An Overview on Privacy Preserving Data Mining Methodologies

Umesh Kumar Singh, Bhupendra Kumar Pandya , Keerti Dixit
Institute of Computer Science
Vikram University
Ujjain, India

Abstract— Recent interest in the collection and monitoring of data using data mining technology for the purpose of security and business-related applications has raised serious concerns about privacy issues. For example, mining health care data for the detection of disease outbreaks may require analyzing clinical records and pharmacy transaction data of many individuals over a certain area. However, releasing and gathering such diverse information belonging to different parties may violate privacy laws and eventually be a threat to civil liberties. Privacy preserving data mining strives to provide a solution to this dilemma. It aims to allow useful data patterns to be discovered without compromising privacy. This paper presents an brief overview on preserving data mining methodologies.

Keywords: Privacy Preserving Data Mining

I. INTRODUCTION

Privacy preserving data mining in a broad sense has been an area of research since 1991 [1] both in the public and private [2] sector and has also been discussed at numerous workshops and international conferences [3]. Recent interest in the collection and monitoring of data using data mining technology for the purpose of security and business-related applications has raised serious concerns about privacy issues. Sometimes, individuals or organizational entities may not be willing to disclose the sensitive raw data; sometimes the knowledge and/or patterns detected by a data mining system may be used in a counter-productive manner that violates the privacy policy. The main objective of privacy preserving data mining is to develop algorithms for modifying the original data or modifying the computation protocols in some way, so that during and after the mining process, the private data and private knowledge remain private while other underlying data patterns or models can still be effectively identified.

.

II. LITERATURE ON PRIVACY PRESERVING DATA MINING SELECTING A TEMPLATE

DATA HIDING:

The main objective of data hiding is to transform the data so that the private data remains private during and/or after data mining operations.

Data Perturbation:

Data perturbation techniques can be grouped into two main categories, which we call the value distortion technique and probability distribution technique. The value distortion technique perturbs data elements or attributes directly by either some other randomization procedures. On the other hand, the probability distribution technique considers the private database to be a sample from a given population that has a given probability distribution. In this case, the perturbation replaces the original database by another sample from the same [estimated] distribution or by the distribution itself.

Note that there has been expensive research in the area of statistical databases [SDB] on how to provide summary statistical information without disclosing individual's confidential data. The privacy issues arise when the summary statistics are derived from data of very few individuals. A popular disclosure control method is data perturbation, which alters individual data in a way such that the summary statistics remain approximately the same. However, problems in data mining become somewhat different from those in SDBs. Data mining techniques, such as clustering, classification, prediction and association rule mining are essentially relying on more sophisticated relationships among data records or data attributes, but not just simple summary statistics. This research work specifically focuses on data perturbation for privacy preserving data mining. In the following, we will primarily discuss different perturbation techniques in the data mining area. Some important perturbation approaches in SDBs are also covered for the sake of completeness.

ADDITIVE PERTURBATION:

The work in proposed an additive data perturbation technique for building decision tree classifiers. In this technique, each client has a numerical attribute x_i and the server [or data miner] wants to learn the distribution of these attributes o build a classification model. The clients randomize their attributes x_{ii} by adding random noise r_i drawn independently from a known distribution such as a uniform distribution or Gaussian distribution. The server [or data miner] collects the values of $x_{ii} + r_i$ and reconstructs x_i's distribution using a version of the Expectation-Maximization [EM] algorithm.

This algorithm probably converges to the maximum likelihood estimate of the desire original distribution. Kargupta et al [4,5,6], later questioned the use of random additive noise and pointed out that additive noise can be easily filtered out in many cases that will possibly compromise the privacy. To be more specific, they proposed a random matrix-based Spectral Filtering [SF] technique to recover the original data from the perturbed data. Their empirical results have shown that the recovered data can be reasonably close to the original data.

However, two important questions remain to be answered: (1) What are the theoretical lower bound and upper bound of the reconstruction error; and (2) What are the key factors that influence the accuracy of the data reconstruction. Guo and Wu [7] investigated the Spectral Filtering technique and derived an upper bound for the Frobenius norm of the reconstruction error using matrix perturbation theory. They also proposed a Singular Value Decomposition (SVD)-based reconstruction method and derive a lower bound for reconstruction error. They then proved the equivalence between the SF and SVD approach, and as a result, the lower bound of SVD approach can also be considered as the lower bound of the SF approach. Huang et al. [8] pointed out that the key factor that decides the accuracy of data reconstruction is the correlation among the data attributes. Their results have shown that when the correlations are high, the original data can be reconstructed more accurately, that is, more private information can be disclosed. They further proposed two data reconstruction methods based on data correlations: one used the principal Component Analysis [PCA], and the other used the Bayes Estimate [BE] technique, which in essence processing literature on filtering random additive noise, the utility of random additive noise for privacy preserving data mining is not quite clear.

