Proceedings of the 2007 IEEE Symposium on Computational Intelligence and Data Mining (CIDM 2007)

Quantifying Privacy for Privacy Preserving Data Mining

Justin Zhan
Carnegie Mellon University
justinzh@andrew.cmu.edu

Abstract

Data privacy is an important issue in data mining. How to protect respondents' data privacy during the data collection and mining process is a challenge to the security and privacy community. In this paper, we describe two schemes for privacy preserving naive Bayesian classification which is one of data mining tasks. More importantly, for each scheme, we present a method to measure data privacy. We finally compare these two methods.

Key Words: Privacy Quantification, Data Mining, Naive Bayesain Classification.

1 Introduction

Data mining and knowledge discovery in databases are important research areas that investigate the automatic extraction of previously unknown patterns from large amounts of data. The field connects the three worlds of databases, artificial intelligence and statistics. The information age has enabled many organizations to gather large volumes of data. However, the usefulness of this data is negligible if meaningful information or knowledge cannot be extracted from it. Data mining and knowledge discovery attempt to answer this need. In contrast to standard statistical methods, data mining techniques search for interesting information without demanding an *apriori* hypotheses. As a field, it has introduced new concepts and algorithms such as association rule mining, classification, clustering, etc. Data mining techniques are widely used and are becoming more and more popular with time.

The term *privacy* is used frequently in ordinary language, yet there is no single definition of this term [4]. The concept of privacy has broad historical roots in sociological and anthropological discussions about how extensively it is valued and preserved in various cultures [6, 12, 13, 15]. Yet historical use of the term is not uniform, and there remains confusion over the meaning, value and scope of the concept of privacy. Privacy refers to the right of users to conceal their personal information and have some degree of control over the use of any personal information disclosed to others in [1, 3, 8]. So far, many schemes have been proposed for privacy preserving data mining. However, a challenging issue is how to define privacy. In this paper, we present two different approaches to deal with privacy preserving naive Bayesian classification, which is one of data mining tasks, and compare two privacy measurements that we proposed in [19, 20].

The rest of the paper is organized as follows: we discuss how to build naive Bayesian classifiers using multi-variant randomized response techniques in Section 2. In Section 3, we present how to apply homomorphic encryption techniques to build naive Bayesian classifiers. We give our conclusion in Section 4.

2 Building Naive Bayesian Classifiers Using Multi-variant Randomized Response Techniques

Zhan et al. [18] proposed to use the multi-variant randomized response technique (MRR) to address the problems of privacy preserving naive Bayesian classification. We describe their scheme in the following.

Suppose data are binary and there are N *private* attributes (A_1, A_2, ..., A_N) in a data set A. We construct N *personal* attributes (Y_1, Y_2, ..., Y_N). We want one *private* attribute (question) to pair with one *personal* attribute (question), therefore we make the number of attributes of Y and the number of attributes of A be equal. Let A and Y represent any logical expression based on those attributes $A_i (i \in [1, N])$ and $Y_i (i \in [1, N])$. For example, A can be $(A_1 = 0) \wedge (A_2 = 1)$ and Y can be $(Y_1 = 0) \wedge (Y_2 = 1)$.

Let $P(Y)$ be the proportion of the records in the personal data that satisfy $Y = \texttt{true}$. Let $P^*(A)$ be the proportion of the records in the whole *randomized* data set that satisfies $A = \texttt{true}$. Let $P(A)$ be the proportion of the records in the whole *non-randomized* data set that satisfy $A = \texttt{true}$. $P^*(A)$ can be observed from the randomized data, but $P(A)$, the actual proportion that we are interested in, cannot be observed from the randomized data because the non-randomized data set is not available to anybody; we

have to estimate $P(A)$. The goal of MRR is to find a way to estimate $P(A)$ from $P^*(A)$.

