Bloom Filter Bootstrap: Privacy-Preserving Estimation of the Size of an Intersection

Hiroaki Kikuchi[1,⋆] and Jun Sakuma[2]

[1] Department of Frontier Media Science,
School of Interdisciplinary Mathematical Sciences,
Meiji University 4-21-1 Nakano, Nakano Ku, Tokyo, 164-8525 Japan
kikn@meiji.ac.jp
[2] Graduate School of SIE, Computer Science Department,
University of Tsukuba 1-1-1 Tennodai, Tsukuba, 305-8573 Japan
jun@cs.tsukuba.ac.jp

Abstract. This paper proposes a new privacy-preserving scheme for estimating the size of the intersection of two given secret subsets. Given the inner product of two Bloom filters (BFs) of the given sets, the proposed scheme applies Bayesian estimation under assumption of beta distribution for an a priori probability of the size to be estimated. The BF retains the communication complexity and the Bayesian estimation improves the estimation accuracy.

An possible application of the proposed protocol is an epidemiological datasets regarding two attributes, *Helicobactor pylori* infection and stomach cancer. Assuming information related to Helicobactor Pylori infection and stomach cancer are separately collected, the protocol demonstrates that a χ^2-test can be performed without disclosing the contents of the two confidential databases.

1 Introduction

With the rapid development of database systems and online services, large amounts of information are being collected and accumulated from various data sources independently and simultaneously. Privacy- preserving data mining (PPDM) has been attracting significant attention as a technology that could enable us to perform data analysis over multiple databases containing sensitive information without violating subjects' privacy.

In this paper, we investigate the problem of set intersection cardinality. Given two private sets, the goal of this problem is to evaluate the cardinality of the intersection without disclosing the sets mutually. Set intersection cardinality has been extensively studied as a building block of PPDM, including association rule mining [17], model and attribute selection[16], and other aspects [4]. Our major application of this problem is epidemiological analysis, including

⋆ This work was done when author was in Tokai University.

L. Wang and B. Shafiq (Eds.): DBSec 2013, LNCS 7964, pp. 145–163, 2013.

privacy-preserving cohort studies. We wish to perform cohort studies over multiple independently collected medical databases, which are not allowed to disclose identifying information about patients.

Consider two databases developed independently by two organizations. One organization collects individual medical information, including patient ID, patient name, patient address, presence or absence of disease 1, disease 2, and so on. The other organization collects individual genome information from research participants; including participant ID, participant name, participant address, presence or absence of genome type 1, genome type 2, and so on. The objective of a cohort study may be to investigate the association between the outbreak of a specific disease and genomes. For this analysis, the analyst makes use of four-cell contingency tables; each cell counts the number of patients who have (do not have) a specific disease and have (do not have) a specific genome type. If both tables are private, the set intersection cardinality may be used for evaluating of the count of each cell without sharing database content. In this study, we consider the following four requirements for practical situations.

Requirement 1. The time and communication complexity should be linear with respect to the number of records n. This is because statistical analysis, including cohort studies, usually treats databases with a large number of records.

Requirement 2. The time and communication complexity should be independent of the size of the ID space. In the use case described above, both organizations independently collect information from individuals. Thus, unique IDs are not given to records. Instead, the protocol must generate a unique ID for each record with the combination of individual attributes, such as the name and address. Because the space required for the combination of such user attributes is often much larger than the number of individuals, this requirement is important.

Requirement 3. The protocol should be designed considering the asymmetry of computational capabilities of organizations. Assume that a research institute that holds genome information provides epidemiological analysis services upon request to hospitals that hold medical information. In such a case, it is expected that the computational capabilities of the hospitals are poor. Therefore, a reasonable solution can be the outsourcing of computation; the research institute offers servers with high computational power and the hospital outsources most of the computation required for the analysis to the research institute. This example indicates that the protocol of set intersection cardinality should be designed considering the asymmetry of computational capabilities.

Requirement 4. The outputs of the protocol may be random shares. This requirement implicitly suggests that the set intersection cardinality may be used as a part of a larger-scale protocol. If the outputs of the protocol are random shares, these can be seamlessly used for inputs to other privacy-preserving protocols.

In this paper, we propose a set intersection cardinality protocol that satisfies these requirements.

Related Work

Let S_A and S_B be private inputs of the set intersection cardinality. Let n_A and n_B be the cardinalities of S_A and S_B, respectively.

Agrawal et al. [1] presented a set intersection cardinality protocol using commutative encryption under DDH (Decisional Diffie-Hellman) assumption. The time complexity of this protocol is $O(n_A + n_B)$; this is linear in the size of the databases and is independent of the size of the ID space. However, this protocol assumes that the two parties have nearly the same computation power. Furthermore, the protocol cannot output random shares. De Cristofaro and Tsudik [5] introduced an extension of [1]. It also requires $O(n)$ computation by both parties.

Freedman et al. [7] proposed a set intersection protocol using oblivious polynomial evaluation. This protocol can be converted to the set intersection cardinality with a slight modification, and achieves $O(n_B + \log\log n_A)$ time/communication complexity. Furthermore, the time complexity is independent of the ID space size and random shares can be output. This protocol also assumes that both parties have equal computational power.

All the above protocols guarantee exact outputs. Kantarcioglu et al. [11] approach the set intersection cardinality differently. Their protocol maps the input set onto a binary vector using a Bloom filter (BF)[2], and the set intersection cardinality is statistically estimated from the scalar product of the two binary vectors. With this approach, the results become approximations, although the computation cost is expected to be greatly reduced. The dimensionality of the vector used in this protocol is equal to the ID space size; this does not meet Requirement 2. In [11], a technique to shorten large IDs using hash functions was used with their protocol. As shown later by our theoretical analysis, given an error rate ϵ, the optimal range of hash functions for n elements is $O(n^2)$. This indicates that such Naive ID generation can be too inefficient for practical use.

Camenisch and Zaverucha [3] has introduced the certified set intersection cardinality problem. This protocol considers asymmetry in the security assumptions of the parties, but does not consider asymmetry in their computational capability.

Ravikumar et al. used the TF-IDF measures to estimate the scalar product in [15]. As for epidemiological study, Lu et al. studied the contingency tables in [13].

Thus, to our knowledge, no set intersection cardinality protocol satisfies the four requirements above, which should be met for practical privacy-preserving data analysis, especially for the outsourcing models.