DATA MICRO-AGGREGATION:

Data Micro-aggregation is a popular data perturbation approach in the area of secure statistical databases [SDBs]. For a dataset with a single private attribute, univariate micro-aggregation sorts data records by the private attribute, group's adjacent records into groups of small sizes, and replaces the individuals private values in each group with the group average. Multivariate micro-aggregation considers all the attributes and groups data using a clustering technique. This approach primarily considers the preservation of data covariance instead of the pair wise distance among data records.

Recently, two multivarity micro-aggregation approaches have been proposed by researchers in the data mining area. Agarwal and Yu [9] presented a condensation approach to privacy preserving data mining. This approach first partitions the original data into multiple groups of predefined size. For each group, a certain level of statistical information [e.g., mean and covariance] about different data records is maintained. This statistical information is used to create anonymized data that has similar statistical characteristics to the original datset, and only the anonymized data is released for data mining applications. This approach preserves data covariance instead of the pair-wise distance among data records. Proposed a kd-tree based perturbation method, which recursively partitions a dataset into smaller subset such that data records in each subset are more homogeneous after each partition. The private data in each subset are then perturbed using the subset average. The relationships between attributes are expected to be preserved.

DATA ANONYMIZATION:

Sweeny developed the k-anonymity framework [10] [11] wherein the original data is transformed so that the information for any individuals can not be distinguished from [k-1] others. Generally speaking, anonymization is achieved by suppressing [deleting] individual values from data records [e.g., the zip codes 21250-21259 might be replaced with 2125*]. A variety of refinements of this framework have been proposed since its initial appearance. Some of the work start from the original dataset and systematically or greedily generalize it into one that is k-anonymous. Some start with a fully generalized dataset and systematically specialize the dataset into one that is minimally k-anonymous. The problem of k-anonymous is not simply to find any k-anonymous, but to, instead, finds one that is "good" or even "best" according to some quantifiable cost metric. Each of the previous work provides its own unique cost metrics for modeling desirable anonymization.

Recently, Machanavajjhala [12] & [13] pointed that simple k-anonymity is vulnerable to strong attacks due to the lack of diversity in the sensitive attributes. They proposed a new privacy definition called l-diversity. The main idea behind l-diversity is the requirement that the values pf the sensitive attributes are well represented in each group. Other enhanced k-anonymity models have been proposed elsewhere.

Data Swapping technique transforms the database by switching a subset of attributes record entries are unmatched, but the statistics [e.g., marginal distributions of individuals attributes] are maintained across the individual fields. This technique was first proposed by Dalenius and Reiss. A variety of refinements and applications of data swapping have been addressed since its initial appearance.

SECURE MULTI-PARTY COMPUTATION [SMC]:

Secure Multi-Party Computation [SMC] considers the problem of evaluating a function of two or more parties' secret inputs, such that no party learns anything but the designated output of the function. Concretely, we assume we have inputs x_1, …,x_n, where party i owns xii, and we want to compute function f[x1,…,xn] = [y1,…,yn] such that party I gets yi and nothing more than that.

Example:- As an example we may consider Yao's millionaire's problem: two millionaires meet in the street and want to find out who is richer without having to reveal their actual fortune to each other. The function computed in this case is a simple comparison between two numbers. If the result is that the first millionaire is richer, then he knows that, but this should be all information he learns about the other guy.

ADVERSARIAL BEHAVIOUR:

It is common to model cheating by considering adversarial parties that attempt to obtain information about the private inputs of their peers. SMC typically studies two types of adversaries: A semi-honest adversary [also known as passive, or honest but curious adversary] is a party who follows the protocol properly, yet attempts to learn additional information by analyzing all the intermediate results and the messages received during the protocol execution. On the other hand, a malicious adversary may arbitrarily deviate from the protocol specification. A malicious adversary could refuse to participate in the protocol when the protocol is first invoked, could substitute its input and enter the protocol with an input other than the one provided with it, and could abort the protocol prematurely. It is obviously easier to design a solution that is secured against semi-honest adversaries than it is to design a solution for malicious adversaries. In practice, people usually first design a secure protocol for the semi-honest scenario, and then transform it to a protocol that is secure against malicious adversaries. This transformation can be done by requiring each party to use zero-knowledge proofs to prove that each step it is taking follows the protocol specification.

