacy preserving discovered association rules.

3.1 The Communication Protocol

We now describe the communication protocol used between the data providers and data miners. On the side of the data miner, there are two current threads that perform the following operations iteratively after initializing V_k:

Thread of registering data provider:

```
R1. Negotiate on the truncation level $k$ with a data provider;
R2. Wait for a ready message from a data provider;
R3. Upon receiving the ready message from a data provider,
      – Register the data provider;
      – Send the data provider current $V_k$;
R4. Go to Step R1;
```

Thread of receiving data transaction:

```
T1. Wait for a (perturbed) data transaction $R(t)$ from a data provider;
T2. Upon receiving the data transaction from a registered data provider,
      – Update $V_k$ based on the newly received perturbed data transaction;
      – Deregister the data provider;
T3. Go to Step T1;
```

For a data provider, it performs the following operations to transfer its transaction to the data miner:

```
P1. Send the data miner a ready message indicating that this provider is ready to contribute to the mining process.
P2. Wait for a message that contains $V_k$ from the data miner.
P3. Upon receiving the message from the data miner, compute $R(t)$ based on $t$ and $V_k$.
P4. Transfer $R(t)$ to the data miner.
```

3.2 Related Components

It is clear from the above description that the key components of our communication protocol are (a) the method of computing V_k; and (b) the algorithm for perturbation function $R(\cdot)$. We discuss these components in the following. Negotiation is also critical. The details of negotiation protocol can be found in [12].

Computation of V_k. Recall that V_k carries information from the data miner to data providers on how to distort a data transaction in order to preserve privacy. In our system, V_k is an estimation of the eigenvectors of $A = T'T$. Due to space limit, we refer users to [12] about the justification of V_k on providing accurate mining results.

As we are considering dynamic case where data transactions are dynamically fed to the data miner, the miner keeps a copy of all received transactions and need to update it when a new transaction is received. Assume that the initial set of received transactions T^* is empty[2] and every time when a new (distorted) data transaction, $R(t)$, is received, T^* is updated by appending $R(t)$ at the bottom of T^*. Thus, T^* is the matrix of perturbed transactions. We derive V_k from T^*.

In particular, the computation of V_k is done in the following steps. Using singular value decomposition (SVD) [13], we can decompose $A^* = T^{*\prime}T^*$ as (2) where diagonal matrix $\Sigma^* = \mathrm{diag}(s_1^2, \ldots, s_n^2)$ and $s_1^2 \geq \ldots \geq s_n^2$.

$$A^* = T^{*\prime}T^* = V^*\Sigma^*V^{*\prime} \tag{2}$$

V^* is an $n \times n$ unitary matrix composed of the eigenvectors of A^*.

[2] T^* may also be composed of some transactions provided by privacy-careless data providers.

V_k is composed of the first k vectors of V^* (i.e., eigenvectors corresponding to the largest k eigenvalues of A^*). In other words, if $V^* = [v_1, \ldots, v_n]$, then

$$V_k = [v_1, \ldots, v_k] \tag{3}$$

Thus, we call V_k as the k-truncation of V^*. Several incremental algorithms have been proposed to update V_k when a new (distorted) data transaction is received by the data miner [14, 15]. The computing cost of updating V_k is addressed in Sect. 5.

Note that k is a given integer less than or equal to n. As we will see in Sect. 4, k can play a critical role in balancing accuracy and privacy. We will also show that by using V_k in conjunction with $R(\cdot)$, to be discussed next, we can achieve desired accuracy and privacy.