Our Contribution

In this manuscript, we present a protocol that satisfies the four requirements. Considering the first and second requirement, the sets are independently mapped onto BFs, and then the set intersection cardinality is statistically estimated from the scalar product of the two binary vectors representing the BFs.

As discussed later, the size of the BF must be $O(n^2)$ to control the false positive rate in [11]; this does not meet Requirement 2. Our protocol therefore uses a number of BFs of size $O(n)$. The set intersection cardinality is obtained by iteratively applying Bayesian estimation to the scalar products of the BFs.

In the proposed protocol, the scalar product protocol is used as a building block. Modulo exponentiation is performed only by one party and this fits well with the outsourcing model (Requirement 3). In addition, the outputs can naturally be made random shares (Requirement 4).

We demonstrate our protocol with an epidemiological datasets regarding two attributes, *Helicobactor pylori* infection and stomach cancer. Assuming information related to Helicobactor Pylori infection and stomach cancer are separately collected, we demonstrate that a χ^2-test can be performed without disclosing the contents of the two databases.

2 Preliminary

2.1 Bloom Filter

A BF is a simple space-efficient data structure for representing a set to support membership queries[2]. Recently, BFs have been used not only for database applications but also for network problems including detecting malicious addresses, packet routing, and the measurement of traffic statistics.

A BF for representing a set $S = \{a_1, \ldots, a_n\}$ of n elements is an array of m bits, initially all set to 0. The BF uses k independent hash functions $H_1, \ldots, H_k$ such that $H_i : \{0,1\}^* \to \{1, \ldots, m\}$. The hash functions map each element in the map to a random number uniformly chosen from $\{1, \ldots, m\}$. Let $B(S)$ be a set representing a BF defined by $B(S) = \bigcup_{a \in S} B(a)$ such that $B(a) = \{H_1(a), \ldots, H_k(a)\}$. Now let $\boldsymbol{b}$ be an m-dimensional vector, $(b_1, \ldots, b_m)$, which is an alternative representation of the BF, defined by $b_i = \begin{cases} 1 \text{ if } i \in B(S), \\ 0 \text{ if } i \notin B(S), \end{cases}$ for $i = 1, \ldots, m$. For example, the hash functions that map an element a as $H_1(a) = 2$, $H_2(a) = 7$ characterize a BF with $m = 8$, $B(a) = \{2, 7\}$. Alternatively, $\boldsymbol{b}(a) = (0, 1, 0, 0, 0, 0, 1, 0)$. We can use either the set or vector representation of BF, depending on the cryptographic building blocks used. Note the following relationship between the set and vector representations, $\boldsymbol{b}(S_1) \cdot \boldsymbol{b}(S_2) = |B(S_1) \cap B(S_2)|$.

To test if a is an element of set S, we can verify that

$$\forall i = 1, \ldots, k \, H_i(a) \in B(S), \tag{1}$$

which holds if $a \in S$. However, it also holds, with a small probability, even if $a \notin S$. That is, BFs suffer from false positives. According to [2], after all the elements of S are hashed into the BF, the probability that element i does not belong to $B(S)$, i.e., that the i-th bit of $\boldsymbol{b}(S)$ is 0, is $p = \left(1 - \frac{1}{m}\right)^{kn} \approx e^{-kn/m}$. We therefore have a probability of false positives given by $p' = \left(1 - (1 - \frac{1}{m})^{kn}\right)^k \approx \left(1 - e^{-kn/m}\right)^k$. If k is sufficiently small for given m and n, Equation (1) is likely to hold only for the element of S. Conversely, with too large a value for k, the BF is mostly occupied by 1 values. In [2,6], the optimal BF was found for $k^* = \ln 2 \cdot (m/n)$, which minimized the false-positive probability.

2.2 Cryptographic Primitives

Secure Scalar Product. The scalar product of two vectors is performed securely by using a public-key encryption algorithm in Algorithm 1.

Algorithm 1. Secure Scalar Product

Input: Alice has an n-dimensional vector $\boldsymbol{x} = (x_1, \ldots, x_n)$. Bob has an n-dimensional vector $\boldsymbol{y} = (y_1, \ldots, y_n)$.
Output: Alice has s_A and Bob has s_B such that $s_A + s_B = \boldsymbol{x} \cdot \boldsymbol{y}$.

1. Alice generates a homomorphic public-key pair and sends the public key to Bob.
2. Alice sends to Bob n ciphertexts $E(x_1), \ldots, E(x_n)$, encrypted with her public key.
3. Bob chooses s_B at random, computes $c = E(x_1)^{y_1} \cdots E(x_n)^{y_n} / E(s_B)$ and sends c to Alice.
4. Alice uses her secret key to decrypt c to obtain $s_A = D(c) = x_1 y_1 + \cdots + x_n y_n - s_B$

Security Model. We assume that the parties are *honest-but-curious*, which is known as *semi-honest* model, with parties that own private datasets following protocols properly but trying to learn additional information about the datasets from received messages.

We also assume the Decisional Diffie-Hellman hypothesis (DDH), that is, a distribution of (g^a, g^b, g^{ab}) is indistinguishable from a distribution of (g^a, g^b, g^c), where a, b, c are uniformly chosen from Z_q.

3 Difficulties in ID-less Datasets

3.1 Problem Definition

We are considering the problem of a two-party protocol that can evaluate the size of the intersection of two sets without revealing the sets themselves.

Let A and B be parties owing subsets S_A and S_B, respectively. For an agreed threshold t, they each wish to know if

$$X = |S_{A \cap B}| = |S_A \cap S_B| \geq t \tag{2}$$

is true, without revealing S_A or S_B to the other party. Here, X is a random variable describing the size of the intersection $S_{A\cap B}$.

Note that we are not interested in learning about the intersection, itself but are only interested in evaluating its size because the size is often useful in many privacy-preserving applications. For example, an epidemic study might test if the difference between two subsets is statistically significant. The difference of $|X_{A\cap B}|$ and t may even be confidential in some applications.

3.2 Naïve ID Generation

Consider a dataset of n elements with multiple attributes, such as name, sex, age and address, but with no unique identity being assigned. Instead, the elements are uniquely specified by attributes, e.g., name and birthday. Let A be a set of attrbutes $A = \{a_1, \ldots, a_n\}$.