PRIVACY:

Generally speaking, an SMC protocol privately computes a function if any information that a party can obtain can be essentially obtained by that party through its own inputs and outputs. An alternative definition compares the results of actual computation to that of an ideal computation. Here the ideal computation assumes there exists a trusted party who does not deviate from the protocol specification at all, and does not attempt to cheat. All parties send their private inputs to the trusted party, who computes the function and sends the appropriate results back to all the parties. We say a protocol is secure or private if anything that an adversary can learn in the actual world can also be learned in the ideal world, namely from its own inputs and from the outputs it receives from the trusted party. In essence, protocols satisfying this definition prevent an adversary from gaining any extra advantage in the actual world over what it could have gained in an ideal world.

III. BUILDING BLOCKS:

We describe here some representative building blocks of secure multi-party computation.

Oblivious Transfer: In cryptography, an oblivious transfer protocol is a protocol by which a sender sends some information to the receiver, but remains oblivious as to what is sent. Oblivious transfer is one of the most important protocol for secure computation. It has been shown by kilian that oblivious transfer is sufficient for secure computation in the sense that given an implementation of oblivious transfer it is possible to securely evaluate any polynomial time computable function without any additional primitive.

Homomorphic Encryption: A public-key cryptosystem p [G,E,D] is a collection of probabilistic polynomial time algorithms for key generation, encryption and decryption. The key generation algorithm G produces a private key sk and public key pk with specified key size. Anybody can encrypt a message with the public key, but only the holder of a private key can actually decrypt the message and read it. The encryption algorithm E takes as an input a plaintext m, a random value r and a public key pk and outputs the corresponding cipher-text Epk[m,r]. The decryption algorithm D takes as an input a cipher text c and a private key sk[corresponding to the public key pk] and outputs a plaintext Dsk[c]. It is required that Dsk[Epk[m,r]]=m.

The plaintext is usually assumed to be from Zμ, where μ is the product of two large primes. A public-key cryptosystem is homomorphic when

¥ m1, m2, r1, r2 € μ

Dsk [Epk [m1, r1] Epk [m2, r2] mod μ2] = m1 + m2 mod μ;

Dsk [Epk [m1, r1] m2] mod μ2 = m1 m2 mod μ;

Dsk [Epk [m2, r2] m1] mod μ2 = m1 m2 mod μ;

This features allows a party to add or multiply plaintext by doing simple computations with cipher texts, without having the secret key. Several homomorphism cryptosystems in the literature are proved to be secure under reasonable complexity assumptions.

A natural application of homomorphism encryption is private inner product computation. It considers the problem of computing the inner product of two vectors owned by two different parties [Alice and Bob for example], respectively, so that neither party should learn anything beyond what is implied by the party's own vector and the output of the computation. Here the output of the party is either the inner product or nothing, depending on what the party is supposed to learn. It is directly based on homomorphism encryption and has been proved to be private in a strong sense. To be more specific, no probabilistic polynomial time algorithm substituting one party can obtain a non-negligible amount of

information about the other party's private input, except what can be deduced from the input and output of this party.

Commutative Encryption: Simply speaking, a commutative encryption is a pair of encryption function f and g such that f[g[x]]=g[f[x]].

Definition [Commutative Encryption] A commutative encryption F is a computable polynomial time function f : key F → Dom f, defined on finite computable domains, and satisfying all properties listed below. We denote fe[x]≡f[e,x], and use "€r" to mean "is chosen uniformly at random from."

Commutative : For all e, e' € key F, we have fe o fe' = fe' o fe

Each fe : Dom F → Dom F is a bisection.

The inverse fe-1 is also computable in polynomial time given e.

The distribution of < x, fe[x], y, fe[y] > is computationally indistinguishable from the distribution < x, fe[x], y, z >, where x, y, z € r Dom F and e € r key F.

Property 1: says that the composition of the encryption with two different keys is the same irrespective of the order of encryption.

Property 2: says that two different values will never have the same encrypted value.

Property 3: says that given an encrypted value fe[x] and the encryption key e, we can find x in polynomial time.

Property 4: says that given a value x and its encryption fe[x] [but not the key e] and a new value y, we cannot distinguish between fe[y] and a random value z in polynomial time.

Thus we cannot encrypt y or decrypt fe[y] in polynomial time.

As an example, let Dom F be all quadratic residues modulo p, where p is a safe prime number, i.e., both p and q = [p-1]/2 are primes. Let key F be {1,2,…,q1}.

Then assuming the Decisional Diffie-Hellman hypodissertation [DDH], the power function

fe[x]≡xe mod p

is a commutative encryption because

fe[fd [x]] = [xd mod p] e mod p = xde mod p = [xe mod p] d mod p = fd[fe [x]].