Multi-variant Randomized Response Scheme:

In this scheme, all the attributes including the class label either keep the same values or obtain the values from personal data. In other words, when sending the private data to the data collector, respondents either tell their answers to the private questions or tell their answers to the personal questions. The probability for the first event is θ, and the probability for the second event is $1-\theta$. For example, assume a respondent's attribute values A_1 and A_2 are 11 for private data; and the respondent's attribute values Y_1 and Y_2 are 01. The respondent generates a random number between 0 and 1; if the number is less than θ, she sends 11 to the data collector; if the number is bigger than θ, she sends 01 to the data collector. Since the data collector only knows θ which is the same for all respondents and does not know the random number generated by each respondent, he cannot know whether the respondent tells the values from private data or personal data. To simplify our presentation, we use $P(A(11))$ to represent $P(A_1 = 1 \wedge A_2 = 1)$, $P(Y(11))$ to represent $P(Y_1 = 1 \wedge Y_2 = 1)$ where"$\wedge$" is the logical `and` operator. Because the contributions to $P^*(A(11))$ partially come from $P(A(11))$, and partially come from $P(Y(11))$, we can derive the following equation:

$$P^*(A(11)) \quad = \quad P(A(11)) \cdot \theta + P(Y(11)) \cdot (1-\theta)$$

Since $P(Y(11))$ is known as Y is personal data, θ is determined before collecting the data, and $P^*(A(11))$ can be directly computed on the disguised (randomized) data set. By solving the above equation, we can obtain $P(A(11))$, the information needed to build a naive Bayesian classifier. The general model is described in the following:

$$P^*(A) \quad = \quad P(A) \cdot \theta + P(Y) \cdot (1-\theta) \qquad (1)$$

Introducing Naive Bayesian Classification:

The naive Bayesian classifier is one of the most successful algorithms in many classification domains. Despite of its simplicity, it is shown to be competitive with other complex approaches, especially in text categorization and content based filtering. The naive Bayesian classifier applies to learning tasks where each instance x is described by a conjunction of attribute values and where the target function $f(x)$ can take on any value from some finite set V. A set of training examples of the target function is provided, and a new instance is presented, described by the tuple of attribute values $< a_1, a_2, \cdots, a_n >$. The learner is asked to predict the target value for this new instance. Under a conditional independence assumption, i.e., $P(a_1, a_2, \cdots, a_n|v_j)$ = $\Pi_{i=1}^{n} P(a_i|v_j)$, a naive Bayesian classifier can be derived as follows:

$$v_{NB} = argmax_{v_j \in V} P(v_j) \Pi_{i=1}^{n} P(a_i|v_j)$$

$$= argmax_{v_j \in V} P(v_j) \Pi_{i=1}^{n} \frac{P(a_i \wedge v_j)}{P(v_j)}$$

Building Naive Bayesian Classifiers:

To build a NB classifier, we need to compute $P(v_j)$ and $P(a_j \wedge v_j)$. To compute $P(v_j)$, we can use the general model (Eq. 1) with A being $(C = v_j)$ and Y being $(CY = v_j)$ where C is the class label for the private data A and CY is the class label of personal data Y. $P^*(A)$ can be computed directly from the (whole) randomized data set. $P(Y)$ is known since it is personal and θ is known as well. By knowing θ, data collector, who conducts the training, only knows the probability of the training data being private, but does not exactly know if each value is private data or not. By solving the above equation, we can get $P(A)$ which is $P(C = v_j)$ in this case. Similarly, we can compute $P(a_i \wedge v_j)$ using the general model (Eq. 1) with A being $(A_i = a_i \wedge C = v_j)$ and Y being $(Y_i = a_i \wedge CY = v_j)$.

Measuring Privacy:

To quantify privacy, we first measure privacy for a single entry; we then select the minimal privacy value and treat it as the privacy level for the group. The reason why we choose the minimal value for the group is that, the entries are randomized together, by finding the original value for one entry will cause disclosing the original values for other entries in the group.

For a single entry, original value can be 1 or 0; randomized value can be 1 or 0 as well. Privacy comes from uncertainty of original value given a randomized value. In other words, if original value is 1, given randomized value 1 or 0, privacy will be the probability of data collector guess the original value being 0. There are four possible randomization results:

- Original value is 1, the value after randomization is still 1;
- Original value is 1 but the value after randomization is 0;
- Original value is 0 but the value after randomization is 1;
- Original value is 0, the value after randomization is still 0.