The simplest way to generate a pseudo identity is to use a hash function $h : \{0,1\}^* \to \{1, \ldots, \ell\}$. Using this hash function, we assign $h(a_i)$ to the i-th element. For efficiency reasons, we assume the range is sufficiently large that we can neglect the occurrence of a collision such that $h(a_i) = h(a_j)$ for some $i \neq j$. Letting h_A be the set of all pseudo identities, defined as $h_A = \{h(a_i) \mid a_i \in A\}$, we can see observe any collision of identities by testing whether $|h_A| = n$.

If the size ℓ of the ID set increases, collisions can be avoided, but the computational cost will accordingly increase with ℓ. Clearly, $\ell \geq n$, but finding the optimal size is not trivial. To solve the tradeoff between accuracy and performance reduction, let us assume we have an optimal ℓ that is sufficiently large to uniquely determine the given set of n elements.

This problem is equivalence to the problem known as "birthday paradox", whereby, among a set of n randomly chosen people, there is a probability that some pair of them have the same birthday. When identities (birthdays) are chosen with a uniform probability of $1/\ell$, the probability that all n identities are unique is given by

$$\prod_{j=1}^{n-1}\left(1-\frac{j}{\ell}\right) \approx \prod_{j=1}^{n-1} e^{-j/\ell} = e^{-n(n-1)/2\ell} \approx e^{-n^2/2\ell}.$$

Therefore, given the probability ϵ with which n hash values are unique, we have $\frac{n^2}{2\ell} = \ln \epsilon^{-1}$, from which follows the solution of our problem. The optimal range of hash functions for n elements is given as $\ell = n^2/2 \ln \epsilon^{-1}$, for which n elements will have distinct identities with a probability of ϵ.

3.3 Kantarcioglu's Scheme

In [11], Kantarcioglu, Nix and Vaidya proposed the following cryptographic protocol using BF in an approximate algorithm for the threshold scalar (dot) product.

Let Y be a random variable representing the number of matching bits in the two BFs of S_A and S_B. That is, Y is defined by $Y = |B(S_A) \cap B(S_B)|$. There is

a positive correlation between X, defined by true size of intersection $S_{A\cap B}$, and Y, which enables us to predict X from Y which can be obtained from BFs in a secure way.

Based on the properties of BFs [2], Equation (2) is equivalent to

$$Z_A + Z_B + Z_{AB} \geq Z_A Z_B \frac{1}{m}(1 - \frac{1}{m})^{-kt}, \tag{3}$$

where Z_A (Z_B) is the number of 0s in $B(S_A)$ ($B(S_B)$), respectively. Z_{AB} is the number of matching 0s in the two BFs of S_A and S_B. That is, $Z_{AB} = m - |B(S_A) \cap B(S_B)| = m - Y$. To evaluate the inequality privately, Kantarcioglu et al. performs a secure protocol for the scalar product of two vectors [8] to obtain u_1 and u_2 such that $\boldsymbol{b}(S_A) \cdot \boldsymbol{b}(S_B) = m - Z_{AB} = u_1 + u_2$ and a secure protocol for the multiplication of two integers Z_A and Z_B to obtain v_1 and v_2 such that $v_1 + v_2 = (1 - 1/m)^{-kt}/mZ_AZ_B$. Finally, they use SFE for the shared comparison of two integers to test if $(Z_A + u_1 - m) + (Z_B + u_2) \geq (v_1 + v_2)$.

According to their experimental results [11], their approximation algorithm using BFs with $m = 3,000, k = 2$, and $n = 20,000$ ran in 4 minutes, whereas an *exact* version required 27 minutes.

3.4 Difficulties in ID-less Datasets

In [11], Kantarioglu et al. claim that as long as, $m \ll n$, their method would be much faster than the typical implementation of a secure scalar (dot) product protocol[1]. Their experimental results show that the accuracy of approximation increases as m increases[2]. We will show that these properties do not hold in our target, *ID-less datasets* model, where the two datasets have no consistent identities and hence n elements are specified with some unique attribute(s).

1. **(Accuracy)** The size of intersection is approximated in their scheme based on the expected value of probability of common bits in BFs. The accuracy is expected to be improved as m increases. However, this is not true in large m because that the vector becomes too sparse. To be adaptively dense vector, we must increase the number of hash functions, k. This is not trivial. In [11], the experimental behavior with some parameters were shown and no guarantee in accuracy.
2. **(Performance)** The size of BF, m, increases up to n^2 in ID-less datasets. As we discussed in Section 3.2, the range of hash function should be as large as n^2 in order to minimize the probability to fail to uniquely identify elements. This is too large to find the intersection since some schemes running in $O(n)$ complexity in private set intersection are known, e.g., [1], [5].

[1] In Section 2.2 (Computation and Communicational cost). In Section 3, they assume that the vector of 20000 elements, whose density was 10 %, that is, the vector contains 2000 1's ($= n$), and it performs 20000-dimensional vector's scalar product for exact match and $m = 3000$ BF for their scheme.

[2] In Section 3.1, Figure 1(b).

Table 1. Comparison between [11] and ours

item	[11]	Proposed
approximation	Equation (3)	Equation (7), (6)
priori distribution	–	Beta distribution
BF size (m)	large (n^2)	small ($n/\ln 2$)
accuracy	no guarantee	improved with Bayesian estimation from s tests

3. **(Overhead)** Their scheme requires the secure multiplication as well as scalar product. It is not necessary in private set intersection.

In later section, we will present our scheme which overcomes the above limitations. Table 1 gives a summary of comparison between the scheme in [11] and proposed scheme.

4 Proposed Scheme

4.1 Probability Distribution of Matching Bits in BFs

Suppose that given $S_{A\cap B} = S_A \cap S_B$, random variable X of the cardinality of $S_{A\cap B}$, and instance $x = X$, we wish to estimate the number of matching 1s bits in their two BFs, i.e., $y = |B(S_A) \cap B(S_B)|$. The quantity y is equal to the number of 1s values in the conjunction of the two BF vectors. This subsection presents the mathematical properties of BFs, which will be used to estimate X in the subsequent subsection.

