CONTENTS

Page No.

List of Tables

List of Figures

INTRODUCTION

1.1 Introduction

Data mining has attracted more and more attention in recent years, probably because of the popularity of the “big data” concept. Data mining is the process of discovering interesting patterns and knowledge from large amounts of data.

Security and Privacy protection have been a public policy concern for decades. However, rapid technological changes, the rapid growth of the internet and electronic commerce, and the development of more sophisticated methods of collecting, analyzing, and using personal information have made privacy a major public and government issues. The field of data mining is gaining significance recognition to the availability of large amounts of data, easily collected and stored via computer systems. Recently, the large amount of data, gathered from various channels, contains much personal information. When personal and sensitive data are published and/or analyzed, one important question to take into account is whether the analysis violates the privacy of individuals whose data is referred to. The importance of information that can be used to increase revenue cuts costs or both.

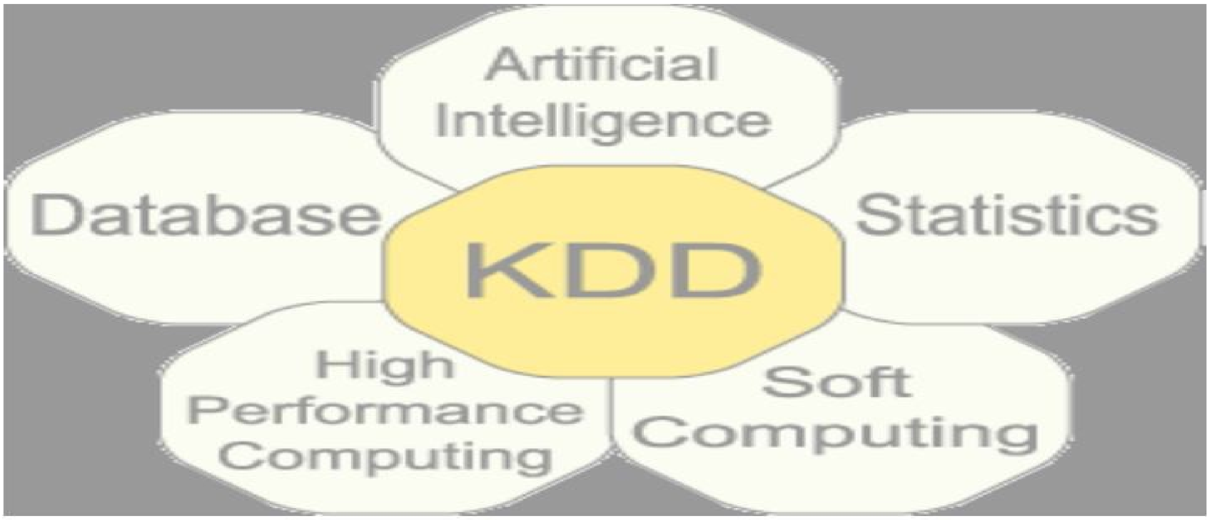


Figure-1: KDD Model

Data mining, popularly known as Knowledge Discovery in Databases (KDD), it is the nontrivial extraction of implicit, previously unknown and potentially useful information from data in databases. Knowledge discovery is needed to make sense and use of data. Though, data mining and knowledge discovery in databases are frequently treated as synonyms, data mining is actually part of the knowledge discovery process. Data mining involves the use of sophisticated data analysis tools to discover previously unknown, valid patterns and relationships in large data sets. These tools can include statistical models, mathematical algorithms, and machine learning methods. Consequently, data mining consists of more than collecting, organizing and managing data; it also includes analysis and prediction.

1.1.1 What is Data Mining?

Data mining is an iterative and interactive process of discovering something innovative. Data mining is the process of discovering meaningful new correlations, patterns and trends by sifting through large amounts of data stored in repositories, using pattern recognition technologies as well as statistical and mathematical techniques." There are other definitions: Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner".

DATA
DATA MINING
PATTERNS

Figure-2: Data mining

1.1.2 The Process of KDD

KDD refers to "Knowledge Discover from Data". To obtain useful Knowledge from data, the following steps are performed in an iterative way.

- Step 1: Data preprocessing. Basic operations include data selection (to retrieve data relevant to the KDD task from the database), data cleaning (to remove noise and inconsistent data, to handle the missing data fields, etc.) and data integration (to combine data from multiple sources).

- Step 2: Data transformation. The goal is to transform data into forms appropriate for the mining task, that is, to find useful features to represent the data. Feature selection and feature transformation are basic operations.

- Step 3: Data mining. This is an essential process where intelligent methods are employed to extract data patterns (e.g. association rules, clusters, classification rules, etc).

- Step 4: Pattern evaluation and presentation. Basic operations include identifying the truly interesting patterns which represent knowledge, and presenting the mined knowledge in an easy-to-understand fashion.

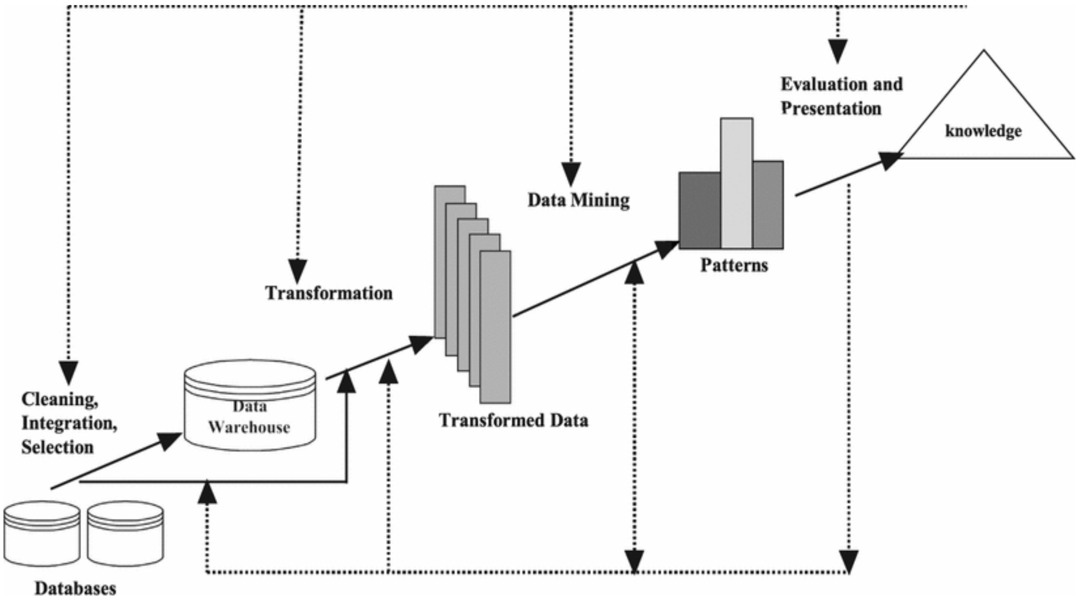


Figure-3: The Process of KDD

1.1.3 The Privacy Concern and PPDM

The objective of PPDM is to safeguard sensitive information from unsolicited or unsanctioned disclosure, and meanwhile, preserve the utility of the data. The consideration of PPDM is two-fold. First, sensitive raw data, such as individual’s ID card number and cell phone number, should not be directly used for mining. Second, sensitive mining results whose disclosure will result in privacy violation should be excluded.

To protect privacy of individual, several methods can be applied on data before or along the process of mining. PPDM techniques can be classified according to current studies, which are described in detail in this section.

A. Classification of PPDM

The PPDM techniques can be broadly classified as illustrated by fig.4, based on two scenarios: Central Server and Distributed.

I. Central Server Scenario: In this Scenario, the PPDM techniques deal with how data is protected before publishing it for data mining task. It is also referred as Data Publishing scenario. Here data owners/data miners are independent of handling privacy issues. Any number of miners can be involved in the process of mining with respect to the published data.

II. Distributed Scenario: In distributed scenario, PPDM techniques deal with the protection against private databases involved in the process of mining. Here the data owners can also be the miners and get aggregate results on the union of their databases. This is a situation where the privacy is ensured on results of data mining. Most of the works in this regard have specific goals in data mining and involve more complex procedures like Cryptographic based on Secure Multi-party Computation (SMC) principle.

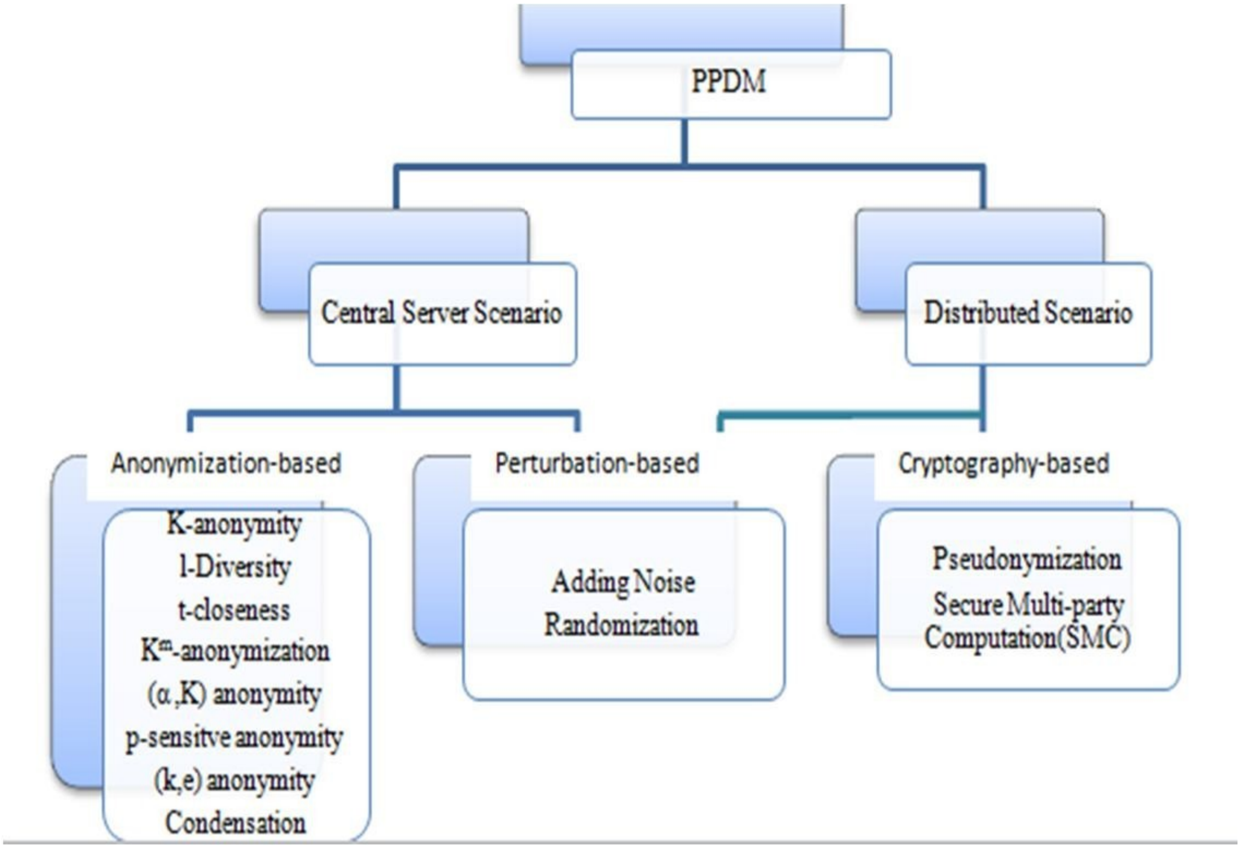


Figure-4: PPDM Classification Hierarchy

Privacy preserving data mining (PPDM) has emerged to address this issue. Most of the techniques for PPDM uses modified version of standard data mining algorithms, where the modifications usually using well known cryptographic techniques ensure the required privacy for the application for which the technique was designed. The several approaches used by PPDM can be summarized as below:

1. The data is altered before delivering it to the data miner.
2. The data is distributed between two or more sites, which cooperate using a semi-honest protocol to learn global data mining results without revealing any information about the data at their individual sites.