Consequentially, there are four components in the privacy measure:

- The probability that original value is 1, multiplies the probability that original value is 1 and the value after randomization is still 1 , then times the probability that guessing the original value is 0 given the randomized value is 1.

- The probability that original value is 1, multiplies the probability that original value is 1 but the value after randomization is still 0, then times the probability that guessing the original value is 0 given the randomized value is 0.

- The probability that original value is 0, multiplies the probability that original value is 0 but the value after randomization is 1 , then times the probability that guessing the original value is 1 given the randomized value is 1.

- The probability that original value is 0, multiplies the probability that original value is 0 and the value after randomization is still 0, then times the probability that guessing the original value is 1 given the randomized value is 0.

Let's use the following denotations:

- Let's O_m be the original value;

- Let's R_m be the value after randomization;

- Let's W_a be the probability that a value is 1 in data set A, and the probability that a value is 0 in data set A is $(1 - W_a)$;

- Let's W_y be the probability that a value is 1 in data set Y, and the probability that a value is 1 in data set Y is $(1 - W_y)$;

Privacy denoted by $PRE(PSE)$ for a single entry before mining can be derived as follows:

$$
\begin{aligned}
&PSE(PRE) = \\
&Pr(O_m = 1) * Pr(R_m = 1|O_m = 1) * Pr(O_m = 0|R_m = 1) \\
&\qquad + \\
&Pr(O_m = 1) * Pr(R_m = 0|O_m = 1) * Pr(O_m = 0|R_m = 0) \\
&\qquad + \\
&Pr(O_m = 0) * Pr(R_m = 1|O_m = 0) * Pr(O_m = 1|R_m = 1) \\
&\qquad + \\
&Pr(O_m = 0) * Pr(R_m = 0|O_m = 0) * Pr(O_m = 1|R_m = 0) \\
&= \\
&Component_1 + Component_2 \\
&\\
&+Component_3 + Component_4
\end{aligned}
$$

The first component can be computed as follows:

$$
\begin{aligned}
&Component_1 = \\
&W_a * [\theta + (1-\theta) * W_y] * \frac{Pr(R_m=1|O_m=0)*Pr(O_m=0)}{Pr(R_m=1)} \\
&\\
&= \frac{W_a*[\theta+(1-\theta)*W_y]*(1-\theta)*(1-W_y)*(1-W_a)}{Pr(R_m=1|O_m=1)*Pr(O_m=1)+Pr(R_m=1|O_m=0)*Pr(O_m=0)} \\
&\\
&= \frac{W_a*[\theta+(1-\theta)*W_y]*(1-\theta)*(1-W_y)*(1-W_a)}{[\theta+(1-\theta)*W_y]*W_a+(1-\theta)*(1-W_y)*(1-W_a)}
\end{aligned}
$$

Similarly, we can obtain other components.

$$
\begin{aligned}
&Component_2 = \\
&\frac{W_a*(1-\theta)*(1-W_y)*[\theta+(1-\theta)*(1-W_y)]*(1-W_a)}{[\theta+(1-\theta)*(1-W_y)]*(1-W_a)+(1-\theta)*(1-W_y)*W_a}
\end{aligned}
$$

$$
\begin{aligned}
&Component_3 = \\
&\frac{(1-W_a)*(1-\theta)*W_y*[\theta+(1-\theta)*W_y]*W_a}{(\theta+(1-\theta)*W_y)*W_a+(1-\theta)*W_y*(1-W_a)}, and
\end{aligned}
$$

$$
\begin{aligned}
&Component_4 = \\
&\frac{(1-W_a)*[\theta+(1-\theta)*(1-W_y)]*(1-\theta)*(1-W_y)*W_a}{[\theta+(1-\theta)*(1-W_y)]*(1-W_a)+(1-\theta)*(1-W_y)*W_a}
\end{aligned}
$$