An element a in $S_A \cup S_B$ belongs to either $S_{A\cap B}$ or $S_A \cup S_B - S_{A\cap B}$. The former case always ensures that $a \in B(S_A) \cap B(S_B)$. Therefore, the probability that a certain bit in the conjunction of BFs is 0 after k random bits are set to 1 is $q_X = (1 - \frac{1}{m})^{kx}$. In the latter case, an element in $S_A \cup S_B - S_{A\cap B}$ does not always have a value of 1 because it yields a false positive. That is, an element a in S_A can have the same hash value $H_i(a) = H_j(b)$ as some element $b \neq a$ in S_B. The probability that a certain bit is 0 in the BF for a in $S_A - S_{A\cap B}$ is $q_A = (1 - \frac{1}{m})^{k(n_A - x)}$. Similarly, the BF of an element in $S_B - S_{A\cap B}$ having a certain bit being 0 has a probability of $q_B = (1 - 1/m)^{(n_B - x)k}$. Therefore, the probability of a certain bit in the BF for $S_A \cup S_B - S_{A\cap B}$ being 1 is given by the product of the compliment of each event, namely $(1 - q_A)(1 - q_B) = 1 - q_A - q_B + q_A q_B$.

Because the conjunction of BF has 1 for a certain bit by being either an element of $S_{A\cap B}$ or $S_A \cup S_B - S_{A\cap B}$, we have the probability θ for a bit being 1 as the disjunction of the two events namely

$$\begin{aligned}\theta &= 1 - q_X(1 - (1 - q_A)(1 - q_B)) \\ &= 1 - (1 - \frac{1}{m})^{kn_A} - (1 - \frac{1}{m})^{kn_B} + (1 - \frac{1}{m})^{k(n_A + n_B - x)}. \end{aligned} \tag{4}$$

Consequently, the conditional probability of $Y = |B(S_A) \wedge B(S_B)|$ being y, given $x = |S_A \cap S_B|$, is given by the binomial distribution $B(m, \theta)$, of m independent binary events with success probability θ. That is,

$$Pr(Y = y|X = x) = \binom{m}{y}\theta^y(1-\theta)^{m-y}. \quad (5)$$

4.2 Bayesian Estimation of X

Given known parameter values and $Pr(X|Y)$, we wish to identify the posterior distribution $Pr(Y|X)$ using Bayes' rule.

One possible solution is an approximation based on a the likelihood value from a single observation, as described by Kantarcioglu et al. [11]. Their scheme suffers from the complexity of $O(m)$. That is, a secure scalar product will require m ciphertexts, which is greater than n. Moreover, the accuracy achieved is inadequate.

Instead, we will use recursive Bayesian estimation using several small BFs. That is more efficient because each individual BF used to perform the secure scalar product between two BFs will be smaller. Moreover, the iteration over multiple BFs improves the accuracy of the estimation. Given the properties of beta distribution, the iteration process can be performed with lightweight overheads.

Using the conjugate prior distribution of Equation (5), we assume a beta distribution $Be(\alpha, \beta)$, which gives

$$Pr(\theta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{\int_0^1 \theta^{\alpha-1}(1-\theta)^{\beta-1}dy}.$$

The initial prior distribution is given by $Be(1, 1)$, which yields a uniform distribution $Pr(\theta) = 1$. Using Bayes' theorem, we obtain the posterior probability of θ given y as

$$Pr(\theta|y) = \frac{Pr(\theta)Pr(y|\theta)}{\int Pr(\theta)Pr(y|\theta)d\theta} \propto Pr(\theta)Pr(x|\theta) \propto \theta^{\alpha-1+y}(1-\theta)^{\beta-1+m-y},$$

which results again in a beta distribution $Be(\alpha', \beta')$ with new parameters as $\alpha' = \alpha + y$, and $\beta' = \beta + m - y$.

Helicobacter Pylori infection is considered to be an event that occurs to each individual independently. Modeling such a situation with the binomial distribution is considered to be reasonable; beta distribution, the natural conjugate prior distribution of the binomial distribution, is used as the prior distribution in our protocol mainly due to its mathematical convenience. The initial prior was set to the non-informative uniform distribution in the experiments. Nonetheless, it is difficult to exclude the subjectivity from the settings of the prior distributions, and the obtained experimental results need to be carefully examined.

The mean of the beta distribution is denoted by $E[\theta] = \alpha/(\alpha + \beta)$. We can therefore estimate $\hat{\theta}$ when the BFs of two sets have y matching bits as follows, $\hat{\theta} = \frac{\alpha'}{\alpha'+\beta'} = \frac{1+y}{2+m}$. After estimating $\hat{\theta}$, the size of the intersection is given by the inverse of Equation (4), a mapping θ^{-1}, as

$$\hat{x} = n_A + n_B - \frac{1}{k}\log_{1-\frac{1}{m}}\left(\hat{\theta} - 1 + (1-\frac{1}{m})^{kn_B} + (1-\frac{1}{m})^{kn_A}\right). \quad (6)$$

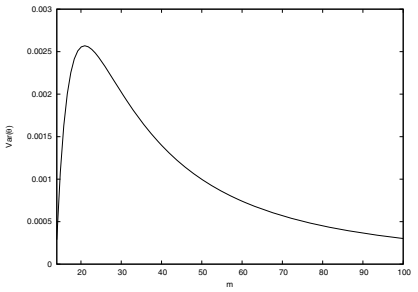

Fig. 1. (1) Distribution of the variance of $\hat{\theta}$, $Var[\theta]$, with respect to m, the size of the BF, for $n = 10, k = 3$, and $y = 14$

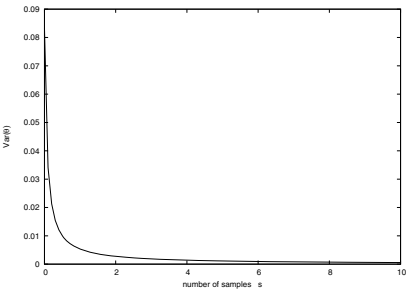

Fig. 2. (2) Distribution of the variance of $\hat{\theta}$, $Var[\theta]$, estimated from s independent BFs of the same size

The inverse mapping can be evaluated locally in the final stage of privacy preservation (without encryption). We are not concerned that if Equation (6) might appear complicated to evaluate.

4.3 "Bootstrap" of BFs

To improve the accuracy, there are two approaches.