B. Privacy Preserving Techniques

The main objective of privacy preserving data mining is to develop data mining methods without increasing the risk of mishandling of the data used to generate those methods. Most of the techniques use some form of alteration on the original data in order to attain the privacy preservation. The altered dataset is obtainable for mining and must meet privacy requirements without losing the benefit of mining.

1. **Randomization**

Randomization technique is an inexpensive and efficient approach for privacy preserving data mining (PPDM). In order to assure the performance of data mining and to preserve individual privacy, this randomization schemes need to be implemented. The randomization approach protects the customers' data by letting them arbitrarily alter their records before sharing them, taking away some true information and introducing some noise. Some methods in randomization are numerical randomization and item set randomization Noise can be introduced either by adding or multiplying random values to numerical records or by deleting real items and adding "fake" values to the set of attributes.

2. **Anonymization**

To protect individuals' identity when releasing sensitive information, data holders often encrypt or remove explicit identifiers, such as names and unique security numbers. However, unencrypted data provides no guarantee for anonymity. In order to preserve privacy, k-anonymity model has been proposed by Sweeney which achieves k-anonymity using generalization and suppression, In K-anonymity, it is difficult for an imposter to determine the identity of the individuals in collection of data set containing personal information. Each release of data contains every combination of values of quasi-identifiers and that is indistinctly matched to at least k-1 respondents.

3. **Secure multi-party computation**

An alternative approach based on the multiparty computation is that every part of private data is validly known to one or more parties. Revealing private data to parties such as by whom the data is owned or the individual to whom the data refers to is not a condition of violating privacy. The problem arises when the private information is revealed to some other third parties. To deal with this problem, we use a specialized form of privacy preserving distributed data mining. Parties that each knows some of the private data participate in a protocol that generates the data mining results, that guarantees no data items is revealed to other parties. Thus the process of data mining doesn't cause, or even increase the opportunity for breach of privacy.

4. **Sequential pattern hiding**

Sequential pattern hiding method is necessary to conceal sensitive patterns that can otherwise be extracted from published data, without seriously affecting the data and the non-sensitive interesting patterns. Sequential pattern hiding is a challenging problem, because sequences have more composite semantics than item sets, and calls for efficient solutions that offer high utility.

1.2 Problem Definition

Security and privacy issues are magnified by velocity, volume, and variety of big data, such as large-scale cloud infrastructures, diversity of data sources and formats, streaming nature of data acquisition, and high volume inter-cloud migration. Therefore, traditional security mechanisms, which are tailored to securing small-scale static (as opposed to streaming) data, are inadequate. In this paper, we highlight top ten big data-specific security and privacy challenges.

The term big data refers to the massive amounts of digital information companies and governments collect about us and our surroundings. Every day, we create 2.5 quintillion bytes of data—so much that 90% of the data in the world today has been created in the last two years alone. Security and privacy issues are magnified by velocity, volume, and variety of big data, such as large-scale cloud infrastructures, diversity of data sources and formats, streaming nature of data acquisition and high volume inter-cloud migration. The use of large scale cloud infrastructures, with a diversity of software platforms, spread across large networks of computers, also increases the attack surface of the entire system. Traditional security mechanisms, which are tailored to securing small-scale static data, are inadequate.

1.3 Significance of work

We highlight the top ten big data specific security and privacy challenges. We interviewed Cloud Security Alliance members and surveyed security practitioner-oriented trade journals to draft an initial list of high-priority security and privacy problems, studied published research, and arrived at the following top ten challenges:

1. Secure computations in distributed programming frameworks
2. Security best practices for non-relational data stores
3. Secure data storage and transactions logs
4. End-point input validation/filtering
5. Real-time security/compliance monitoring
6. Scalable and compassable privacy-preserving data mining and analytics
7. Cryptographically enforced access control and secure communication
8. Granular access control
9. Granular audits
10. Data provenance

Secure Computations in Distributed Programming Frameworks

Distributed programming frameworks utilize parallelism in computation and storage to process massive amounts of data. A popular example is the MapReduce framework, which splits an input file into multiple chunks. In the first phase of MapReduce, a Mapper for each chunk reads the data, performs some computation, and outputs a list of key/value pairs. In the next phase, a Reducer combines the values belonging to each distinct key and outputs the result. There are two major attack prevention measures: securing the mappers and securing the data in the presence of an untrusted mapper.

Security best practices for non-relational data stores

Non-relational data stores popularized by NoSQL databases are still evolving with respect to security infrastructure. For instance, robust solutions to NoSQL injection are still not mature. Each NoSQL DBs were built to tackle different challenges posed by the analytics world and hence security was never part of the model at any point of its design stage. Developers using NoSQL databases usually embed security in the middleware.

Secure Data Storage and Transactions Logs

Data and transaction logs are stored in multi-tiered storage media. Manually moving data between tiers gives the IT manager direct control over exactly what data is moved and when. However, as the size of data set has been, and continues to be, growing exponentially, scalability and availability have necessitated auto-tiering for big data storage management. Auto-tiering solutions do not keep track of where the data is stored, which poses new challenges to secure data storage. New mechanisms are imperative to thwart unauthorized access and maintain the 24/7 availability.

End-Point Input Validation/Filtering

Many big data use cases in enterprise settings require data collection from many sources, such as end-point devices. For example, a security information and event management system (SIEM) may collect event logs from millions of hardware devices and software applications in an enterprise network. A key challenge in the data collection process is input validation: how can we trust the data? How can we validate that a source of input data is not malicious and how can we filter malicious input from our collection? Input validation and filtering is a daunting challenge posed by untrusted input sources, especially with the bring your own device (BYOD) model.

Real-time Security/Compliance Monitoring

Real-time security monitoring has always been a challenge, given the number of alerts generated by (security) devices. These alerts (correlated or not) lead to many false positives, which are mostly ignored or simply “clicked away,” as humans cannot cope with the shear amount. This problem might even increase with big data, given the volume and velocity of data streams. However, big data technologies might also provide an opportunity, in the sense that these technologies do allow for fast processing and analytics of different types of data. Which in its turn can be used to provide, for instance, real-time anomaly detection based on scalable security analytics.

Scalable and compassable privacy-preserving data mining and analytics

Big data can be seen as a troubling manifestation of Big Brother by potentially enabling invasions of privacy, invasive marketing, decreased civil freedoms, and increase state and corporate control. A recent analysis of how companies are leveraging data analytics for marketing purposes identified an example of how a retailer was able to identify that a teenager was pregnant before her father knew. Similarly, anonymizing data for analytics is not enough to maintain user privacy. For example, AOL released anonymized search logs for academic purposes, but users were easily identified by their searchers. Netflix faced a similar problem when users of their anonymized data set were identified by correlating their Netflix movie scores with IMDB scores.

Cryptographically Enforced Access Control and Secure Communication

To ensure that the most sensitive private data is end-to-end secure and only accessible to the authorized entities, data has to be encrypted based on access control policies. Specific research in this area such as attribute-based encryption (ABE) has to be made richer, more efficient, and scalable. To ensure authentication, agreement and fairness among the distributed entities, a cryptographically secure communication framework has to be implemented.

Granular Access Control

The security property that matters from the perspective of access control is secrecy—preventing access to data by people that should not have access. The problem with course-grained access mechanisms is that data that could otherwise be shared is often swept into a more restrictive category to guarantee sound security. Granular access control gives data managers a scalpel instead of a sword to share data as much as possible without compromising secrecy.

Granular Audits

With real-time security monitoring (see section 12.0), we try to be notified at the moment an attack takes place. In reality, this will not always be the case (e.g., new attacks, missed true positives). In order to get to the bottom of a missed attack, we need audit information. This is not only relevant because we want to understand what happened and what went wrong, but also because compliance, regulation and forensics reasons. In that regard, auditing is not something new, but the scope and granularity might be different. For example, we have to deal with more data objects, which probably are (but not necessarily) distributed.

Data Provenance

Provenance metadata will grow in complexity due to large provenance graphs generated from provenance-enabled programming environments in big data applications. Analysis of such large provenance graphs to detect metadata dependencies for security/confidentiality applications is computationally intensive.

1.3 Existing Work

PPDM Techniques

In this section, the PPDM algorithms can be interpreted in detail according to the technique/mechanism it uses to achieve privacy. The techniques can be listed under three major classical PPDM techniques: Anonymization, Perturbation and Cryptographic.
This paper explores various PPDM algorithms based on the above classification hierarchy.

A. Anonymization Based

Sometimes the data must be publically published in its original form. Even though it is not encrypted and perturbed, some sort of precaution should be implemented before releasing the data in terms of anonymization. This is a kind of generalization of some attributes which protects against identity disclosure. Anonymization can be obtained through methods such as generalization, suppression, data removal, permutation, swapping etc. k-anonymity method is treated as the classical anonymization method and most of the studies are based on k-anonymity.

Previous works by Samarati and Sweeney shows that the removal of the personally identifying information from data is insufficient for the data protection, rather it is better to use k – anonymity method for publishing data. The quasi–identifier (QI), which is the combination of person specific identifiers are considered here for the process of anonymization. One of the common methods to achieve k –anonymity is to generalize identifiers (for example date of birth can be generalized to month of birth). The experimental results of their study indicate that their approaches produce k – anonymization with less generalization compared to previous approaches. They conclude that a bottom-up approach for k – anonymization is preferable for small number of quasi- identifying attributes.

A task independent technique based on anonymization which preserves information, privacy and utility of data is introduced in [13].Their algorithm is applied on the original data table to alter only the sensitive raw data before applying any mining methods. In most of the Privacy preserving generalization methods, loss of information is due to generalization (transformation) of QI attributes and sensitive attributes. They demand both privacy and no information loss by only transforming part of the QI and sensitive attributes. The [22] proposes a modified entropy l-diversity model in order to address privacy of medical data. Here, more detailed attacking conditions and characteristic of medical information are taken into consideration.

Approaches which address specific problems are also developed using anonymization method. The k- anonymity based method is illustrated in [31] is used to search for optimal feature set partitioning and [34] for cluster analysis. And [33] proposes a data reconstruction approach to achieve k-anonymity protection in predictive data mining. In this approach the potentially identifying attributes are first mapped using aggregation for numeric data and swapping for nominal data. A genetic algorithm technique is then applied to the masked data to find a good subset of it. This subset is then replicated to form the released dataset that satisfies the k-anonymity constraint.

Another anonymization technique known, as Condensation is a statistical approach which constructs constrained clusters in dataset and then generates pseudo data from the statistics of these clusters [9]. This method is called as condensation because of its approach of using condensed statistics of the clusters to generate pseudo data. It constructs groups of non-homogeneous size from the whole data, such that it is guaranteed that each record lies in a group whose size is at least equal to its anonymity level. Here, pseudo data is generated from each group so as to create a synthetic data set with the same aggregate distribution as the original data. This approach also can be effectively used for the problem of classification .The pseudo- data provides an additional layer of protection, as it becomes difficult to perform adversarial attacks on synthetic data. Since the aggregate behavior of the data is preserved, it becomes useful for a variety of data mining problems.

B. Perturbation Based

Disclosing a ‘perturbed’ version of a data before releasing it for data mining is one of the data distortion method for privacy protection. Adding noise from a known distribution is one of the perturbation technique widely accepted. Before conducting a data mining operation, the miner should reconstruct the perturbed version to obtain the original data distribution. Perturbation methods can be used in both scenarios, central server as well as in distributed scenario. These methods use some sort of data distortion techniques like adding noise, randomization or condensation. Studies in distributed scenario include [18,19,41].An approach based on geometric data perturbation and a data mining service oriented framework is introduced in[18].The paper [19] propose a perturbation based technique which modifies the data mining algorithm, so that they can be directly used on the perturbed data. They directly build a classifier for the original dataset from the perturbed training dataset by skipping the steps of reconstructing the original data distribution. The study [41] deals with clustering problem in the distributed environment. Their approach is based on Principal Component Analysis (PCA) technique instead of Geometrical Data Transformation Methods (GDTM) used traditionally.