We then get

$$
\begin{aligned}
&PSE(PRE) = \\
&\\
&\frac{W_a*(\theta+(1-\theta)*W_y)*(1-\theta)*(1-W_y)*(1-W_a)}{(\theta+(1-\theta)*W_y)*W_a+(1-\theta)*(1-W_y)*(1-W_a)} \\
&\qquad + \\
&\frac{W_a*(1-\theta)*(1-W_y)*(\theta+(1-\theta)*(1-W_y))*(1-W_a)}{(\theta+(1-\theta)*(1-W_y))*(1-W_a)+(1-\theta)*(1-W_y)*W_a} \\
&\qquad + \\
&\frac{(1-W_a)*(1-\theta)*W_y*(\theta+(1-\theta)*W_y)*W_a}{(\theta+(1-\theta)*W_y)*W_a+(1-\theta)*W_y*(1-W_a)} \\
&\qquad + \\
&\frac{(1-W_a)*(\theta+(1-\theta)*(1-W_y))*(1-\theta)*(1-W_y)*W_a}{(\theta+(1-\theta)*(1-W_y))*(1-W_a)+(1-\theta)*(1-W_y)*W_a} \\
&\\
&= \frac{(1-W_a)*W_a*(1-\theta)*(\theta+(1\theta)*W_y)}{(\theta+(1-\theta)*W_y)*W_a+(1-\theta)*W_y*(1-W_a)} \\
&\qquad + \\
&\frac{2*W_a*(1-W_a)*(1-\theta)*(1-W_y)*(\theta+(1-\theta)*(1-W_y)}{(\theta+(1-\theta)*(1-W_y))*(1-W_a)+(1-\theta)*(1-W_y)*W_a}
\end{aligned}
$$

3 Building Naive Bayesian Classifiers Using Homomorphic Encryption Techniques

In our privacy-oriented protocols, we use the additive homomorphism offered by [10] in which Paillier proposed a new trapdoor mechanism based on the idea that it is hard to factor a number n = pq where p and q are two large prime numbers. We utilize the following instantiation of the homomorphic encryption functions: $e(m_1) \times e(m_2) = e(m_1 + m_2)$ where m_1 and m_2 are the data to be encrypted. Because of the property of associativity, $e(m_1 + m_2 + .. + m_n)$ can be computed as $e(m_1) \times e(m_2) \times \cdots \times e(m_n)$ where $e(m_i) \neq 0$. That is

$$d(e(m_1 + m_2 + \cdots + m_n)) = d(e(m_1) \times e(m_2) \times \cdots \times e(m_n))$$

Note that a corollary of it is as follows:

$$d(e(m_1)^{m_2}) = d(e(m_1 \times m_2)),$$

where $\times$ denotes multiplication.

To build a naive Bayesian classifier, we need to conduct the following major steps:

1. To compute $Pr(a_i, v_j) = \prod_{i=1}^{n} Pr(a_i|v_j)Pr(v_j)$.
2. To Compute $Pr(v_j)\prod_{i=1}^{n} \frac{Pr(a_i,v_j)}{Pr(v_j)}$ for each $v_j \in V$.
3. To Compute V_{NB}.

Next, we will present a privacy-consicious scheme to deal with privacy preserving naive Bayesian classification for horizontal collaboration.

In horizontal collaboration, there are N parties denoted by P_1, P_2, $\cdots$, and P_n. Assume that P_1 has a private data set DS_1, P_2 has a private data set DS_2, $\cdots$ and P_n has a private data set DS_n. The goal is to compute $e(Pr(a_i, v_j))$, $e(Pr(v_j)\prod_{i=1}^{n} \frac{Pr(a_i,v_j)}{Pr(v_j)})$ for each $v_j \in V$, and V_{NB} for horizontal collaboration involving $DS_1, \cdots$, and DS_n without compromising data privacy.

To deal with the above problem, we provide the following protocols.

Highlight of Protocol 1: In our protocol, we first select a key generator. Let us assume that P_n is the key generator who generates a homomorphic encryption key pair (e, d). P_n encrypts $\prod_{a_i \in P_n} Pr(a_i|v_j)$ and sends it to P_1. P_1 computes $e(\prod_{a_i \in P_n} Pr(a_i|v_j) \cdot \prod_{a_i \in P_1} Pr(a_i|v_j))$ and sends it to P_2, and so on. Finally, P_n computes $e(\prod_{i=1}^{\tau} Pr(a_i|v_j))$.