(1) Enlarge the size of BF, m, and the estimate $\hat{\theta}$, [3]
(2) Estimate $\hat{\theta}$ from multiple observations of $Y_1, Y_2, \ldots, Y_s$.

Using a BF with more bits m could decrease the false positives in the membership test with the cost increasing as m. It is of interest that the value of m does not play a significant role in estimating of the intersection size, as we had expected. We will now show the mathematical properties that explain this observation.

(1) Variance of the Beta Distribution for a Large BF. According to the known variance of the beta distribution $Var[\theta] = \alpha\beta/((\alpha+\beta)^2(\alpha+\beta+1))$, we illustrate the change of variance with respect to m in Fig. 1. Since the variance determines the standard deviation, which provides a confidence interval for the estimation, we can predict the accuracy via the reduction in variance. Fig. 1 shows that the variance of $\hat{\theta}$ decreases slightly as m increases. However, the reduction in variance is not significant, given the increased cost of the required ciphertexts. For example, a BF with $m = 100$ requires 10 times more ciphertexts than that for an element in S with $n = |S| = 10$.

[3] We do not consider the number of hash functions k because there are some constraints between m and k, such as $kn < m$ and $k = (\ln 2)m/n$ for minimizing false positives.

(2) Variance from "Bootstrap" s Small BFs. Let $y_1, y_2, \ldots, y_s$ be the sequence of matching bits in s independent BFs for S_A and S_B. Recursive Bayesian estimation based on the sequence gives the posterior probability $Pr(\theta|y_1, \ldots, y_s)$ for the beta distribution $Be(\alpha', \beta')$ defined by

$$\alpha' = \alpha + \sum_{i=1}^{s} y_i, \qquad \beta' = \beta - \sum_{i=1}^{s} y_i + sm.$$

The estimation of $\hat{\theta}$ is provided from the mean of the beta distribution, namely

$$\hat{\theta} = \frac{\alpha + \sum_{i=1}^{s} y_i}{\alpha + \beta + sm} \tag{7}$$

Fig. 2 illustrates the reduction in the variance of $\hat{\theta}$. It implies that the bootstrapping reduces the confidence interval for the estimation of θ significantly with increasing s.

4.4 Proposed Scheme

We give the procedure for estimating the size of the intersection without revealing each set in Algorithm 2. At Step 1, both parties A and B compute BFs for their n-element sets S_A and S_B with parameters, size of BF m and the number of hash function k such that $k = (m/n) \ln 2$. For tradeoff between efficiency and accuracy, $k = 1$ and $m = n/\ln 2$ can be used. Since this process can be performed locally and the hash function performs very efficiently, we consider the overhead is negligible. Both parties participate the secure scalar product protocol (Algorithm 1), which is the most significant part in computation. The scalar product of two BFs, y, gives the number of common 1's bit in BFs, which can be divided into two integers, making the SFE possible to approximate $\hat{\theta}$ in Equation (7) without revealing any y_i. Step 4 is performed in public (or locally) after $\hat{\theta}$ reaches at convergence.

Instead of extend the size of BF, we perform the secure scalar product protocols multiple times to get the sequence of $y_1, y_2, \ldots, y_s$, which will be used to predict the $\hat{\theta}$ in Bayesian estimation. Both parties iterate the test until the expected accuracy is given. The confidence interval is given by the standard deviation of estimated value.

4.5 Security

The following theorem shows the security of Algorithm 2.

Theorem 1. *Suppose A and B behaves in the semi-honest model. Let S_A and S_B be inputs for Bloom Filter Bootstrap. Then, after execution of Bloom Filter Bootstrap, A and B learns random shares of y_i for $i = 1, \ldots, s$; nothing but y_i and what can be inferred from y_i is learned by both A and B.*

Algorithm 2. Bloom Filter Bootstrap $BFB(S_A, S_B)$

Input: S_A, S_B of n elements, m (size of the BF), k (number of hash functions).
Output: $\hat{x}$ (estimate of the size of the intersection of s_A and S_B).

1. A computes BF $\boldsymbol{b}(S_A)$ for S_A and B computes BF $\boldsymbol{b}(S_B)$.
2. A and B jointly perform Algorithm 1 to obtain $y_i = \boldsymbol{b}(S_A) \cdot \boldsymbol{b}(S_B)$ for $i = 1, \ldots, s$.
3. Estimate $\hat{\theta}$ using Equation (7).
4. Identify $\hat{x}$ using Equation (6).

Sketch of the Proof. Message exchange occurs only in step 2, so the security of step 2 is proved. Since step 2 is multiple invocation of the scalar product protocol, the security is reduced to that of the scalar product protocol. By following the security proof in [8], the security of Bloom Filter Bootstrap is immediately proved. Note that computation in step 4 is performed by A without communication with B, the security is not compromised by execution of these steps.

4.6 Complexity

We examine the complexities of our proposed scheme in terms of computation and communication costs. When these quantities are almost identical, we unify these by simply n. Protocols are compared in Table 2. In comparison with [11], we assume the ID-less model, where the size of BF can increase up to n^2.

Table 2 shows that the computational cost for A is linear to ms, while the cost for B is 0 (no modular exponentiation is required). Hence, it is preferable for outsourcing solution to our Requirement 3, where hospitals do not have powerful computational resources and become B in our protocol.

The protocols are classified into three groups. The first group is the scheme based on Oblivious Polynomial Evaluation. Scheme FNP[7] is designed to reveal not only the size of intersection but also the elements in the intersection. We show the performance for comparison purpose.

The second class, consisting of AES[1] and CT[5], is classified as Oblivious Pseudo-Random Functions (OPRF). AES depends on the commutative one-way function, while CT uses the RSA (Fig. 3 in [5]) and the blind RSA (Fig. 4) encryptions. The privacy of scheme (Fig. 3 in [5]) is proved as the view of honest-but-curious party is indistinguishable under the One-More Gap Diffie-Hellman assumption in the random oracle model.