Even though, Perturbation techniques can be used for achieving privacy in data publishing and also in the process of data mining these have certain limitations: i) since this model uses distributions instead of original records, it restricts the range of algorithmic techniques that can be used on the data. ii) And another limitation is the loss of implicit information available in multidimensional records.

C. Cryptography Based

The approaches discussed in the previous sections are applicable when data can be disclosed beyond the control of the data collection process. If the data is distributed across multiple sites which are legally prohibited from sharing their collections with each other, it is still possible to construct a data mining model. The paper [6] illustrated this scenario. It also addresses the problem of reconstructing missing values of building an accurate data mining model. They propose a cryptographic protocol based on decision-tree classification on horizontally partitioned databases. They assume that there are two data source and allows each data source to compute missing values without sharing any information about their data providing complete privacy preservation. Cryptographic techniques are extensively studied in distributed environment. The Oblivious Transfer is used as the basic building block in [11]. Another efficient privacy preserving protocol is described in [12] which provide a solution for the specific problem of distributed ID3. The implementation of these protocols is by using hard-wired circuits.

Secure multiparty computation (SMC) is a technique that can be used to maintain privacy in different distributed data mining environments. SMC can be based on three general types of techniques: homo morphic encryption, circuit evaluation and secret sharing [16]. SMC protocols deals with two types of adversaries: semi-honest and malicious. The majority of applications are based on semi-honest types, in which the

adversaries follow the protocol specification but try to learn information exchanged more than the results of the protocol. The paper [7] illustrates that SMC protocols with malicious adversary model have high complexities. They also claim that these models are not suitable for data mining application in its current form. The authors propose an "Accountable Computing (AC) framework" which assigns liability for privacy to the responsible party. The paper [17] includes an analysis of the accuracy and efficiency of protocols based on SMC. The [27] provides a privacy preserving framework for SMC, based on Gaussian mixture models. The protocol mentioned in [30] is based on encryption which protects the privacy of each distributed database. The protocol needs only two communications between each data site and the mixer in one round of data collection. The [32, 36] introduce a cryptographic approach for privacy preservation for classification problem. The protocol mentioned in [35] is based on homomorphic encryption for association rule mining.

Pseudonymization is an approach that breaks the link between personal and medical information. It provides a form of traceable anonymity of health records. Instead of completely removing personal identification information from the medical data, identification information is transferred into a piece of information (i.e., a pseudonym) which cannot be mapped to a patient without knowing a certain secret. Encryption is a well-established technique for building pseudonyms [25].The encryption can be performed either at the database- level or at the application-level. Their work is based on the application-level encryption.

COMPARISON OF PPDM TECHNIQUES

Table I. Comparison of PPDM Techniques

Technique	Method/s	Scenario/s	Data Mining Task			
			Classification	Clustering	Association rule	ection Outlier
Anonymization	Generalization, Suppression, Permutation	Data Publishing	☑	☑	☑	☑
Perturbation	Adding Noise, Swapping	Data Publishing, Distributed	☑	☑	☑	☑
Randomization	Adding Noise, Scrambling	Data Publishing, Distributed	☑	☐	☐	☐
Condensation	Aggregation	Data Publishing	☑	☐	☐	☐
SMC	Cryptographic	Distributed	☑	☑	☑	☑
Pseudonymization	Cryptographic	Distributed	☑	☑	☑	☑

Using Tab.1, we present a preliminary comparison of various PPDM techniques, to provide an insight into, which technique is suitable for which scenario. It also illustrates the methods that are normally used by different techniques and suggests a number of techniques that can be implemented for solving particular problems in Data Mining.

LITERATURE REVIEW

Current models and algorithms proposed for PPDM mainly focus on how to hide those sensitive information from certain mining operations. However, the whole KDD process involve multi-phase operations. Besides the mining phase, privacy issues may also arise in the phase of data collecting or data preprocessing, even in the delivery process of the mining results. If sensitive information is lost or used in any way other than intended, the result can be severe damage to the person or organization to which that information belongs. The term "sensitive data" refers to data from which sensitive information can be extracted. Throughout the paper, we consider the two terms "privacy" and "sensitive information" are interchangeable.

In this paper, four different types of users, namely four *user roles*, in a typical data mining scenario (see Fig.):

- **Data Provider**: the user who owns some data that are desired by the data mining task.
- **Data Collector**: the user who collects data from data providers and then publish the data to the data miner.
- **Data Miner**: the user who performs data mining tasks on the data.
- **Decision Maker**: the user who makes decisions based on the data mining results in order to achieve certain goals.

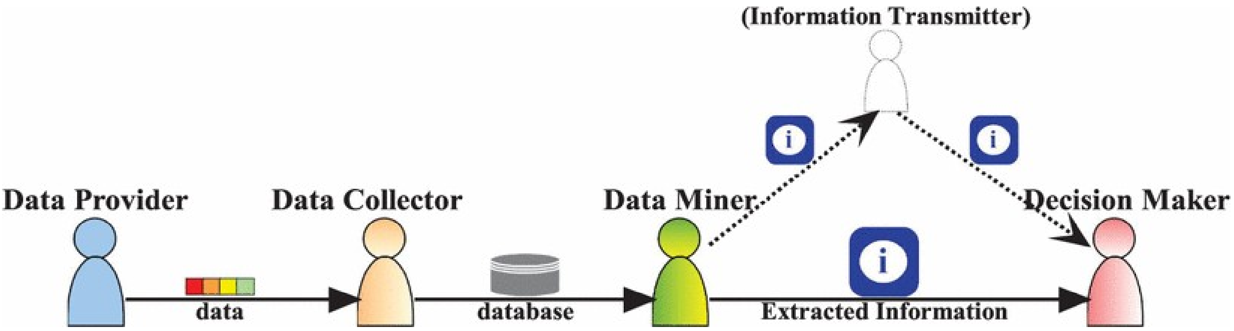


Figure-5: A simple illustration of the application scenario with data mining at the core.

By differentiating the four different user roles, we can explore the privacy issues in data mining in a principled way. All users care about the security of sensitive information, but each user role views the security issue from its own perspective.

I. Data Provider

The major concern of a data provider is whether he can control the sensitivity of the data he provides to others. On one hand, the provider should be able to make his very private data, namely the data containing information that he does not want anyone else to know, inaccessible to the data collector. On the other hand, if the provider has to provide some data to the data collector, he wants to hide his sensitive information as much as possible and get enough compensation for the possible loss in privacy.

II. Data Collector

The data collected from data providers may contain individuals' sensitive information. Directly releasing the data to the data miner will violate data providers' privacy, hence data modification is required. On the other hand, the data should still be useful after modification, otherwise collecting the data will be meaningless. Therefore, the major concern of data collector is to guarantee that the modified data contain no sensitive information but still preserve high utility.

III. Data Miner

The data miner applies mining algorithms to the data provided by data collector, and he wishes to extract useful information from data in a privacy-preserving manner. PPDM covers two types of protections, namely the protection of the sensitive data themselves and the protection of sensitive mining results. With the user role-based methodology proposed in this paper, we consider the data collector should take the major responsibility of protecting sensitive data, while data miner can focus on how to hide the sensitive mining results from untrusted parties.

IV. Decision Maker

As decision maker can get the data mining results directly from the data miner, or from some *Information Transmitter*. It is likely that the information transmitter changes the mining results intentionally or unintentionally, which may cause serious loss to the decision maker. Therefore, what the decision maker concerns is whether the mining results are credible.

SYSTEM ANALYSIS

3.1 Requirement Model

User Interface Design:

In this module we create a user page using Graphical User Interface(GUI), which will be the media to Connect user with the server and through which client can able to give request to the server and server can send the response to the client, through this module we can establish the communication between client and server using webpage.

A program interface that takes advantage of the computer's graphics capabilities to make the program easier to use. Well-designed graphical user interfaces can free the user from learning complex command languages. The user interacts with information by manipulating visual widgets that allow for interactions appropriate to the kind of data they hold. The widgets of a well-designed interface are selected to support the actions necessary to achieve the goals of the user.

Create Multiple Organizations:

This is second module of our project. Here we are design no. of parties. Each and every party may have information to store their database. All the parties may send their inputs to Data Analysis module. Here all n no. of parties will send their inputs to single data analysis. The data analysis will store their inputs either horizontal or vertical partitions.

Data Analysis and Integration:

Our Data Analysis designed using cryptographic techniques. Data are generally assumed to be either vertically or horizontally partitioned. In the case of horizontally partitioned data, different sites collect the same set of information about different entities. In the case of vertically partitioned data, we assume that different sites collect information about the same set of entities. A party can store their input data either vertical partition or horizontal partitioned.

Inputs computation model:

This model to design for compute all the truthful inputs of all participating parties here going to assumptions like the first priority for every participating party is to learn the correct result. Another one is, if possible, every participating party prefers to learn the correct result exclusively.

Association Data Mining:

Association Data mining as mining technique. The association rule mining and analyze whether the association rule mining can be done in an incentive compatible manner over horizontally or vertically partitioned database. If get in the requested query then it search where it is located either horizontal partition or vertical partition retrieve the result from partition after that result send to particular party.

- Either $\exists v_{-i} \in D_{-i},\ g_i(f(t_i(v_i), v_{-i}), v_i) \neq f(v_i, v_{-i})$
- Or $\forall v_{-i} \in D_{-i},\ f(t_i(v_i), v_{-i}) = f(v_i, v_{-i})$

------- [3.1.1]