We present the formal protocol as follows:

Protocol 1 .

1. *P_n generates a cryptographic key pair (e, d) of a homomorphic encryption scheme.*
2. *P_n computes $e(\prod_{a_i \in P_n} Pr(a_i|v_j))$ denoted by $e(G_n)$ and sends it to P_1.*
3. *P_1 computes $e(G_n)^{G_1} = e(G_1G_n)$ where $G_1 = \prod_{a_i \in P_1} Pr(a_i|v_j)$, then sends $e(G_1G_n)$ to P_2.*
4. *P_2 computes $e(G_1G_n)^{G_2} = e(G_1G_2G_n)$ where $G_2 = \prod_{a_i \in P_2} Pr(a_i|v_j)$, then sends $e(G_1G_2G_n)$ to P_3.*
5. *Continue until P_{n-1} obtains $e(G_1G_2 \cdots G_n) = e(\prod_{i=1}^{\tau} Pr(a_i|v_j))$.*

The Correctness Analysis of Protocol 1: When P_1 receives $e(G_n)$, he computes $e(G_n)^{G_1}$ which is equal to $e(G_1G_n)$ according to Equation 2. He sends it to P_2 who computes $e(G_1G_n)^{G_2}$ which is equal to $e(G_1G_2G_n)$ according to Equation 2. Continue to send the result to the next party. Finally, P_{n-1} obtains $e(G_1G_2 \cdots G_n) = e(\prod_{i=1}^{\tau} Pr(a_i|v_j))$. Therefore, the Protocol 1 correctly computes $e(\prod_{i=1}^{\tau} Pr(a_i|v_j))$.

To compute $e(Pr(v_j))$, each party computes $Pr(v_j)$ for their own class label set. Let assume that P_1 has the share s_1, P_2 has the share s_2, $\cdots$, P_n has the share s_n. Our goal is to compute $e(\sum_{i=1}^{n} s_i) = e(Pr(v_j))$. We can apply Protocol 2 to deal with this problem.

Protocol 2 .

1. P_n generates a cryptographic key pair (e, d) of a semantically secure homomorphic encryption scheme. P_n also generates an integer X which is greater than N.
2. P_1 computes $e(s_1 + R_1 \times X)$ and sends it to P_2 where R_1 is a random integer generated by P_1.
3. P_2 computes $e(s_1 + R_1 \times X) \times e(s_2 + R_2 \times X) = e(s_1 + s_2 + (R_1 + R_2)X)$ and sends it to P_3. R_2 is a random integer generated by P_2.
4. Repeat until P_n computes $e(s_1 + R_1 \times X) \times e(s_2 + R_2 \times X) \times \cdots \times e(s_n + R_n \times X) = e(\sum_{i=1}^{n} s_i + \sum_{i=1}^{n} R_i \times X)$.
5. P_n computes $d(e(\sum_{i=1}^{n} s_i + (\sum_{i=1}^{n} R_i) \times X))\ modX = (\sum_{i=1}^{n} s_i + (\sum_{i=1}^{n} R_i) \times X)\ modX = \sum_{i=1}^{n} s_i$.

The Correctness Analysis of Protocol 2: To show the s is correct, we need to consider:

$$d[e(s_1) \times e(s_2) \times \cdots \times e(s_n)]$$
$$= d[e(s_1 + R_1 \times X) \times e(s_2 + R_2 \times X) \times \cdots \times e(s_n + R_n \times X)] modX.$$

The left hand side

$$d[e(s_1) \times e(s_2) \times \cdots \times e(s_n)] = \sum_{i=1}^{n} s_i.$$

The right hand side

$$d[e(s_1 + R_1 \times X) \times e(s_2 + R_2 \times X) \times \cdots \times e(s_n + R_n \times X)] modX$$
$$= [\sum_{i=1}^{n} s_i + \sum_{i=1}^{n} R_i) \times X] modX.$$

Since $X > N$, $\sum_{i=1}^{n} s_i \leq N$, and $\sum_{i=1}^{n} R_i$ is an integer,

$$[\sum_{i=1}^{n} s_i + (\sum_{i=1}^{n} R_i) \times X] modX = \sum_{i=1}^{n} s_i.$$

Therefore, the $\sum_{i=1}^{n} s_i$ is correctly computed.