Table 2. Complexity Comparison of protocols

	FNP[7]	AES[1]	CT[5]	KNV[11]	Proposed
primitives	OPE	commutative enc.	(blind) RSA	SSP w. BF	SSP w. BF
comp. at A	$n_A \log\log n_B$	$n_A + n_B$	$2n_A + 1$	m	ms
BF size	–	–	–	$n^2 \geq m > kn$	$m = n/\ln 2$
comp. at B	$n_B + n_A \log\log n_B$	$2n_A + n_B$	$n_A + n_B + 1$	0	0
complexity	$O(n_A \log\log n_B)$	$O(n)$	$O(n)$	$O(n^2)$	$O(n)$
comm. cost	$n_A + n_B$	$n_A + n_B$	$2n_A + n_B$	$m + 1$	$ms + 1$

OPE (Oblivious Polynomial Evaluation), SSP (Secure Scalar Product).

Table 3. Results of estimating X for various intersection sizes, x, for the dataset ($n_A = n_B = 100, m = 400, k = 3$)

x	20	40	60	80
$E[Y]$	125.24	141.45	160.98	184.11
$\sigma(Y)$	6.78	5.92	5.34	5.15
$E(\theta)$	0.31	0.35	0.40	0.46
$\hat{x}$	19.523	38.869	58.969	79.411

Table 4. Results of estimating X for various BF sizes, m for the dataset ($n_A = n_B = 100, x = 40$)

m	200	400	600	800
k	1	3	4	6
$E[Y]$	46.62	141.45	189.64	283.66
$\sigma(Y)$	3.146	5.923	6.436	7.488
$E(\theta)$	0.24	0.35	0.32	0.35
$\hat{x}$	39.490	38.869	39.604	39.227

The last class is based on BF and Secure Scalar Product schemes. KNV[11] uses a single BF with large size, while ours iterates s independent BFs with small size. The sizes are shown in Table.

5 Accuracy Evaluation

5.1 Simulation with DBLP Dataset

We evaluate the accuracy of the proposed scheme using a public dataset of author names, DBLP[4].

Four pairs of datasets S_A and S_B with $n_A = n_B = 100$ were chosen from DBLP with the intersection sizes $x = 20, 40, 60, 80$. Table 3 shows the experimental results for the estimation of x, for $x = 20, 40, 60$, and 80, where we used a BF with of size $m = 400$, a number of hash functions $k = 3$, and iterated the estimation s times. The results show that our scheme estimates the intersection within an error of ± 1. The numbers of matching bits in the BFs, Y, are distributed according to the binominal distribution. Note that all BFs estimate a size of the intersection close to the actual size of 40, but the differences are unstable.

5.2 Optimal BF Design

The accuracy of estimation depends on the size of BF, m, and the number of hash function, k, and the iteration of testing, s. In order to clarify the strategy for optimal accuracy, we examine the Mean Absolute Error (MAE) with respect to m and k. Figure 3 shows MAE in terms of m from 40 through 280, where $n_A = n_B = 100$, $x = 20$, $k = 1$ and $s = 20$. Figure 4 shows MAE with respect to $k = 1, \ldots, 6$ where $m = 200$. The MAE decreases as m increases, while the computational/communicational overhead increases accordingly. On the other hand, the increase of k does not reduce MAE.

A possible reason for the source of the error might be the restriction of m and k. As we discussed in Section 4.3, the optimal size for the BF is not trivial. We

[4] DBLP, A Citation Network Dataset, V1, (`http://arnetminer.org/citation`).

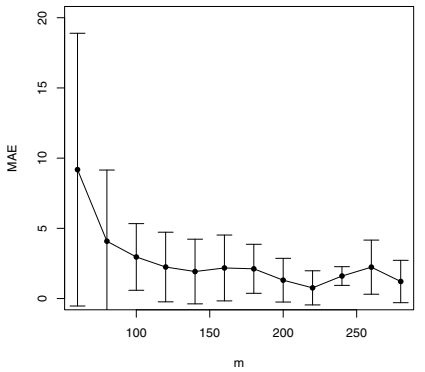

Fig. 3. Mean Absolute Error (MAE) with respect to the size of BF, m

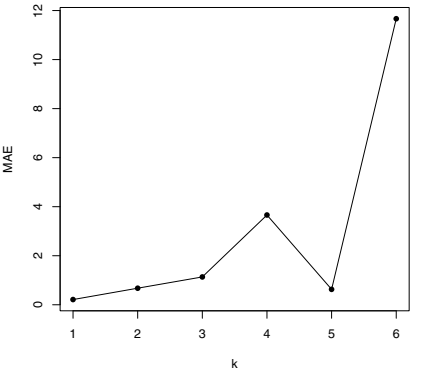

Fig. 4. Mean Absolute Error (MAE) with respect to the number of hash functions, k

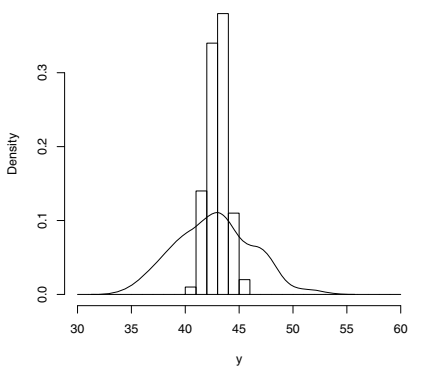

Fig. 5. Distributions of matching bits in the BFs, y, drown in solid line, and distribution of the means, $E[y]$, in bars

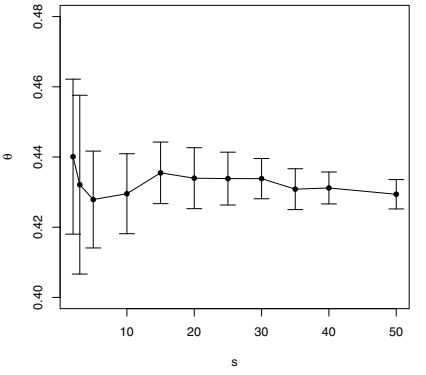

Fig. 6. Conversion of probability of 1s bit in BF, θ, with respect to iteration of BFs, s

therefore suggest choosing $k = 1$ first and then determining a near-optimal BF size by

$$m = kn/\ln 2 = 1 \cdot 100/\ln 2 = 144.26.$$

Since large m increases the computational cost at secure scalar product, we conclude minimize k, i.e., $k = 1$ and optimize $m = n/\ln 2$.