Perturbation Function $R(\cdot)$. Recall that once a data provider receives a perturbation guidance V_k from the data miner, the provider applies a perturbation function, $R(\cdot)$, to its data transaction, t. The result is a distorted transaction that will be transmitted to the data miner. The computation of $R(t)$ is defined as follows. First, for the given V_k, the data transaction, t, is transformed by $\tilde{t} = tV_kV_k'$. Note that the elements in $\tilde{t}$ may not be integers. Algorithm Mapping is employed to integerize $\tilde{t}$. In the algorithm, ρ_t is a pre-defined parameter. Finally, to enhance the privacy preserving capability, we need to insert additional noise into $R(t)$. This is done by Algorithm Random-Noise Perturbation.

```
Algorithm Mapping
for every element $\langle\tilde{t}\rangle_i$ in $\tilde{t}$ do
    if $\langle\tilde{t}\rangle_i \geq 1 - \rho_\mathrm{t}$ then
        $\langle R(t)\rangle_i = 1$
    else
        $\langle R(t)\rangle_i = 0$
    end if
end for
```

```
Algorithm Random-Noise Perturbation
for every item $a_i \notin t$ do
    Choose a real number $j$ uniformly at random on $[0, 1]$
    if $j \geq 1 - \rho_\mathrm{m}$ then
        $\langle R(t)\rangle_i = 1$
    end if
end for
```

Now, computation of R(t) has been completed and it is ready to be transmitted to the data miner.

We have described our system – the communication protocol and its key components. We now discuss the accuracy and privacy metrics of our system.

4 Analysis on Accuracy and Privacy

In this section, we will propose the metrics of accuracy and privacy with analysis of the tradeoff between them. We will derive a upper bound on the degree of accuracy in the mining results (frequent itemsets). An analytical formula for evaluating the privacy metric is also provided.

4.1 Accuracy Metric

We use the error of support of frequent itemsets to measure the degree of accuracy in our system. This is because general objective of association rule mining is to identify all frequent itemsets with support larger than a threshold min_supp. There are two kinds of errors: *false drops*, which are undiscovered frequent itemsets and *false positives*, which are itemsets wrongly identified to be frequent. Formally, given itemset I_j, let the support of I_j in the original transactions T and the perturbed transactions $R(T)$ be $supp(I_j)$ and $supp'(I_j)$, respectively. Recall that the set of frequent h-itemsets in T is L_h. With these notations, we can define those two errors as follows:

Definition 1. *For a given itemset size h, the error on false drops, ρ_1, and the error on false positives, ρ_2, are defined as*

$$\rho_1 = \max_{I_j \in L_h} (supp(I_j) - supp'(I_j)), \tag{4}$$

$$\rho_2 = \max_{I_j \notin L_h} (supp'(I_j) - supp(I_j)). \tag{5}$$

We define the degree of accuracy as the maximum of ρ_1 and ρ_2 on all itemset sizes.

Definition 2. *The degree of accuracy in a privacy preserving association rule mining system is defined as $\gamma = \max_{h \geq 1} \max(\rho_1, \rho_2)$.*

With this definition, we can derive an upper bound on the degree of accuracy.

Theorem 1. $\gamma \leq 2.618\sigma_{k+1}^2/m$*, where σ_i is the ith eigenvalue of $A = T'T$.*

The proof can be found in [12].

This bound is fairly small when m is sufficiently large, which is usually the case in reality. Actually, our method tends to enlarge the support of high-supported itemsets and reduce the support of low-supported itemsets. Thus, the effective error that may result in false positives or false drops is much smaller than the upper bound. We may see this from the simulation results later.

4.2 Privacy Metric

In our system, the data miner cannot deduce the original t from $\tilde{t} = tV_kV_k'$ because V_kV_k' is a singular matrix with $det(V_kV_k') = 0$ (i.e., it does not have an inverse matrix). Since $t \rightarrow \tilde{t} \rightarrow R(t)$, t cannot be deduced from $R(t)$ deterministically. To measure the probability that an item in t is identified from $R(t)$, we need a privacy metric.