We follow the Protocol 2 until P_{n-1} obtains $e(\sum_{i=1}^{n} s_i) = e(Pr(v_j))$. Next, we use Protocol 3 to compute $e(Pr(v_j) \prod_{i=1}^{\tau} Pr(a_i|v_j))$.

Highlight of Protocol 3: In our protocol, P_{n-1} generates t random numbers from the real domain, sends $e(Pr(v_j))$, $e(r_1)$, $\cdots$, $e(r_t)$ to P_n in a random order. P_n decrypts them and sends the decrypted sequence to P_1. P_1 and P_{n-1} jointly computes $e(Pr(v_j) \prod_{i=1}^{\tau} Pr(a_i|v_j))$.

We present the formal protocol as follows:

Protocol 3 .

1. *P_{n-1} generates a set of random numbers: R_1, R_2, $\cdots$, R_t. He then sends $e(Pr(v_j))$, $e(R_1)$, $\cdots$, $e(R_t)$ to P_n in a random order.*
2. *P_n decrypts each element in the sequence, then sends them to P_1 in the same order as P_{n-1} did.*
3. *P_{n-1} sends $e(\prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))$ to P_1.*
4. *P_1 computes $e(\prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))^{Pr(v_j)}$, $e(\prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))^{R_1}$, $\cdots$, $e(\prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))^{R_t}$. P_1 then sends them to P_{n-1}.*
5. *P_{n-1} obtains $e(Pr(v_j) \prod_{i=1}^{\tau} Pr(a_i|v_j))$.*

The Correctness Analysis of Protocol 3: In step 4, P_1 computes $e(\prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))^{Pr(v_j)}$, $e(\prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))^{R_1}$, $\cdots$, and $e(\prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))^{R_t}$. They are equal to $e(Pr(v_j) \prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))$, $e(R_1 \prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))$, $\cdots$, and $e(R_t \prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))$ respectively according to Equation 2. In step 5, P_{n-1} gets $e(Pr(v_j) \prod_{i=1}^{\tau} Pr(Pr(a_i, v_j)))$ since he knows the permutations.

Through the above protocol, $e(Pr(v_j) \prod_{i=1}^{n} Pr(a_i|v_j))$ can be computed for each $v_j \in V$. Without loss of generality, Let us assume P_1 gets $e(V_{NB_1})$, $e(V_{NB_2})$, $\cdots$, $e(V_{NB_k})$ The goal is to find the largest one which can be achieved by Protocol 4 in [16].

Measuring Privacy:

In this following, we first present some notations, we then quantify the privacy according to our definition [17].

- We use ADV_{P_i} to denote P_i's advantage to gain access to the private data of any other party via the component protocol.
- $Pr(T_{P_i}|VIEW_{P_j}, Protocol\varsigma)$: the probability that p_j sees P_i's private data via protocol ς.
- We use ADV_S to denote the advantage of one party to gain the other party's private data via the component protocol by knowing the semantically secure encryptions. According to definition of semantic security [7], ADV_S is negligible.

Theorem 1 *Protocol 1 preserves data privacy at a level equal to ADV_S.*

Proof 1 *We will identify the value of ϵ such that*

$$|Pr(T|CP) - Pr(T)| \leq \epsilon$$

holds for $T = T_{P_i}$, $i \in [1, n]$, and CP = Protocol 1.