The accuracy can be improved by iteration of small BF tests rather than increasing the size of BFs. In fact, Figure 5 demonstrates the reduction of variance of observation of $E[Y]$, indicated by bar plot, when $s = 10$. The solid line represents the distribution of Y, which is widely distributed than that of $E[Y]$. It is known as Central Limit Theorem[14], that as s increases, the amount of sampling variation decreases. Figure 6 shows that the variance of estimated probability $\hat{\theta}$

reduces as the iteration s increases. The experiment shows even small $s = 10$ gives conversion of probability θ. The selection of optimal s can be made based on the variance of the prediction of θ. As we have showed in Section 4.3, the variance of beta distribution decreases with s, which determines the accuracy of approximation.

Finally, we obtain the estimate of intersection size, $\hat{x}$, by Equation 6. We illustrate the distribution of θ and the corresponding estimation of x.

5.3 Performance

We implemented the proposed scheme in Java, JDK 1.6, with BigInteger class. As additive homomorphic public key algorithm, we use Paillier cryptosystem with 1024 bit key. With platform of commodity PC, Intel Core (TM) i7-663DQM, 2 GHz, 4 GB, running Windows 7 (64 bit), the encryption runs in $t_e = 15.7$ [s], the decryption takes $t_d = 21.5$ [s] in average. The secure scalar product of 64-bit vectors ($n_A = n_B = 64, x = 5$) is performed in 5.28 [s], i.e., 82.5 [ms/element]. With this platform, the processing time to deal with the problem in [11], $n = 2000$, $k = 1$, and $m = n/\ln 2 = 2885$, is 4 minute and 125 second.

6 Privacy-Preserving Risk Analysis of *H. pylori*

Helicobacter pylori, or H. pylori, is a bacterium that is found in the stomachs of two-thirds of the world's population. Epidemiology studies have shown that individuals infected with *H. pylori* have an increased risk of cancer of the stomach[10,12].

Although *H. pylori* has been classified as a cancer-causing agent, it is not known how *H. pylori* infection increases the risk of cancer of the stomach. Some researchers have estimated that the risk of cancer the noncardiac region of the stomach is nearly six times higher for *H. pylori*–infected individuals than for uninfected people[9]. Some cohort studies revealed that the risk of gastric cardiac cancer among *H. pylori*–infected individuals was about one-third of that among uninfected individuals. The source of uncertainty is that the number of gastric cancers in the cohort study was too small to make a definitive statement. Cancer is a highly confidential matter and people will not reveal that they have it.

Our proposed methodology addresses the problem of epidemiology studies that preserve the privacy of the patients. The cryptographic protocol allows several small cohorts to be aggregated and analyzed for more certain evidence of increase or reduction of risk. Given two datasets of patients with cancer and *H. pylori*, the proposed protocol determines the size of the intersection of the two sets without revealing any entries in the datasets. With a secure hash function, the proposed scheme identifies a patient from their personal attributes.

6.1 Contingency Tables

The epidemiology study aims to determine whether an *H. pylori*-infected individual has increased the risk of gastric cancer. The evidence is shown by a measure of *relative risk* (RR), the probability of disease among exposed individuals divided by the probability of disease among the unexposed. Suppose that a sample of N individuals is arranged in the form of the 2×2 contingency table in Table 5; the relative risk (RR) of *H. pylori* is estimated by

$$RR = \frac{\Pr(\text{ cancer } | \text{ H. pylori})}{\Pr(\text{cancer } | \text{unexposed})} = \frac{a}{a+b} \Big/ \frac{c}{c+d} \approx \frac{ad}{bc},$$

where we assume $a \ll b$ and hence $a + b = b$.

To examine whether *H. pylori*-infection increases the risk of cancer, i.e., $RR > 1$, we test the null and the alternative hypotheses.

H_0**:** The proportion of patients with cancer among individuals infected with *H. pylori* is equal to the proportion of patients with cancer among those uninfected.

H_A**:** The proportions of patients with cancer are not identical in the two populations.

The chi-square test compares the observed frequencies in each category of the contingency table, O, with the expected frequencies given that the null hypothesis is true, E. To perform the test, we calculate the sum

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i} = \frac{(N-1)\left((ad-bc) \pm N/2\right)^2}{(a+c)(b+d)(a+b)(c+d)},$$

where k is the number of cells in the table. The probability distribution of this sum is approximated by a χ^2 distribution with $(2-1)(2-1) = 1$ degree of freedom. Alternatively, by taking it squire root, we may assume that χ is normally distributed with mean 0 and standard deviation 1.

Table 5. 2×2 Contingency table for *H. pylori* and stomach cancer

H. pylori	Cancer	No cancer	total
Yes	a	b	$a+b$
No	c	d	$c+d$
total	$a+c$	$b+d$	N

Table 6. Chiba Cancer Center dataset `CAN`

year	male	female	total
2003	2,330	1,134	3,464
2004	2,610	1,242	3,852
2005	2,559	1,205	3,763
total	7,500	3,581	11,081

Table 7. MHW dataset of *H. pylori* infections `PYL`

year	male	female	total
2001	2,671	5,206	7,877

6.2 Datasets

In our experiment, we have two datasets collected by independent agencies.

1. Patients with gastric cancer `CAN`.
   The Chiba Cancer Center has performed an epidemiology study of causes and effects of cancer conditions since 1975 in Chiba Prefecture, Japan. Table 6 shows the statistics for three years from 2003, used in this study. The dataset contains private attributes, including name, gender, birthday, mailing address, ZIP code, and medical treatments, e.g., patient ID, days of operations, day of death, type of cancers, and degree of tumor differentiation.
2. Individuals infected with *H. pylori* `PYL`.
   The Japanese Ministry of Health and Welfare (MHW) carried out a medical examination in 2001 in a small village in Chiba Prefecture. The dataset contains the number of *H. pylori*-infected individuals but their cancer status is not known.

6.3 Hypothesis Testing

Our proposed algorithm estimates the size of the intersection of the two datasets, thus allowing the estimation of relative risk of *H. pylori*.