3.2 Sample UML Diagrams for the project work

USE CASE DIAGRAM

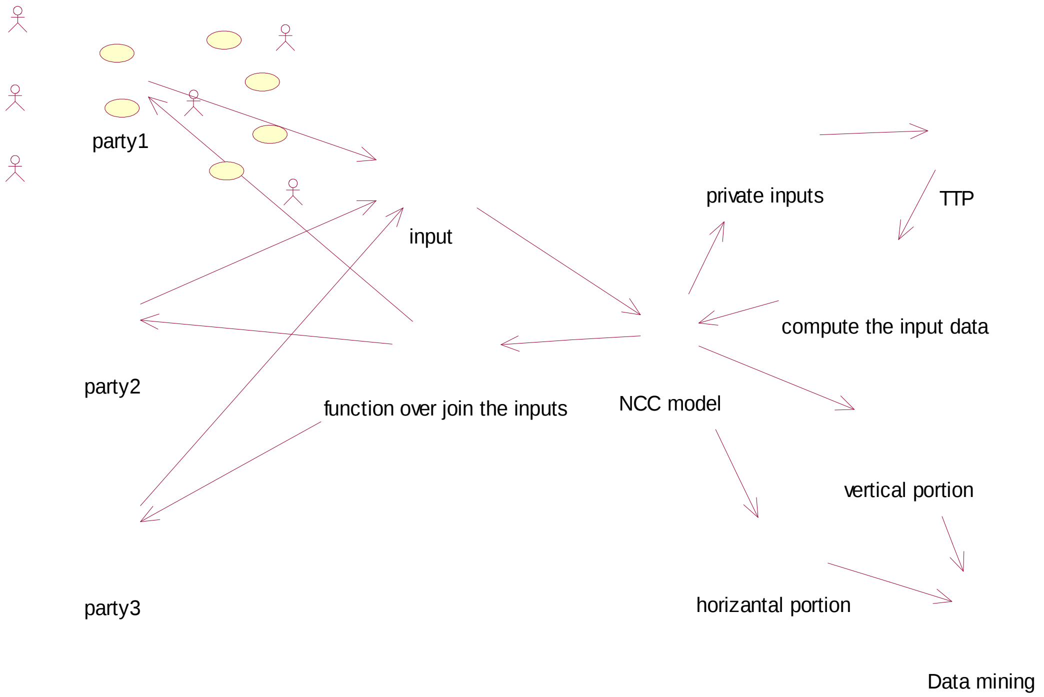


Figure-6: Use case diagram

CLASS DIAGRAM

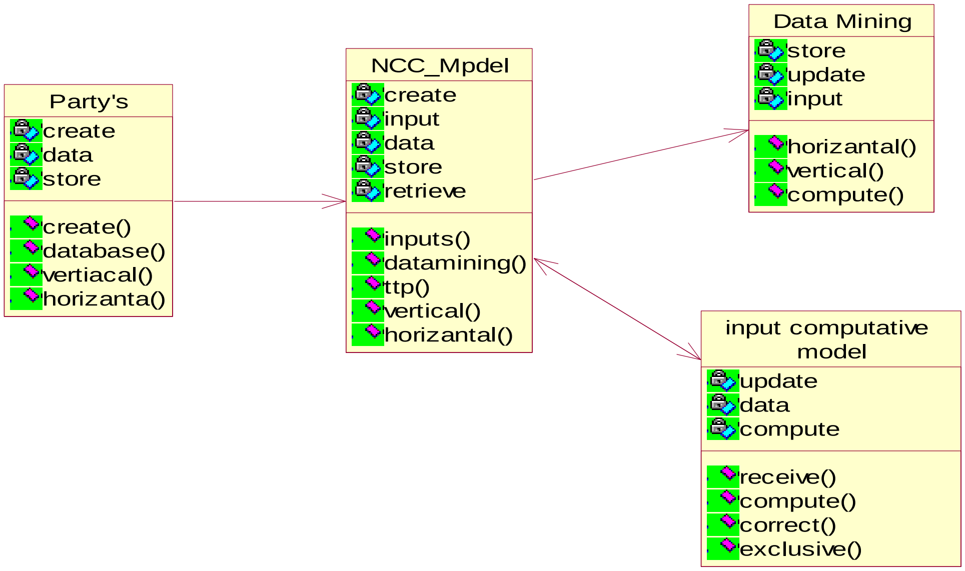


Figure-7: Class diagram

SEQUENCE DIAGRAM

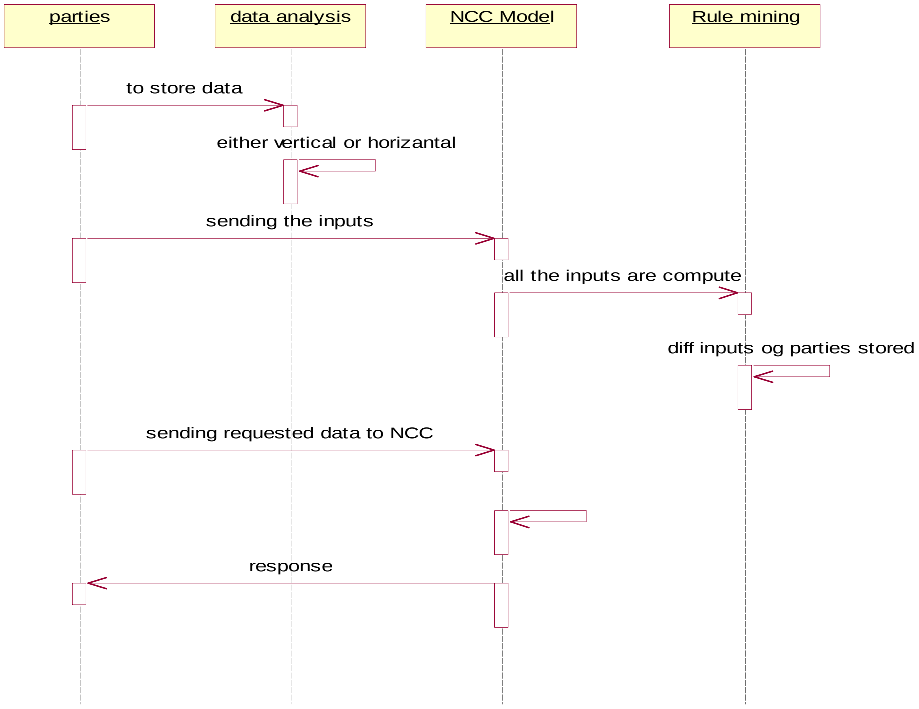

parties
data analysis
NCC Model
Rule mining
to store data
either vertical or horizantal
sending the inputs
all the inputs are compute
diff inputs og parties stored
sending requested data to NCC
response

Figure-8: Sequence diagram

ACTIVITY DIAGRAM

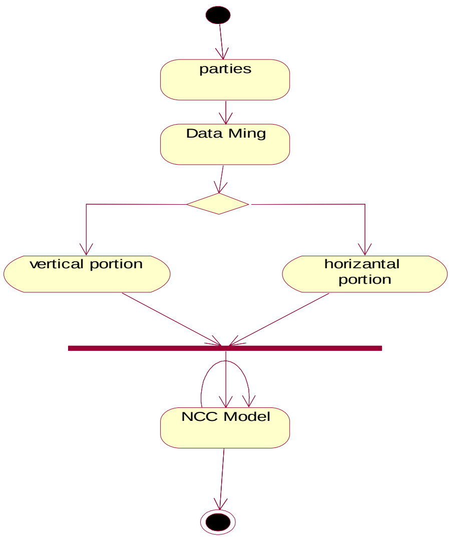


Figure-9: Activity diagram

SYSTEM DESIGN

4.1 Architecture of Data Mining

The architecture contains modules for secure safe-thread communication, database connectivity, organized data management and efficient data analysis for generating global mining model.

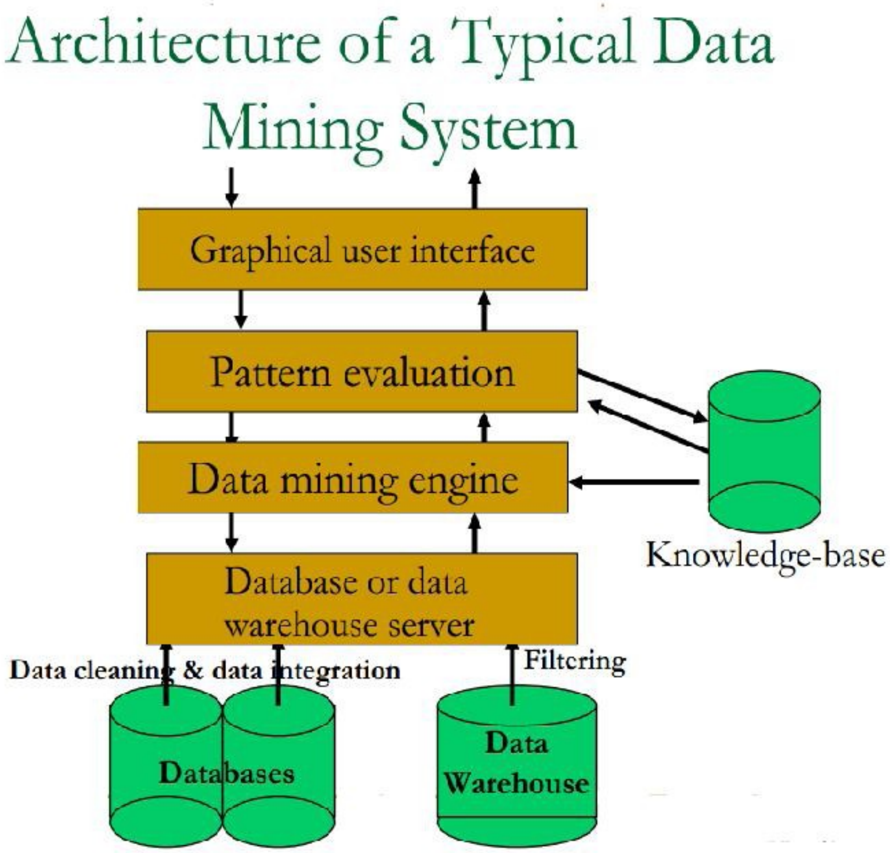

Architecture of a Typical Data Mining System
Graphical user interface
Pattern evaluation
Data mining engine
Knowledge-base
Database or data warehouse server
Data cleaning & data integration
Filtering
Databases
Data Warehouse

Figure-10: Architecture of a typical data mining system

4.2 Architecture behind the proposed model

The architecture behind the proposed model is illustrated in Figure The client/owner encrypts its transaction database (TDB). Using an encrypt/decrypt module, which can be essentially treated as a 'black box' from its perspective. This module is responsible for transforming the TDBD into an encrypted database D∗. The server conducts data mining and sends the (encrypted) patterns to the owner. The encryption scheme has the property that the returned number of occurrences of the patterns is not true. The encrypt/decrypt module recovers the true identity of the returned patterns as well their true number of occurrences. The strong theoretical results show a remarkable guarantee of protection against the attacks presented and the practicability and the effectiveness of the proposed schema. The application of this framework on real-world databases showed that the privacy protection is much better than the theoretical worst case.

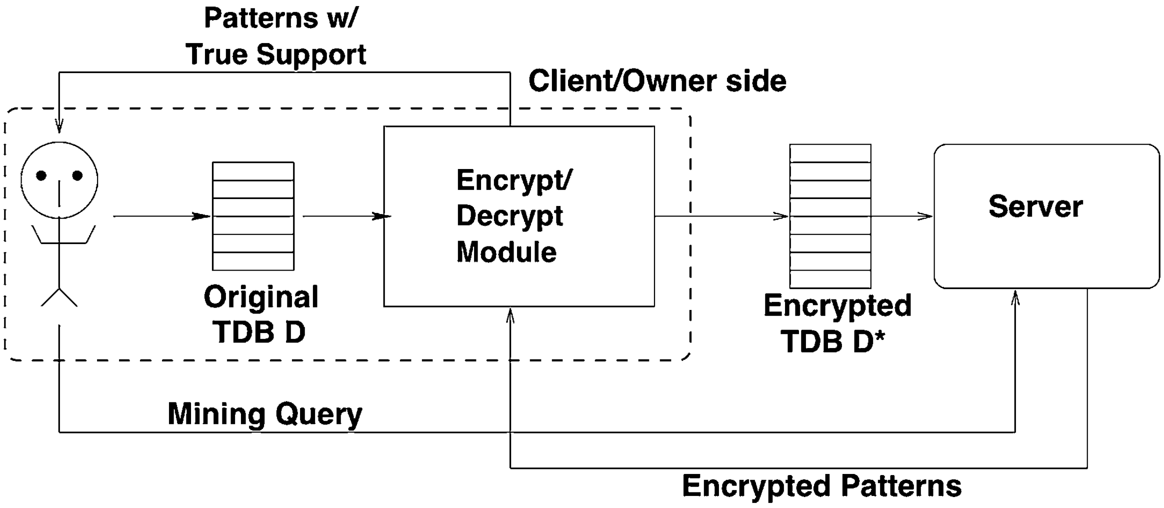


Figure-11: Mining Query

4.3 work flow of proposed system

In below diagram contains client Login, Database, Work Allocation, Worker Page, Computing, Reposting, and Work Grouping. First computation node will start running. After party node enter user name and password that is validated by compatible node. Then computation node assigns the work to the data mining nodes. Data mining node finishes his work and reposted to the compatible node. TTP collects the inputs of parties and group of parties input for particular work presented by party nodes.