According to our notation,

$$ADV_{P_i} = Pr(T_{P_j}|VIEW_{P_i}, Protocol1) - Pr(T_{P_j}|VIEW_{P_i}), j \neq i, i \neq n.$$

Since all the information that P_i obtains from other parties is the encrypted by e which is semantically secure,

$$ADV_{P_i} = ADV_S.$$

In order to show that privacy is preserved, we need to know the value of the privacy level ϵ. We set

$$\epsilon = ADV_S.$$

Then

$$Pr(T_{P_j}|VIEW_{P_i}, Protocol1) - Pr(T_{P_j}|VIEW_{P_i}) \leq ADV_S, j \neq i, i \neq n,$$

which completes the proof.

Theorem 2 *Protocol 2 preserves data privacy at a level equal to ADV_{P_n}.*

Proof 2 *We will identify the value of ϵ such that*

$$|Pr(T|CP) - Pr(T)| \leq \epsilon$$

holds for $T = T_{P_i}$, $i \in [1, n]$, and CP = Protocol 2.

According to our notation,

$$ADV_{P_i} = Pr(T_{P_j}|VIEW_{P_i}, Protocol2) - Pr(T_{P_j}|VIEW_{P_i}), i \neq n,$$

and

$$ADV_{P_n} = Pr(T_{P_j}|VIEW_{P_n}, Protocol2) - Pr(T_{P_j}|VIEW_{P_n}),$$

where ADV_{P_n} is the advantage of P_n to gain access to other parties' private data by obtaining the final result $\sum_{i=1}^{n-1} s_i$.

Since P_1 obtains no data from other parties, $ADV_{P_1} = 0$. For P_2, $\cdots$, P_{n-1}, all the information that each of them obtains about other parties' data is encrypted, thus,

$$ADV_{P_i} = ADV_S,$$

which is negligible.

In order to show that privacy is preserved, we need to know the value of the privacy level ϵ. We set

$$\epsilon = max(ADV_{P_i}, ADV_{P_n}) = max(ADV_S, ADV_{P_n}) = ADV_{P_n}.$$

Then

$$Pr(T_{P_j}|VIEW_{P_i}, Protocol2) - Pr(T_{P_j}|VIEW_{P_i}) \leq ADV_{P_n}, i \neq n,$$

and

$$Pr(T_{P_j}|VIEW_{P_n}, Protocol2) - Pr(T_{P_j}|VIEW_{P_n}) \leq ADV_{P_n},$$

which completes the proof.

Theorem 3 *Protocol 3 preserves data privacy at a level equal to ADV_{P_n}.*

Proof 3 *We will identify the value of ϵ such that*

$$|Pr(T|CP) - Pr(T)| \leq \epsilon$$

holds for $T = T_{P_i}$, $i \in [1, n]$, and CP = Protocol 3.

According to our notation,

$$ADV_{P_1} = Pr(T_{P_j}|VIEW_{P_1}, Protocol3) - Pr(T_{P_j}|VIEW_{P_1}), j \neq 1,$$

$$ADV_{P_{n-1}} = Pr(T_{P_i}|VIEW_{P_{n-1}}, Protocol3) - Pr(T_{Others}|VIEW_{P_{n-1}}), i \neq n-1,$$

and

$$ADV_{P_n} = Pr(T_{P_i}|VIEW_{P_n}, Protocol3) - Pr(T_{P_i}|VIEW_{P_n}), i \neq n.$$

Since all the information that P_1 and P_{n-1} obtain from other parties is encrypted by e which is semantically secure,

$$ADV_{P_1} = ADV_S,$$

and

$$ADV_{P_{n-1}} = ADV_S.$$

In order to show that privacy is preserved, we need to know the value of the privacy level ϵ. We set

$$\epsilon = max(ADV_{P_n}, ADV_{P_{n-1}}, ADV_{P_1}) = max(ADV_{P_n}, ADV_S) = ADV_{P_n}.$$

Then

$$Pr(T_{P_j}|VIEW_{P_1}, Protocol3) - Pr(T_{P_j}|VIEW_{P_1}) \leq ADV_{P_n}, j \neq 1,$$

$$Pr(T_{P_i}|VIEW_{P_{n-1}}, Protocol3) - Pr(T_{P_i}|VIEW_{P_{n-1}}) \leq ADV_{P_n}, i \neq n-1,$$

and

$$Pr(T_{P_i}|VIEW_{P_n}, Protocol3) - Pr(T_{P_i}|VIEW_{P_n}) \leq ADV_{P_n}, i \neq n,$$

which completes the proof [1].