The statistics show that the population in Chiba Prefecture in 2003 was $6,056,462$ ($3,029,486$ male). The dataset in Table 6 has $n_A = 7401$ recodes of patients with cancer. Table 7 contains $n_B = 2629$ individuals infected with *H. pylori*. We apply a BF with size $m = 14,000$, $k = 1$ and $s = 10$ to the two datasets and obtain the scalar product, $y = \boldsymbol{b}(\texttt{CAN}) \cdot \boldsymbol{b}(\texttt{PYL})$ as $\mu(y) = 1023.9$ on average. Based on Bayes' theorem, we estimate the probability $\hat{\theta}$ in Equation (7) as

$$\hat{\theta} = \frac{\alpha + \sum^{s} y_i}{\alpha + \beta + sm} = 0.073142.$$

From Equation (6), $\hat{x} = 81.1702$, while the exact size of the intersection is 80. The number of individuals who are infected with *H. pylori* but do not have is therefore $n_a - \hat{x} = 2549$. The other values can be obtained similarly. Finally, the numbers of individuals are summarized in Table 8.

Table 8. Experimental results for `CAN` and `PYL`

H. pylori	Cancer	No cancer	total
Yes	80	2,549	2,629
No	7,321	2,990,050	2,997,371
total	7,401	2,992,599	3,000,000[5]

[5] The number is referred from statistics in Chiba prefecture. There are potential individuals infected by H. Pylori who was not counted in the table.

An estimate of the relative risk of having cancer among *H. pylori*-infected individuals is therefore

$$RR = \frac{80 \cdot 222,964}{2,549 \cdot 7,321} = 12.81.$$

The chi-square test of the null hypothesis yields

$$\begin{aligned}\chi &= \frac{\sqrt{3,000,000-1}(80 \cdot 222,964 - 2,549 \cdot 7,321 - 3,000,000/2)}{\sqrt{7,401 \cdot 2,992,599 \cdot 2,629 \cdot 230,285}} \\ &= 28.71 > N(.05/2) = 1.960,\end{aligned}$$

which is too high to assume the null hypothesis. Therefore, we reject the null hypothesis at the 0.05 level of confidence.

In the experiment in Intel Xeon E5620 2.40 GHz, Memory 16GB, the processing of the BF takes 17,030 second (=4.7 hour), while the naive ID generation requires a scalar product of $n^2 = 4.9 \times 10^7$, which is estimated to be 223 hours.

7 Conclusions

We have proposed an efficient algorithm for the estimation of the size of the intersection of two private sets. The proposed scheme gives a Bayesian estimation of the intersection size based on the mathematical properties of the number of matching bits in two BFs. A well-known secure scalar product protocol enables us to evaluate the number of matching bits in a privacy-preserving way and to test hypothesizes that are useful in epidemiological studies. We have shown the properties of the accuracy of estimation for various parameters and the experimental results for the DBLP public dataset. One of our main results is that the bootstrap approach, iterating small BFs several times, is better than using a single large BF.

The extension of scalar product protocol to multiple parties can be done by replacing the Step 3 as that Bob forwards n ciphertexts computed with his secret vector as $E(x_1)^{y_1}, \ldots, E(x_n)^{y_n}$ to Carol who then perform the original Step 3 as $c = E(x_1)^{y_1 z_1} \cdots E(x_n)^{y_n z_n} / E(s_B)$. The extension of Bloom filter to multiple parities is not trivial and one of our future work.

References

1. Agrawal, R., Evfimievski, A., Srikant, R.: Information sharing across private databases. In: Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, pp. 86–97. ACM Press (2003)
2. Broder, A., Mitzenmacher, M.: Network applications of bloom filters: A survey. In: Internet Mathematics, pp. 636–646 (2002)
3. Camenisch, J., Zaverucha, G.M.: Private intersection of certified sets. In: Dingledine, R., Golle, P. (eds.) FC 2009. LNCS, vol. 5628, pp. 108–127. Springer, Heidelberg (2009)

4. Clifton, C., Kantarcioglu, M., Vaidya, J., Lin, X., Zhu, M.Y.: Tools for privacy preserving distributed data mining. ACM SIGKDD Explorations Newsletter 4(2), 28–34 (2002)
5. De Cristofaro, E., Tsudik, G.: Practical private set intersection protocols with linear complexity. In: Sion, R. (ed.) FC 2010. LNCS, vol. 6052, pp. 143–159. Springer, Heidelberg (2010)
6. Fan, L., Cao, P., Almeida, J., Broder, A.Z.: Summary cache: a scalable wide-area web cache sharing protocol. IEEE/ACM Trans. Netw. 8(3), 281–293 (2000)
7. Freedman, M.J., Nissim, K., Pinkas, B.: Efficient private matching and set intersection. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 1–19. Springer, Heidelberg (2004)
8. Goethals, B., Laur, S., Lipmaa, H., Mielikäinen, T.: On private scalar product computation for privacy-preserving data mining. In: Park, C.-s., Chee, S. (eds.) ICISC 2004. LNCS, vol. 3506, pp. 104–120. Springer, Heidelberg (2005)
9. Helicobacter and Cancer Collaborative Group. Gastric cancer and helicobacter pylori: a combined analysis of 12 case control studies nested within prospective cohorts. Gut. 49(3), 347–353 (2001)
10. Atherton, J.C.: The pathogenesis of helicobacter pylori-induced gastro-duodenal diseases. Review of Pathology 1, 63–96 (2006)
11. Kantarcioglu, M., Nix, R., Vaidya, J.: An efficient approximate protocol for privacy-preserving association rule mining. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS, vol. 5476, pp. 515–524. Springer, Heidelberg (2009)
12. Kuipers, E.J., Kusters, J.G., van Vliet, A.H.: Pathogenesis of helicobacter pylori infection. Clinical Microbiology Reviews 19(3), 449–490 (2006)
13. Lu, H., He, X., Vaidya, J., Adam, N.R.: Secure construction of contingency tables from distributed data. In: Atluri, V. (ed.) DAS 2008. LNCS, vol. 5094, pp. 144–157. Springer, Heidelberg (2008)
14. Pagano, M., Gauvreau, K., Pagano, M.: Principles of biostatistics. Brooks/Cole (2000)
15. Ravikumar, P., Ravikumar, P., Fienberg, S.E., Cohen, W.W.: A secure protocol for computing string distance metrics. In: PSDM (2004)
16. Sakuma, J., Wright, R.N.: Privacy-preserving evaluation of generalization error and its application to model and attribute selection. In: Zhou, Z.-H., Washio, T. (eds.) ACML 2009. LNCS, vol. 5828, pp. 338–353. Springer, Heidelberg (2009)
17. Vaidya, J., Clifton, C.: Privacy preserving association rule mining in vertically partitioned data. In: The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 639–644 (2002)