A privacy metric, *privacy breach*, is proposed in [8]. It is defined by the posterior probability $\Pr\{a_i \in t|t'\}$ that an item could be recovered from the perturbed transaction. Unfortunately, this metric is unsuitable in our system settings, especially to Internet applications. Consider a person taking an online survey of the commodities he/she purchased in the last month. A privacy breach of 50% (which is achieved in [8]) does not prevent privacy divulgence effectively. For instance, for a company who uses spam mail to make advertisement, a 50% probability of success (correct identification of a person who purchased similar commodities in the last month) certainly deserves a try because a wrong estimation (a spam mail sent to a wrong person) costs little.

We propose a privacy metric that measures the number of "unwanted" items (i.e., items not contribute to association rule mining) divulged to the data miner. For an item a_i that does not appear in any frequent itemset (i.e., $a_i \notin \bigcup L_k$), the divulgence of a_i

(i.e., $a_i \in R(t)$) does not contribute to the mining of association rules. Due to survey results in [10], a person has a strong will to filter out such "unwanted" information (i.e., information not effective in data mining) before divulging private data in exchange of data mining results. We evaluate the level of privacy by the probability of an "unwanted" item to be included in the transformed transaction. Formally, the level of privacy is defined as follows:

Definition 3. *Given a transaction t, an item $a_i \in t$ appears in a frequent itemset in t if there exists a frequent itemset I_j such that $a_i \in I_j \subseteq t$. Otherwise we say that a_i is infrequent in t. We define the level of privacy as*

$$\delta = \Pr\{a_i \in R(t) | a_i \textit{ is infrequent in } t\} \tag{6}$$

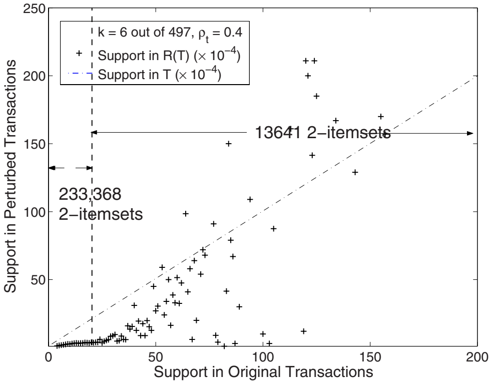

Fig. 3. Comparison of 2-itemsets supports between original and perturbed transactions

Fig. 3 shows a simulation result on all 2-itemsets. The x-axis is the support of itemsets in original transactions. The y-axis is the support of itemsets in perturbed transactions. The figure intends to show how effectively our system blocks the unwanted items from being divulged. If a system preserves privacy perfectly, we should have y equal to zero when x is less than min_supp_2. The data in Fig. 3 shows that almost all 2-itemsets with support less than 0.2% (i.e., $233,368$ unwanted 2-itemsets) have been blocked. Thus, the privacy has been successfully protected. Meanwhile, the supports of frequent 2-itemsets are exaggerated. This should help the data miner to identify frequent itemsets from additional noises.

Formally, we can derive an upper bound on the level of privacy.

Theorem 2. *The level of privacy in our system is bounded by*

$$\delta \leq 1 - \sqrt{\frac{\sigma_{k+1}^2 + \cdots + \sigma_n^2}{\sigma_1^2 + \cdots + \sigma_n^2}}. \tag{7}$$

where σ_i is the ith eigenvalue of $A = T'T$.

The proof can be found in [12].

By Theorems 1 and 2, we can observe a tradeoff between accuracy and privacy. Note that σ_i is sorted in descending order. Thus a larger k results in more "unwanted" items to be divulged. Simultaneously, the degree of accuracy (whose upper bound is in proportion to σ_{k+1}^2) decreases.

4.3 Simulation Results on Real Datasets

We will present the comparison between our approach and the cut-and-paste randomization operator by simulation results obtained on real datasets. We use a real world dataset BMS Webview 1 [16]. The dataset contains web click stream data of several months from the e-commerce website of a leg-care company. It has 59,602 transactions and 497 distinct items.

We randomly choose 10,871 transactions from the dataset as our test band. The maximum transaction size is 181. The average transaction size is 2.90. There are 325 transactions (2.74%) with size 10 or more. If we set $min\_supp = 0.2\%$, there are 798 frequent itemsets including 259 one-itemset, 350 two-itemsets, 150 three-itemsets, 37 four -itemsets and two 5-itemsets.