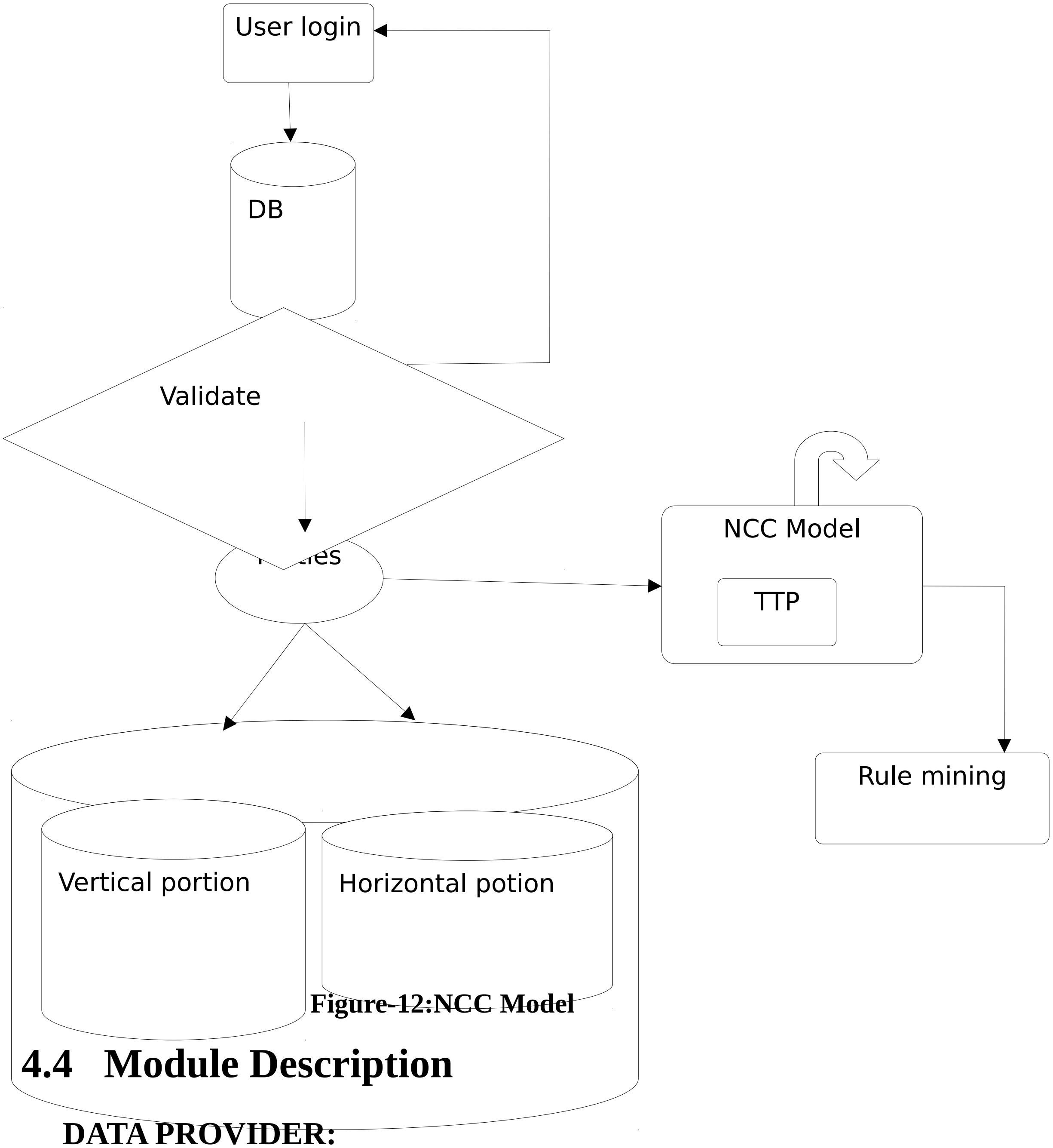


Figure-12:NCC Model

4.4 Module Description

DATA PROVIDER:

A. Concerns of Data Provider

There are actually two types of data providers: One refers to the data provider who provides data to data collector. The other refers to the data collector who provides data to data miner. Data reporting information about an individual are often referred to as ``micro data'‘. If a data provider reveals his micro data to the data collector, his privacy might be comprised due to the unexpected data breach or exposure of sensitive information.

To investigate the measures that the data provider can adopt to protect privacy, we consider the following three situations:

1) If the data provider considers his data to be very sensitive and he refused to provide such data. Some effective measures are desired by provider to prevent his sensitive data from data collector.
2) Data provider may hand over some data to data collector in exchange for certain benefits. He needs to know how to negotiate data collector so that he can get compensation for any possible loss in data.
3) If data provider cannot protect data, Then data provider can distort his data that will be fetched by the data collector, so that his true information cannot be easily disclosed.

B. Approaches to Privacy Protection

LIMIT THE ACCESS

A data provider provides his data to the collector in active way or a passive way.

By active we mean that the data provider voluntarily opts in a survey initiated by the data collector, or fill in some registration forms to create an account in a website. By passive we mean that the data which are generated by the provider's routine activities are recorded by the data collector.

Eg: If data provider is an internet user who is afraid of expose of his online activities can erase his browser’s cache and delete the cookies.

Based on their basic functions, current security tools can be categorized into the following three types:

- **Anti-tracking extensions** - A major technology used for anti-tracking is called Do Not Track

- **Advertisement and script blockers -** This type of browser extensions can block advertisements on the sites. The tools used here are AdBlock Plus, NoScript.
- **Encryption tools -** To make sure a private online communication between two parties cannot be intercepted by third parties, a user can utilize encryption tools, such as MailCloak and TorChat.

TRADE PRIVACY FOR BENEFIT

In some cases, the data provider needs to make a tradeoff between the loss of privacy and the benefits brought by participating in data mining. Suppose data collector need some information about some data item, Then he would pay for that item. If data provider considers salary as his sensitive information, then based on the prices offered by the collector, he chooses one of the following actions:

i) Not to report his salary, if he thinks the price is too low;

ii) To report a fuzzy value of his salary, e.g. ``less than 10,000 dollars", if he thinks the price is just acceptable.

iii) To report an accurate value of his salary, if he thinks the price is high enough.

PROVIDE FALSE DATA

Internet users cannot completely stop the unwanted access to their personal information. So instead of trying to limit the access, the data provider can provide false information to those untrustworthy data collectors. The following three methods can help an Internet user to falsify his data:

- Using ``sock puppets" to hide one's true activities.
- Using a fake identity to create phony information.
- Using security tools to mask one's identity.

DATA COLLECTOR

A. Concerns of Data Collector

A data collector collects data from data providers in order to support the Sub sequent data mining operations. The original data collected from data providers usually contain sensitive information about individuals. If the data collector doesn't take sufficient precautions before releasing the data to public or data miners, those sensitive information may be disclosed, even though this is not the collector's original intention. The data modification process adopted by data collector, with the goal of preserving privacy and utility simultaneously, is usually called privacy preserving data publishing (PPDP).

B. Approaches To Privacy Protection

BASICS OF PPDP

The original data is assumed to be a private table consisting of multiple records.

- Identifier (ID) - Attributes that can directly and uniquely identify an individual,
- Quasi-identifier (QID) - Attributes that can be linked with external data to re-identify individual records
- Sensitive Attribute (SA) Non-sensitive Attribute (NSA) - Attributes that an individual wants to conceal, such as disease and salary.
- Non-sensitive Attribute (NSA): Attributes other than ID, QID and SA.

How the data table should be anonymized mainly depends on how much privacy we want to preserve in the anonymized data.

Typical privacy models are

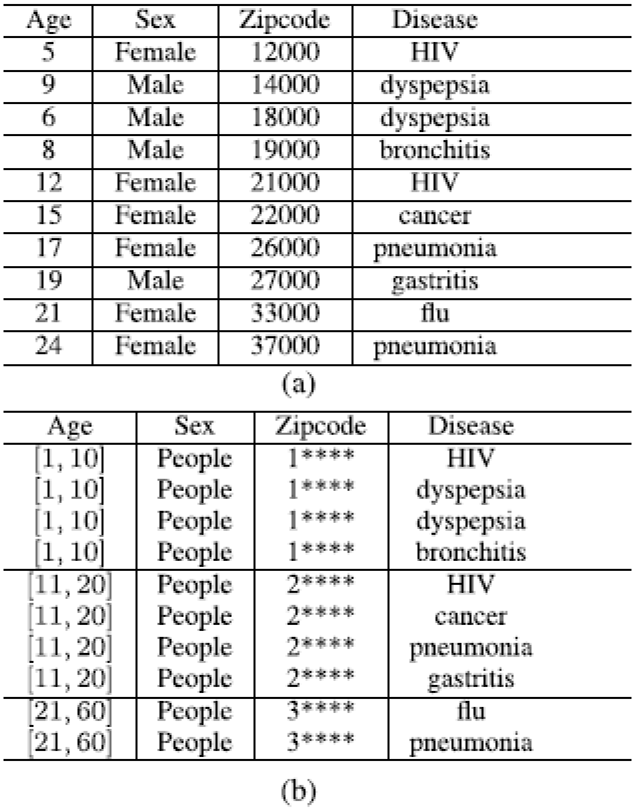
- *k*-anonymity (for preventing record linkage),
- *l*-diversity (for preventing record linkage and attribute linkage),
- *t*-closeness (for preventing attribute linkage and probabilistic attack),
- epsilon-differential privacy (for preventing table linkage and probabilistic attack).

The anonymization operations which are applied for privacy are Generalization, Suppression, Anatomization, Permutation and Perturbation.

- Generalization. This operation replaces some values with a parent value in the taxonomy of an attribute. Typical generalization schemes including full-domain generalization, subtree generalization, multidimensional generalization, etc.

- Suppression. This operation replaces some values with a special value (e.g. a asterisk '*'), indicating that the replaced values are not disclosed. Typical suppression schemes include record suppression, value suppression, cell suppression, etc.
- Anatomization. This operation does not modify the quasi-identifier or the sensitive attribute, but de-associates the relationship between the two. Anatomization-based method releases the data on QID and the data on SA in two separate tables.
- Permutation. This operation de-associates the relationship between a quasi-identifier and a numerical sensitive attribute by partitioning a set of data records into groups and shuffling their sensitive values within each group.
- Perturbation. This operation replaces the original data values with some synthetic data values, so that the statistical information computed from the perturbed data does not differ significantly from the statistical information computed from the original data. Typical perturbation methods include adding noise, swapping data, and generating synthetic data.

The anonymization operations will reduce the utility of data. The reduction of data utility is usually represented by *information loss*: higher information loss means lower utility of the anonymized data.



Age	Sex	Zipcode	Disease
5	Female	12000	HIV
9	Male	14000	dyspepsia
6	Male	18000	dyspepsia
8	Male	19000	bronchitis
12	Female	21000	HIV
15	Female	22000	cancer
17	Female	26000	pneumonia
19	Male	27000	gastritis
21	Female	33000	flu
24	Female	37000	pneumonia

(a)

Age	Sex	Zipcode	Disease
[1, 10]	People	1****	HIV
[1, 10]	People	1****	dyspepsia
[1, 10]	People	1****	dyspepsia
[1, 10]	People	1****	bronchitis
[11, 20]	People	2****	HIV
[11, 20]	People	2****	cancer
[11, 20]	People	2****	pneumonia
[11, 20]	People	2****	gastritis
[21, 60]	People	3****	flu
[21, 60]	People	3****	pneumonia

(b)

Figure-13:Anonymization

ATTACK MODEL

Given the anonymized network data, adversaries usually rely on background knowledge to de-anonymize individuals and learn relationships between de-anonymized individuals. Peng propose an algorithm called *Seed-and-Grow* to identify users from an anonymized social graph, based solely on graph structure. The algorithm a seed sub-graph which is either planted by an attacker or divulged by collusion of a small group of users, and then grows the seed larger based on the adversary's existing knowledge of users' social relations.

Zhu design a *structural attack* to de-anonymize social graph data.

Sun introduce a relationship attack model called *mutual friend attack*, which is based on the number of mutual friends of two connected individuals. Fig. shows an example of the mutual friend attack. The original social network *G* with vertex identities is shown in Fig. 4(a), and Fig. 4(b) shows the corresponding anonymized network where all individuals names are removed. In this network, only Alice and Bob have 4 mutual friends. If an adversary knows this information, then he can uniquely re-identify the edge (*D*; *E*) in Fig. 4(b) is (*Alice*; *Bob*). Tai investigate the *friendship attack* where an adversary utilizes the degrees of two vertices connected by an edge to re-identify related victims in a published social network data set. Fig. 5 shows an example of friendship attack. Suppose that each user's friend count (i.e. the degree of the vertex) is publicly available. If the adversary knows that Bob has 2 friends and Carl has 4 friends, and he also knows that Bob and Carl are friends, then he can uniquely identify that the edge (2; 3) in Fig. 5(b) corresponds to (*Bob*; *Carl*) In [31], another type of attack, namely *degree attack*, is explored. The motivation is that each individual in a social network is inclined to associate with not only a vertex identity but also a community identity, and the community identity reflects some sensitive information about the individual. It has been shown that, based on some background knowledge about vertex degree, even if the adversary cannot precisely identify the vertex corresponding to an individual, community information and neighborhood information can still be inferred.