Based on commutative encryption, Agrawal et al. developed several secure protocols for set intersection, equijoin, intersection size, and equijoin size.

IV. Rule Hiding:

The main objective of rule hiding is to transform the database such that the sensitive rules, for example, associate rules and classification rules, are masked, and all the other underlying patterns can still be discovered.

Association Rule Hiding:

Association Rule Hiding considers the problem of transforming the database so that all the sensitive association rules are concealed and other non-sensitive rules can still be identified. For example, the perturbation-based association rule hiding techniques are implemented by changing a selected set of 1-values to 0-values [in a binary database] or vice-versa so that the frequent item sets that generate the sensitive rules are hidden or the support of sensitive rules is lowered to a user-specified threshold. The blocking-based association rule hiding approach replaces certain attributes of the data with a question mark. The introduction of this new special value in the dataset imposes some changes on the definition of the support and confidence of an association rule. In this regard, the minimum support and minimum confidence will be changed into a minimum support interval and a minimum confidence interval. As long as the support and/or the confidence of a sensitive rule lies below the middle in these two ranges, the confidentially of data is expected to be protected.

Classification Rule Hiding:

The work in presented in framework that combines decision tree classification and parsimonious downgrading. Here the term "parsimonious downgrading" refers to the phenomenon of trimming out sensitive information from a dataset when it is transferred from a secure environment [referred to as high] to a public domain [referred to as low]. The objective of this work is to guarantee that the receiver of the data will be unable to build informative classification models for the data that is not downgraded.

V. Conclusion:

Data mining technologies have enabled commercial and governmental organizations to extract useful knowledge from data for the purpose of business and security related applications. While successful applications are encouraging, there are increasing concerns about the invasions to the privacy of personal information. To address these concerns, researchers in the data mining community have proposed various solutions. This paper presents an overview of them. It has noted that the main consideration in privacy preserving data mining is two fold: 1]data hiding: sensitive raw data should be modified or trimmed out from the original database while the important underlying patterns of the data should still be preserved; and 2]rule hiding: sensitive knowledge which can be discovered from the data should be filtered out.

References

[1] See overview of articles by S. Oliveira at http://www.cs.ualberta.ca/%7Eoliveira/psdm/pub_by_year.html or k. Liu at http://www.cs.umbc.edu/~kunliul/research/privacy_review.html

[2] For example research carried out by IBM, see http://www.almaden.ibm.com/software/disciplines/iis/

[3] See for example overview up to 2004 at http://www.cs.ualberta.ca/%7Eoliveira/psdm/workshop.html

[4] H.Kargupta, S.Datta, Q.Wang, and K.sivakumar,"On the privacy preserving properties of random data perturbation techniques," in proceedings of the IEEE International conference on Data Mining, November 2003.

[5] K.Liu, H.Kargupta, and J.Ryan, "Random projection-based multiplicative data perturbation for privacy preserving distributed data mining,"IEEE Transaction on knowledge and Data Engineering [TKDE], vol.18, no.1, January 2006.

[6] K.liu, C. Giannella, and H. Kargupta,"An attacker's view of distance preserving maps
For privacy preserving distributed data mining," in proceeding of the10th European conference on principles and practice of knowledge Discovery in Databases [PKDD'06], Berlin, Germany, September 2006.

[7] S.Guo and X.Wu,"On the use of spectral filtering for privacy preserving data mining,"in proceedings of the 21st ACM Symposium on Applied computing, Dijon, France, April 2006.

[8] Z. Huang,W.Du, and B. Chen, " Deriving private information from randomized data," in proceeding of the 2005 ACM SIGMOD conference, Baltimroe, MD, June 2005.

[9] C.C Agarwal and P.S. Yu," A condensation based approach to privacy preserving data mining," in proceeding of the 9th International conference on Extending Database Technology [EDBT'04, March 2004.

[10] R.J. Bayardo and R. Agrawal, "Data privacy through optimal k- anonymization," in proceeding of the 21st International conference on Data Engineering [ICDE'05].

[11] R.Chi-wing, J.Li, A. W._C. Fu, and K.Wang, "[α,k]-anonymity: an enhanced k-anonymity model for privacy preserving data publishing," in proceedings of the 12th ACM SIGKDD International conference on knowledge Discovery and Data Mining [SIGKDD'06].

[12] A. Machanavajjhala, J.Gehrke, D. Kifer, and M.Venkitasubramaniam," I-diversity: privacy beyond K-anonymity," in proceeding of the 22nd International Conference on Data Engineering [ICDE'06].
[13] N. Li and T. Li, "t-closeness: privacy beyond K-anonymity and L-diversity," in proceeding of the 23rd International Conference on Data Engineering [ICDE'07].