As a compromise between privacy and accuracy, the cutoff parameter K_m of cut-and-paste randomization operator is set to 7. The truncation level k of our approach is set to 6. Since both our approach and the cut-and-paste operator use the same method to add random noise, we compare the results before noise is added. Thus we set $\rho_{\mathrm{m}} = 0$ for both our approach and the cut-and-paste randomization operator.

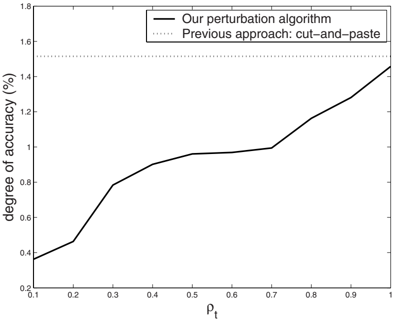

Fig. 4. Accuracy

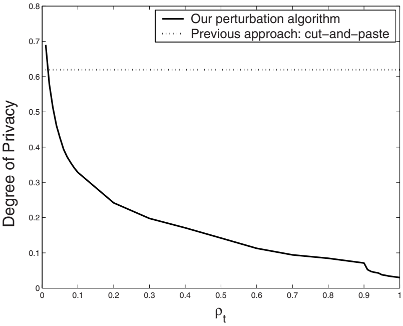

Fig. 5. Privacy

The solid line in Fig. 4 shows the change of degree of accuracy ($\max\{\rho_1, \rho_2\}$) of our approach with the parameter ρ_{t}. The dotted line shows the degree of accuracy while cut-and-paste randomization operator is employed. We can see that our approach reaches a better accuracy level than the cut-and-paste operator. A recommendation made from the figure is that $\rho_{\mathrm{t}} \in (0.7, 0.8)$ is suitable for hard-core privacy protectionists while $\rho_{\mathrm{t}} \in (0.2, 0.3)$ is recommended to persons care accuracy of association rules more than privacy protection.

The relationship between the level of privacy and ρ_{t} in the same settings is presented in Fig. 5. The dotted line shows the level of privacy of the cut-and-paste randomization operator. We can see that the privacy level of our approach is much higher than the cut-and-paste operator when $\rho_{\mathrm{t}} > 0.1$. Thus our approach is always better on both privacy and accuracy issues when $0.1 \leq t \leq 1$.

5 Implementation

A prototype of the privacy preserving association rule mining system with our new scheme has been implemented on web browsers and servers for online surveys. Visitors taking surveys are considered to be data providers. The data perturbation algorithm is implemented as custom codes on web browsers. The web server is considered to be the data miner. A custom code plug-in on the web server implements the PG (perturbation guidance) part of the data miner. All custom codes are component-based plug-ins that one can easily install to existing systems. The components required for building the system is shown in Fig. 6.

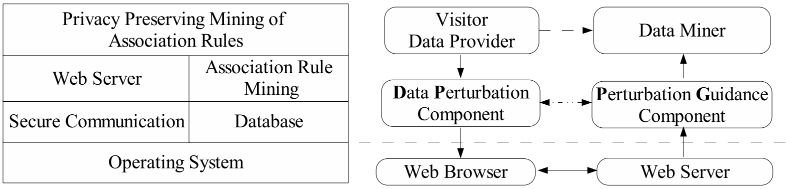

Fig. 6. System implementation

The overhead of our implementation is substantially smaller than previous approaches in the context of online survey. The time-consuming part of the "cut-and-paste" mechanism is on support recovery, which has to be done while mining association rules. The support recovery algorithm needs the partial support of all candidate items for each transaction size, which results in a significant overhead on the mining process.