[1] Note that the information that P_n obtains from P_{n-1} is hidden by t random numbers.

4 Conclusion

To achieve privacy preserving data mining, a set of schemes have been proposed such as Secure Multi-party Computation- based (SMC) techniques [9, 14], randomization-based approach [2, 11, 5], crypto-based approach [16, 19], etc. An important issue among privacy-consicious schemes is privacy quantification. In this paper, we have presented two approaches to build naive Bayesian classifiers using multi-variant randomized response technique and homomorphic encryption. We describe two different privacy quantification schemes. The first one can be treated as a special case of the second one. In the future, we would like to follow the second method to quantify the data privacy.

References

[1] M. Ackerman, L. Cranor, and J. Reagle. Privacy in e-commerce: Examining user scenarios and privacy preferences. In *Proceedings of the ACM Conference on Electronic Commerce, pages 1-8, Denver, Colorado, USA, November*, 1999.

[2] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proceedings of the ACM SIGMOD Conference on Management of Data*, pages 439–450. ACM Press, May 2000.

[3] S. Cockcroft and P. Clutterbuck. Attitudes towards information privacy. In *Proceedings of the 12th Australasian Conference on Information Systems, Australia*, 2001.

[4] J. DeCew. Privacy. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. 2002.

[5] W. Du and Z. Zhan. Using randomized response techniques for privacy-preserving data mining. In *Proceedings of The 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Washington, DC, USA. August 24 - 27*, 2003.

[6] S. Garfinkel. Database nation: The death of the privacy in the 21st century. In *O'Reilly Associates, Sebastopol, CA, USA*, 2001.

[7] S. Golwasser and S. Micali. Probabilistic encryption. journal of computer and system sciences 28, 270299. 1984.

[8] P. Jefferies. Multimedia, cyberspace and ethic. In *Proceedings of International Conference on Information Visualisation, pages 99-104, London, England, July*, 2000.

[9] Y. Lindell and B. Pinkas. Privacy preserving data mining. In *Advances in Cryptology - Crypto2000, Lecture Notes in Computer Science*, volume 1880, 2000.

[10] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptography - EUROCRYPT '99, pp 223-238, Prague, Czech Republic*, 1999.

[11] S. Rizvi and J. Haritsa. Maintaining data privacy in association rule mining. In *Proceedings of the 28th VLDB Conference*, Hong Kong, China, 2002.

[12] A. Rosenberg. Privacy as a matter of taste and right. In *E. F. Paul, F. D. Miller, and J. Paul, editors, The Right to Privacy, pages 68-90, Cambridge University Press*, 2000.

[13] F. D. Schoeman. Philosophical dimensions of privacy. In *Cambridge University Press*, 1984.

[14] J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada, July 23-26*, 2002.

[15] A. F. Westin. The right to privacy. In *New York: Atheneum*, 1967.

[16] J. Zhan. *Privacy Preserving Collaborative Data Mining*. PhD thesis, Department of Computer Science, University of Ottawa, 2006.

[17] J. Zhan. Using homomorphic encryption for privacy preserving collaborative decision tree classification. In *IEEE Symposium on Computational Intelligence and Data Mining, Honolulu, Hawaiia, USA, April 1-5*, 2007.

[18] J. Zhan, L. Chang, and S. Matwin. Privacy-preserving data mining in electronic surveys. In *4th International Conference on Electronic Business, Beijing, China, Dec. 5-9.*, 2004.

[19] J. Zhan and S. Matwin. A crypto-based approach to privacy preserving collaborative data mining. In *Workshop on Privacy Aspect of Data Mining (PADM'06) in conjunction with the IEEE International Conference on Data Mining (ICDM'06), HongKong, December 1*, 2006.

[20] J. Zhan and S. Matwin. Privacy-preserving data mining in electronic surveys. *International Journal of Network Security*, to appear.