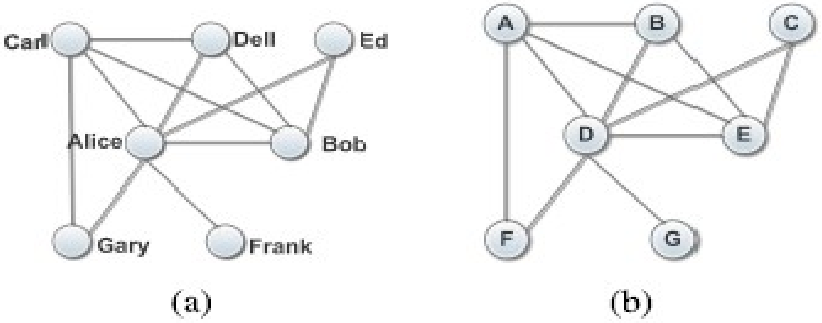


Figure-14:Mutual Friend Attack

For example, the network shown in Fig. 6 consists of two communities, and the community identity reveals sensitive information (i.e. disease status) about its members. Suppose that an adversary knows John has 5 friends, then he can infer that John has AIDS, even though he is not sure which of the two vertices (vertex 2 and vertex 3) in the anonymized network (Fig.6) corresponds to John. From above discussion we can see that, the graph data contain rich information that can be explored by the adversary to initiate an attack. Modeling the background knowledge of the adversary is difficult yet very important for deriving the privacy models.

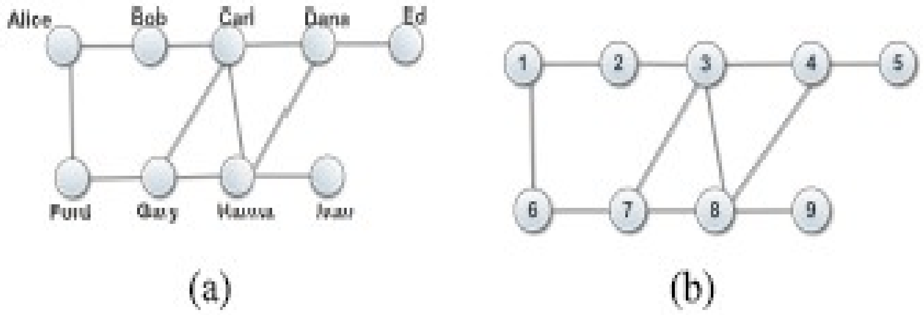


Figure-15:Friendship attack

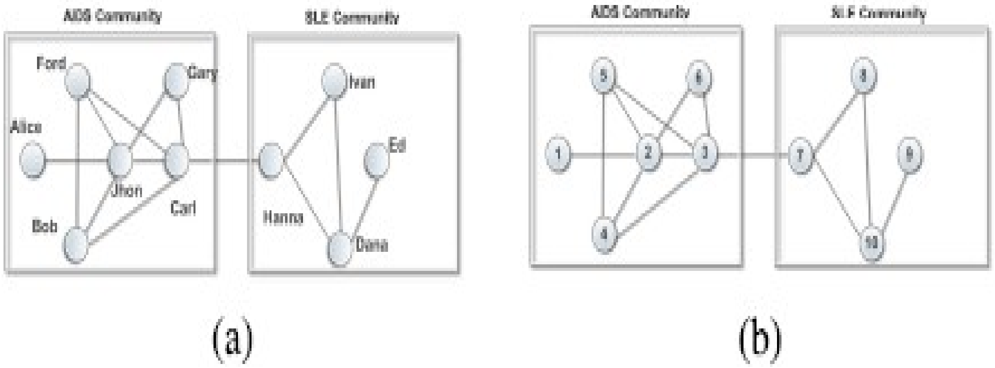


Figure-16:Degree attack

DATA MINER

A. Concerns of Data Miner

Data Miner applies data mining algorithms to the data obtained from Data Collector. The privacy issues coming with the data mining operations are twofold. On one hand, if personal information can be directly observed in the data and data breach happens, privacy of the original data owner (i.e. the data provider) will be compromised. On the other hand, equipping with the many powerful data mining techniques, the data miner is able to find out various kinds of information underlying the data.

B. Approaches to Privacy Protection

Extensive PPDM approaches have been proposed. These approaches can be classified by different criteria, such as data distribution, data modification method, data mining algorithm, etc. Based on the distribution of data, PPDM approaches can be classified into two categories, namely approaches for centralized data mining and approaches for distributed data mining. Distributed data mining can be further categorized into data mining over horizontally partitioned data and data mining over vertically partitioned data. Based on the technique adopted for data modification, PPDM can be classified into perturbation-based, blocking-based, swapping-based, etc. Since we define the privacy-preserving goal of data miner as preventing sensitive information from being revealed by the data mining results, in this section, we classify PPDM approaches according to the type of data mining tasks.

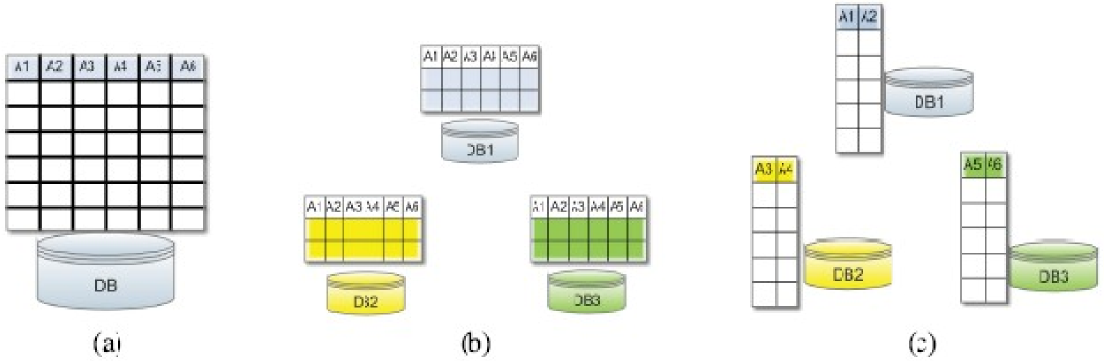


Figure-17: Data distribution. (a) centralized data. (b) horizontally partitioned data. (c) vertically partitioned data.

PRIVACY-PRESERVING ASSOCIATION RULE MINING

Association rule mining is one of the most important data mining tasks, which aims at finding interesting associations and correlation relationships among large sets of data items. The process of association rule mining contains the following two steps:

- Step 1: Find all frequent item sets. A frequent item set is an item set whose occurrence frequency is larger than a predetermined minimum support count.
- Step 2: Generate strong association rules from the frequent item sets. Rules that satisfy both a minimum support threshold (*minsup*) and a minimum confidence threshold (*minconf*) are called strong association rules.

Given the thresholds of *support* and *confidence*, the data miner can find a set of association rules from the transactional data set. Some of the rules are considered to be sensitive, either from the data provider’s perspective or from the data miner’s perspective. To hiding these rules, the data miner can modify the original data set to generate a *sanitized* data set from which sensitive rules cannot be mined, while those non-sensitive ones can still be discovered, at the same thresholds or higher.

Various kinds of approaches have been proposed to perform association rule hiding. These approaches can roughly be categorized into the following five groups:

- Heuristic distortion approaches, which resolve how to select the appropriate data sets for data modification.
- Heuristic blocking approaches, which reduce the degree of support and confidence of the sensitive association rules by replacing certain attributes of some data items with a specific symbol (e.g. '?').
- Probabilistic distortion approaches, which distort the data through random numbers generated from a predefined probability distribution function.
- Exact database distortion approaches, which formulate the solution of the hiding problem as a constraint satisfaction problem (CSP), and apply linear programming approaches to its solution.
- Reconstruction-based approaches, which generate a database from the scratch that is compatible with a given set of non-sensitive association rules.

The main idea behind association rule hiding is to modify the support and/or confidence of certain rules. Zhu employ *hybrid partial hiding* (HPH) algorithm to reconstruct the support of itemset, and then uses Apriori algorithm to generate frequent itemsets based on which only non-sensitive rules can be obtained. Le proposes a heuristic algorithm based on the intersection lattice of frequent itemsets for hiding sensitive rules. The algorithm first determines the *victim item* such that modifying this item causes the least impact on the set of frequent itemsets. Then, the minimum number of transactions that need to be modified are specified. After that, the victim item is removed from the specified transactions and the data set is sanitized.

Among different types of approaches proposed for sensitive rule hiding, we are particularly interested in the reconstruction-based approaches, where a special kind of data mining algorithms, named *inverse frequent set mining* (*IFM*), can be utilized. The problem of IFM was first investigated by Mielikäinen in. The IFM problem can be described as follows: given a collection of frequent itemsets and their support, find a transactional data set such that the data set precisely agrees with the supports of the given frequent itemset collection while the supports of other itemsets would be less than the pre-determined threshold. Guo propose a reconstruction-based approach for association rule hiding where data reconstruction is implemented by solving an IFM problem. Their approach consists of three steps:

- First, use frequent itemset mining algorithm to generate all frequent itemsets with their supports and support counts from original data set.
- Second, determine which itemsets are related to sensitive association rules and remove the sensitive itemsets.
- Third, use the rest itemsets to generate a new transactional data set via inverse frequent sets.

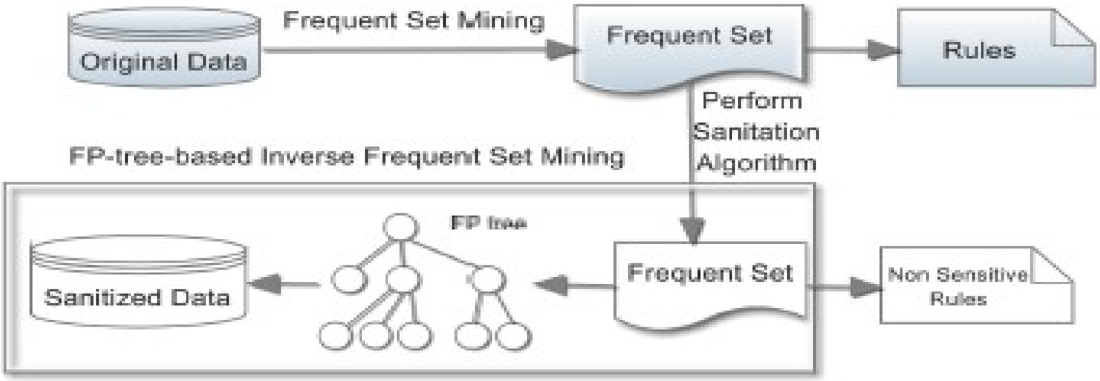


Figure-18: Reconstruction-based association rule hiding

PRIVACY-PRESERVING CLASSIFICATION

Classification is a form of data analysis that extracts models describing important data classes. Data classification can be seen as a two-step process. In the first step, which is called *learning* step, a classification algorithm is employed to build a *classifier* (classification model) by analyzing a training set made up of tuples and their associated class labels. In the second step, the classifier is used for classification, i.e. predicting categorical class labels of new data. Typical classification model include decision tree, Bayesian model, support vector machine, etc.

A: Decision Tree

A decision tree is a flowchart-like tree structure, where each internal node (non-leaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (or terminal node) represents a class label. Given a tuple, the attribute values of the tuple are tested against the decision tree. A path is traced from the root to a leaf node which holds the class prediction for the tuple. Decision trees can easily be converted to classification rules.

To realize privacy-preserving decision tree mining, Dowd propose a data perturbation technique based on *random substitutions*. Given a data tuple, the perturbation is done by replacing the value of an attribute by another value that is chosen randomly from the attribute domain according to a probabilistic model. They show that such perturbation is immune to *data-recovery attack* which aims at recovering the original data from the perturbed data, and *repeated-perturbation attack* where an adversary may repeatedly perturb the data with the hope to recover the original data. Brickell and Shmatikov present a cryptographically secure protocol for privacy-preserving construction of decision trees. The protocol takes place between a user and a server. The user's input consists of the parameters of the decision tree that he wishes to construct, such as which attributes are treated as features and which attribute represents the class. The server's input is a relational database. The user's protocol output is a decision tree constructed from the server's data, while the server learns nothing about the constructed tree. Fong introduce a perturbation and randomization based approach to protect the data sets utilized in decision tree mining. Before being released to a third party for decision tree construction, the original data sets are converted into a group of unreal data sets, from which the original data cannot be reconstructed without the entire group of unreal data sets. Meanwhile, an accurate decision tree can be built directly from the unreal data sets. Sheela and Vijayalakshmi propose a method based on *secure multi-party computation* (SMC) to build a privacy-preserving decision tree over vertically partitioned data. The proposed method utilizes Shamir's secret sharing algorithm to securely compute the cardinality of scalar product, which is needed when computing information gain of attributes during the construction of the decision tree.