In our system, the only overhead (possibly) incurred on the data miner is updating the perturbation guidance V_k, which is an approximation of the first k right eigenvectors of $A^* = T^{*\prime}T^*$. Many SVD updating algorithms have been proposed including SVD-updating, folding-in and recomputing the SVD [14, 15]. Since T^* is usually a sparse matrix, the complexity of updating SVD can be considerably reduced to $O(n)$. Besides, this overhead is not on the critical time path of the mining process. It occurs during data collection instead of data mining process. Note that the transfered "perturbation guidance" V_k is of the length kn. Since k is always a small number (e.g., $k \leq 10$), the communication overhead incurred by "two-way" communication is not significant.

6 Final Remarks

In this paper, we propose a new scheme on privacy preserving mining of association rules. In comparison with previous approaches, we introduce a two-way communication mechanism between the data miner and data providers with little overhead. In particular, we let the data miner send a perturbation guidance to the data providers. Using this intelligence, the data providers distort the data transactions to be transmitted to the miner. As a result, our scheme identifies association rules more precisely than previous approaches and at the same time reaches a higher level of privacy.

Our work is preliminary and many extensions can be made. For example, we are currently investigating how to apply a similar algebraic approach to privacy preserving classification and clustering problems. The method of SVD has been broadly adopted to many knowledge discovery areas including latent semantic indexing, information retrieval and noise reduction in digital signal processing. As we have shown, singular value decomposition can be an effective mean in dealing with privacy preserving data mining problems as well.

References

1. R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD Int. Conf. on Management of Data*, 1993, pp. 207–216.
2. R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Proc. Int. Conf. on Very Large Data Bases*, 1994, pp. 487–499.
3. J. S. Park, M.-S. Chen, and P. S. Yu, "An effective hash-based algorithm for mining association rules," in *Proc. ACM SIGMOD Int. Conf. on Management of Data*, 1995, pp. 175–186.
4. M. Fang, N. Shivakumar, H. Garcia-Molina, R. Motwani, and J. D. Ullman, "Computing Iceberg queries efficiently," in *Proc. Int. Conf. on Very Large Data Bases*, 1998, pp. 299–310.
5. J. Vaidya and C. Clifton, "Privacy preserving association rule mining in vertically partitioned data," in *Proc. ACM SIGKDD Int. Conf. on Knowledge discovery and data mining*, 2002, pp. 639–644.
6. M. Kantarcioglu and C. Clifton, "Privacy-preserving distributed mining of association rules on horizontally partitioned data," in *Proc. ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, 2002, pp. 24–31.
7. Y. Lindell and B. Pinkas, "Privacy preserving data mining," *Advances in Cryptology*, vol. 1880, pp. 36–54, 2000.
8. A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke, "Privacy preserving mining of association rules," in *Proc. ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, 2002, pp. 217–228.
9. S. J. Rizvi and J. R. Haritsa, "Maintaining data privacy in association rule mining," in *Proc. Int. Conf. on Very Large Data Bases*, 2002, pp. 682–693.
10. J. Hagel and M. Singer, *Net Worth*. Harvard Business School Press, 1999.
11. O. Goldreich, *Secure Multi-Party Computation*. Working Draft, 2002.
12. N. Zhang, S. Wang, and W. Zhao, "On a new scheme on privacy preserving association rule mining," Texas A&M University, Tech. Rep. TAMU/DCS/TR2004-7-1, 2004.
13. G. H. Golub and C. F. V. Loan, *Matrix Computations*. Baltimore, Maryland: Johns Hopkins University Press, 1996.
14. J. R. Bunch and C. P. Nielsen, "Updating the singular value decomposition," *Numerische Mathematik*, vol. 31, pp. 111–129, 1978.
15. M. Gu and S. C. Eisenstat, "A stable and fast algorithm for updating the singular value decomposition," Yale University, Tech. Rep. YALEU/DCS/RR-966, 1993.
16. Z. Zheng, R. Kohavi, and L. Mason, "Real world performance of association rule algorithms," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2001, pp. 401–406.