B: Naïve Bayesian Classification

Naïve Bayesian classification is based on Bayes' theorem of posterior probability. It assumes that the effect of an attribute value on a given class is independent of the values of other attributes. Given a tuple, a Bayesian classifier can predict the probability that the tuple belongs to a particular class.

Vaidya study the privacy-preserving classification problem in a distributed scenario, where multi-parties collaborate to develop a classification model, but no one wants to disclose its data to others. Based on previous studies on secure multi-party computation, they propose different protocols to learn naïve Bayesian classification models from vertically partitioned or horizontally partitioned data. For horizontally partitioned data, all the attributes needed for classifying an instance are held by one site. Each party can directly get the classification result, therefore there is no need to hide the classification model. While for vertically partitioned data, since one party does not know all the attributes of the instance, he cannot learn the full model, which means sharing the classification model is required. In this case, protocols which can prevent the disclosure of sensitive information contained in the classification model (e.g. distributions of sensitive attributes) are desired. Skarkala also study the privacy-preserving classification problem for horizontally partitioned data. They propose a privacy-preserving version of the *tree augmented* naïve (TAN) Bayesian classifier to extract global information from horizontally partitioned data. Compared to classical naïve Bayesian classifier, TAN classifier can produce better classification results, since it removes the assumption about conditional independence of attribute. Different from above work, Vaidya consider a centralized scenario, where the data miner has centralized access to a data set. The miner would like to release a classifier on the premise that sensitive information about the original data owners cannot be inferred from the classification model. They utilize differential privacy model to construct a privacy-preserving Naïve Bayesian classifier. The basic idea is to derive the sensitivity for each attribute and to use the sensitivity to compute Laplacian noise. By adding noise to the parameters of the classifier, the data miner can get a classifier which is guaranteed to be differentially private.

C: Support Vector Machine

Support Vector Machine (SVM) is widely used in classification [1]. SVM uses a nonlinear mapping to transform the original training data into a higher dimension. Within this new dimension, SVM searches for a linear optimal separating hyperplane (i.e. a “decision boundary” separating tuples of one class from another), by using *support vectors* and *margins* (defined by the support vectors).

Vaidya propose a solution for constructing a global SVM classification model from data distributed at multiple parties, without disclosing the data of each party. They consider the *kernel matrix*, which is the central structure in a SVM, to be an intermediate profile that does not disclose any information on local data but can generate the global model. They propose a method based on gram matrix computation to securely compute the kernel matrix from the distributed data. Xia consider that the privacy threat of SVM-based classification comes from the support vectors in the learned classifier. The support vectors are intact instances taken from training data, hence the release of the SVM classifier may disclose sensitive information about the original owner of the training data. They develop a privacy-preserving SVM classifier based on hyperbolic tangent kernel. The kernel function in the classifier is an approximation of the original one. The degree of the approximation, which is determined by the number of support vectors, represents the level of privacy preserving. Lin and Chen also think the release of support vectors will violate individual's privacy. They design a privacy-preserving SVM classifier based on Gaussian kernel function. Privacy-preserving is realized by transforming the original decision function, which is determined by support vectors, to an infinite series of linear combinations of monomial feature mapped support vectors. The sensitive content of support vectors are destroyed by the linear combination, while the decision function can precisely approximate the original one. In above discussions we briefly reviewed the privacy-preserving approaches proposed for different classification models. To provide a clear view of these studies, we summarize the main points of some representative approaches in Table.

PRIVACY-PRESERVING CLUSTERING

Cluster analysis is the process of grouping a set of data objects into multiple groups or clusters so that objects within a cluster have high similarity, but are very dissimilar to objects in other clusters. Dissimilarities and similarities are assessed based on the attribute values describing the objects and often involve distance measures. Clustering methods can be categorized into partitioning methods, hierarchical methods, density-based methods, etc.

Current studies on privacy-preserving clustering can be roughly categorized into two types, namely approaches based on perturbation and approaches based on secure multi-party computation (SMC).Perturbation-based approach modifies the data before performing clustering.

Oliveira and Zaiane introduce a family of geometric data transformation methods for privacy-preserving clustering. The proposed transformation methods distort confidential data attributes by translation, scaling, or rotation, while general features for cluster analysis are preserved. Oliveira and Zaiane have demonstrated that the transformation methods can well balance privacy and effectiveness, where privacy is evaluated by computing the variance between actual and perturbed values, and effectiveness is evaluated by comparing the number of legitimate points grouped in the original and the distorted databases. The methods proposed in deal with numerical attributes, while in, Rajalaxmi and Natarajan propose a set of hybrid data transformations for categorical attributes. Recently, Lakshmi and Rani propose two hybrid methods to hide the sensitive numerical attributes. The methods utilize three different techniques, namely singular value decomposition (SVD), rotation data perturbation and independent component analysis. SVD can identify information that is not important for data mining, while ICA can identify those important information. Rotation data perturbation can retains the statistical properties of the data set. Compared to method solely based on perturbation, the hybrid methods can better protect sensitive data and retain the important information for cluster analysis.

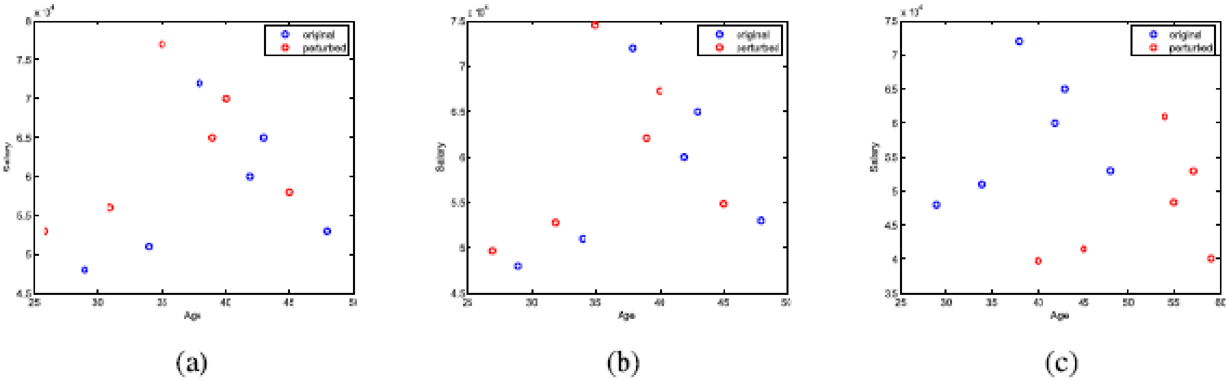


Figure-19: Examples of geometric data transformation.

A *checkThreshold* algorithm is proposed to determine whether the stopping criterion is met. Jha design a privacy-preserving k-means clustering algorithm for horizontally partitioned data, where only the cluster means at various steps of the algorithm are revealed to the participating parties. They present two protocols for privacy-preserving computation of cluster means. The first protocol is based on oblivious polynomial evaluation and the second one uses homomorphic encryption. Based on above studies, many privacy-preserving approaches have been developed for k-means clustering.

Different from previous studies which focus on -means clustering, De and Tripathy recently develop a secure algorithm for hierarchical clustering over vertically partitioned data. There are two parties involved in the computation. In the proposed algorithm, each party first computes clusters on their own private data set. Then, both parties compute the distance between each data point and each of the cluster centers. The resulting distance matrices along with the randomized cluster centers are exchanged between the two parties. Based on the information provided by the other party, each party can compute the final clustering result.

DECISION MAKER

A. Concerns of Decision Maker

The ultimate goal of data mining is to provide useful information to the decision maker, so that the decision maker can choose a better way to achieve his objective, such as increasing sales of products or making correct diagnoses of diseases. At a first glance, it seems that the decision maker has no responsibility for protecting privacy, since we usually interpret privacy as sensitive information about the original data owners (i.e. data providers).

However, if we look at the privacy issue from a wider perspective, we can see that the decision maker also has his own privacy concerns. The data mining results provided by the data miner are of high importance to the decision maker. If the results are disclosed to someone else, e.g. a competing company, the decision maker may suffer a loss. That is to say, from the perspective of decision maker, the data mining results are sensitive information. On the other hand, if the decision maker does not get the data mining results directly from the data miner, but from someone else which we called *information transmitter*, the decision maker should be skeptical about the credibility of the results, in case that the results have been distorted. Therefore, the privacy concerns of the decision maker are twofold: how to prevent unwanted disclosure of sensitive mining results, and how to evaluate the credibility of the received mining results.

B. Approaches to Privacy Protection

To deal with the first privacy issue proposed above, i.e. to prevent unwanted disclosure of sensitive mining results, usually the decision maker has to resort to legal measures. For example, making a contract with the data miner to forbid the miner from disclosing the mining results to a third party. To handle the second issue, i.e. to determine whether the received information can be trusted, the decision maker can utilize methodologies from data provenance, credibility analysis of web information, or other related research fields. In the rest part of this section, we will first briefly review the studies on data provenance and web information credibility, and then present a preliminary discussion about how these studies can help to analyze the credibility of data mining results.

1) Data Provenance

If the decision maker does not get the data mining results directly from the data miner, he would want to know how the results are delivered to him and what kind of modification may have been applied to the results, so that he can determine whether the results can be trusted. This is why “provenance” is needed. The term *provenance* originally refers to the chronology of the ownership, custody or location of a historical object. In information science, a piece of data is treated as the historical object, and *data provenance* refers to the information that helps determine the derivation history of the data, starting from the original source. Two kinds of information can be found in the provenance of the data: the ancestral data from which current data evolved, and the transformations applied to ancestral data that helped to produce current data. With such information, people can better understand the data and judge the credibility of the data.

Since 1990s, data provenance has been extensively studied in the fields of databases and workflows. Several surveys are now available. Simmhan present a taxonomy of data provenance techniques. The following five aspects are used to capture the characteristics of a provenance system:

- Application of provenance. Provenance systems may be constructed to support a number of uses, such as estimate data quality and data reliability, trace the audit trail of data, repeat the derivation of data, etc.
- Subject of provenance. Provenance information can be collected about different resources present in the data processing system and at various levels of detail.
- Representation of provenance. There are mainly two types of methods to represent provenance information, one is annotation and the other is inversion. The annotation method uses metadata, which comprise of the derivation history of the data, as

annotations and descriptions about sources data and processes. The inversion method uses the property by which some derivations can be inverted to find the input data supplied to derive the output data.

- Provenance storage. Provenance can be tightly coupled to the data it describes and located in the same data storage system or even be embedded within the data file. Alternatively, provenance can be stored separately with other metadata or simply by itself.
- Provenance dissemination. A provenance system can use different ways to disseminate the provenance information, such as providing a derivation graph that users can browse and inspect.

2) Web Information Credibility

Because of the lack of publishing barriers, the low cost of dissemination, and the lax control of quality, credibility of web information has become a serious issue. Tudjman identify the following five criteria that can be employed by Internet users to differentiate false information from the truth:

- Authority: the real author of false information is usually unclear.
- Accuracy: false information dose not contain accurate data or approved facts.
- Objectivity: false information is often prejudicial.
- Currency: for false information, the data about its source, time and place of its origin is incomplete, out of date, or missing.
- Coverage: false information usually contains no effective links to other information online.

Metzger summarizes the skills that can help users to assess the credibility of online information. With the rapid growth of online social media, false information breeds more easily and spreads more widely than before, which further increases the difficulty of judging information credibility. Identifying rumors and their sources in micro blogging networks has recently become a hot research topic. Current research usually treats rumor identification as a classification problem, thus the following two issues are involved:

- Preparation of training data set. Current studies usually take rumors that have been confirmed by authorities as positive training samples. Considering the huge amount of messages in micro blogging networks, such training samples are far from enough to train a good classifier. Building a large benchmark data set of rumors is in urgent need.
- Feature selection. Various kinds of features can be used to characterize the micro blogging messages. In current literature, the following three types of features are often used: content-based features, such as word unigram/bigram, part-of-speech unigram/bigram, text length, number of sentiment word (positive/negative), number of URL, and number of hash tag; user-related features, such as registration time, registration location, number of friends, number of followers, and number of messages posted by the user; network features, such as number of comments and number of re-tweets.

So far, it is still quite difficult to automatically identifying false information on the Internet. It is necessary to incorporate methodologies from multiple disciplines, such as nature language processing, data mining, machine learning, social networking analysis, and information provenance, into the identification procedure.

IMPLEMENTATION

Privacy preserving in user profiling in GSM data

In this section we study the privacy guarantees of the knowledge discovery process and we show that it can be made in a privacy by design manner by applying some small change to the process that will not affect the final result of the analysis. An analytical process for user profiling in GSM data; in other words, the proposed methodology identifies a partition of the users tracked by GSM phone calls into profiles like *resident*, *commuters* and *visitors* and quantifies the percentage of the different profiles. The profiling methodology is based on an machine learning step using SOM applied to spatio-temporal user profiles extracted from people call habits. In particular, the whole analytical process is composed of the following steps:

1 Select from the whole network the cells overlapping the area to which we are interested for the analysis (see Figure (left) as an example);

2 Build a time projection by two temporal operations (Figure (right)): (a) the aggregation of the days in weekdays and weekend slots; (b) the splitting of each slot in time intervals representing interesting time windows during the day;

3. Construct for each user the Space Constrained Temporal Profile by using the CDR logs according to the space constraints and the time Projection A SCT profile *P* is an aggregation of call statistics according to a given temporal discretization where only the calls performed in the cells, contained within the a certain area, are considered. In particular, each profile *P* is a matrix and each position *Pij* contains the value *v* that corresponds to the number of days with at least one call from the user in the area of interest during the set of days *j* and the time slot *i*. As an example, in Figure 8(right), *P*2.1 = 4 means that the user visited the area of interest 4 days during

the weekdays of the first week and always in the time interval 08:00:00-18:59:59]. In the following we denote by *P* the set of SCT profiles extracted from CDR logs.

4. The set of SCT profiles, that are a concise representation of the users' behaviors measured by their calls, is then processed by using the SOM algorithm in order to extract the typical global profiles.The SOM output is a set of nodes representing groups of users with similar temporal profiles; therefore, counting the instances in each group, it is possible to estimate the percentage of residents, commuters and visitors.

State-of-the-art on privacy in GSM data

Relatively, little work has addressed privacy issues in the publication and analysis of GSM data. In the literature, many works that treat GSM data state that in this context there is no privacy issue or at least the privacy problems are mitigated by the granularity of the cell phone. However, recently Golle and Partridge showed that a fraction of the US working population can be uniquely identified by their *home* and *work* locations even when those locations are not known at a fine scale or granularity. Given that the locations most frequently visited by a mobile user often correspond to the home and work places, the risk in releasing locations traces of mobile phone users appears very high. Privacy risks even in case of releasing of location information with not fine granularity. We consider the 'top *N*' locations visited by each user instead of the simple home and work. The basic idea of this work is that more generally the number *N* of top preferential locations determines the power of an adversary and the safety of a user's privacy. Therefore, we can say that more top locations an adversary knows about a user, the higher is the probability to re-identify that user. The fewer top locations a user has, the safer they are in terms of privacy presents a study on 30 billion CDRs from a nationwide cellular service provider in the United States with location information for about 25 million mobile phone users on a period of three months. The study highlights important factors that can have a relevant impact on the anonymity. Examples are the value of *N* in finding the top *N* locations, the granularity level of the released locations, the fact that the top locations are sorted or not, the availability of additional social information about users, and geographical regions. The outcomes of this study is that the publication of anonymized location data in its original format, i.e. at the sector level or cell level, put at risk the user privacy because a significant fraction of users can be re-identified from the anonymized data. Moreover, it was shown that different geographical areas have different levels of privacy risks, and at a different granularity level this risk may be higher or lower than other areas. When the spatial granularity level of the cell data is combined with time information and a unique handset identifier, all this information can be used to track people movements. This requires that a good privacy-preserving technique has to be applied when analysis such data. Unfortunately, do not consider this aspect. The studies user re-identification risks in GSM networks in the case user historical data is available to characterize the mobile users *a priori*.

Attack model and privacy by design solution

Given the above overview about the methodology for extracting global profiles and for computing a quantification of the different kinds of global profiles, now we analyze the privacy risks of the users.

We can identify three main phases in this process:

(a) The extraction of the SCT profile for each user;

(b) The extraction of global profiles; and

(c) The quantification of different kinds of global profiles.

It is immediate to understand that the publication of the final result, i.e., the quantification of the global profiles cannot put at risk the individual privacy of any user because this information is a simple aggregation that does not contain any sensitive information about the single users. This means that an attacker by accessing this kind of data cannot infer any information about a user. The first phase instead is more problematic for the individual privacy of users because requires to access the CDR data that contains all information about the user calls. In particular, for each user call we have the identifiers of the cell where the call starts and ends respectively and the date and time when the call starts, and its duration. The positional accuracy of cells is few hundred meters in a city and when this information is combined with the time information all this information can help to track people movements. We studied the user re-identification risks in GSM networks and showed that it is possible to identify a mobile user from CDR records and a pre-existing location profile, based on previous movements. In particular, one of the re-identification methods that they propose allows for the identification of around 80% of users. As a consequence, this kind of data can reveal sensitive user behavior and the telecommunication operator cannot release this data to the analyst without any privacy-preserving data transformation. However, we observe that the only information that the analyst needs for computing the global profiles and their quantification is the set of SCT profiles; therefore, we propose an architecture where, the telecommunication operator computes the SCT profiles and then sends them to the analyst for the computation of the step (b) and (c). This solution avoids the access to the CDR logs for the analyst while provides to him the minimum information to performing the target analysis with correctness. Now the question is: *Can an attacker infer private information about a user by accessing the set of SCT profiles? Is this form of data enough for protecting the individual privacy of each user in the system?* If the answer to this last question is yes, we could have both individual privacy protection and perfect quality of the analytical results. First of all, we observe that a SCT profile can be seen as a spatio-temporal generalization of the CDR data of a user. Clearly, this form of data is more aggregated w.r.t. the CDR logs because it cannot reveal the history of the user movements, the number of calls and the exact day and time of each call. Moreover, this profile is constructed by considering a specific area such as a city therefore, it is impossible to infer where exactly the user went with a finer granularity. The only information that he can infer is that a specific user visited the city in a specific aggregated period. As an example, an attacker could understand that a given user went to Pisa during a specific week-end if the profiles that he was accessing are related to people in Pisa. However, in the following we identify two possible attack models, based on the *linking attack*, that use two different background knowledge. Then, we simulate this two attacks on real-world data for showing the privacy protection provided by our schema.

Background knowledge 1. We assume that the attacker knows a set of locations visited by a user U where he called someone and the time of these calls. This means that he can build a SCT profile PB with this background knowledge, where $PBij = -1$ if the attacker does not have any information about the call activity of the user in the period (i, j) while $PBij = v$, with $v > 0$, if fromthe background knowledge he derives that the user was presen tin the area v times in the period (i, j).

Attack model 1. The attacker, who gains access to the set of SCT profiles, uses the background knowledge PB on the user U to match all the profiles that include PB. The setof matched profiles is the set $C = \{P \in P | \forall PBij \geq 0, PBij \leq Pij\}$. The probability of reidentification of the user U is □$1/|C|$. Clearly, a greater number of candidates corresponds to a more privacy protection.

***Background knowledge* 2**. In our study we also consider a different background knowledge.

We assume that the attacker for some time periods (i, j) knows the exact number of times that a user U visited locations in the area of interest. This means that he can build a profile PB with this background knowledge, where $PBij = -1$ if the attacker does not have any information about the presence of the user U in the area of interest during the period (i, j) while $PBij = v$ with $v \geq 0$, if from the background knowledge he derives that the user was present in the area v times in the period (i, j). As an example, suppose an adversary knows that during the first week Mr. Smith went to Pisa in the time interval [08:00:00-18:59:59] only 4 times over 5 because Friday he was sick, Then, from this information he can construct a profile PB where $PB21 = 4$ while the other entries are equal to –1. Note, that in this case the attacker does not know if the user U did a call during his presence in the area of interest of the analysis and this implies that the malicious part does not know if the user U is represented in the set of profiles.

***Attack model* 2**. The attacker, who gains access to the set of SCT profiles uses thebackground knowledge PB on the user U to select the set of candidate profiles $C = \{P \in P | \forall PBij \geq □, Pi,j \leq PBij\}$. The re-identification probability of the user$U$ is $1/|C| \times Prob$, where $Prob$ is the probability that one of the profiles in P belongs to user U.

Privacy protection analysis

We performed a series of experiments on a real GSM dataset. We obtained a dataset of CDR logs in the Province of Pisa during the period fromJanuary 9th to February 8th 2012 reporting the activities of around 232k persons, for a total of 7.8M call records. Focusing on the urban area of the city of Pisa, we extracted the SCT profiles for the 63k users performing at least one call activity in the observation period. We then simulated two attacks according to the two attacking models above and measured the re-identification probability of each SCT profile.The simulation is performed as follows: we generate a series of profiles PB according to the background knowledge. These profiles are derived from the real user SCT profiles in the dataset. Then, we have performed the attack 1(attack 2) on the set of profiles P. Concerning the attack 2 we have assumed that the adversary knows the exact number of times that the user visited locations in Pisa for each period (i, j), i.e., for all the 4 weeks in the profiles. Figure 9 shows the cumulative distribution of the re-identification probability. We found that in the worst case the probability of re-identification is 0.027% and only about 5% of users in the set of SCT profiles have this level of risk, while the other users have a lower risk of privacy violation. This very high protection is due to the fact that with the background knowledge 2 (BK2) the attacker is not sure that the user is in the set of profiles that he is observing. However, even if we assume that he knows that the specific user is represented in the set of SCT profiles, the probability of re-identification is always low. We indeed have observed that the highest probability of re-identification in this case is 0.21% Concerning the attack 1, that is based on a stronger background knowledge, we have assumed that the attacker knows the user call activities for a specific number of weeks and we have measured the probability to re-identify the user and infer his activities in the remaining weeks. Figure 10 shows the cumulative distribution of the re-identification probability for different levels of background knowledge: 1 week, 2 weeks and 3 weeks. As expected, when we increase the periods of observations of the adversary we have a worst privacy protection. However, when the attacker knows 1 week or 2 weeks of call activities of a specific users the probability of re-identification is always no more than 20% and 25% and this happens for about 0.01% of user in the profile data; 99.99% of users has a lower privacy risk. When we consider a observation period of 3 weeks the privacy protection decreases and for less than 0.1% of users the probability of re-identification is 50%, while for more than 99.9% of people the probability of re-identification is no more than 32%. Moreover, the 99% of users has a risk of re-identification less than about 7%. Clearly, here it is important to note that the background knowledge that we are taking into consideration is very strong. We have also measured the risk of re-identification assuming that the attacker knows the user call activities of different periods of the SCT profile. This kind of attack is similar are enough to uniquely identify 95% of the individuals. In our experiments by using the SCT profiles instead of CDR logs, we have found that with 10 observations the probability of re-identification is less than 20% for all the users and about 99% of people has a risk of re-identification of about 1%. While if we consider 20 observations the situation is very similar to the case in which the attacker knows 3weeks of calls of user in Figure 10. The conclusion is that the illustrated process shows as by knowing the analysis to be performed on the data it is possible to transform the original data in a different form (by aggregations) and find a representation that both contains all the proprieties useful for obtaining a perfect analytical result and preserves the user privacy.

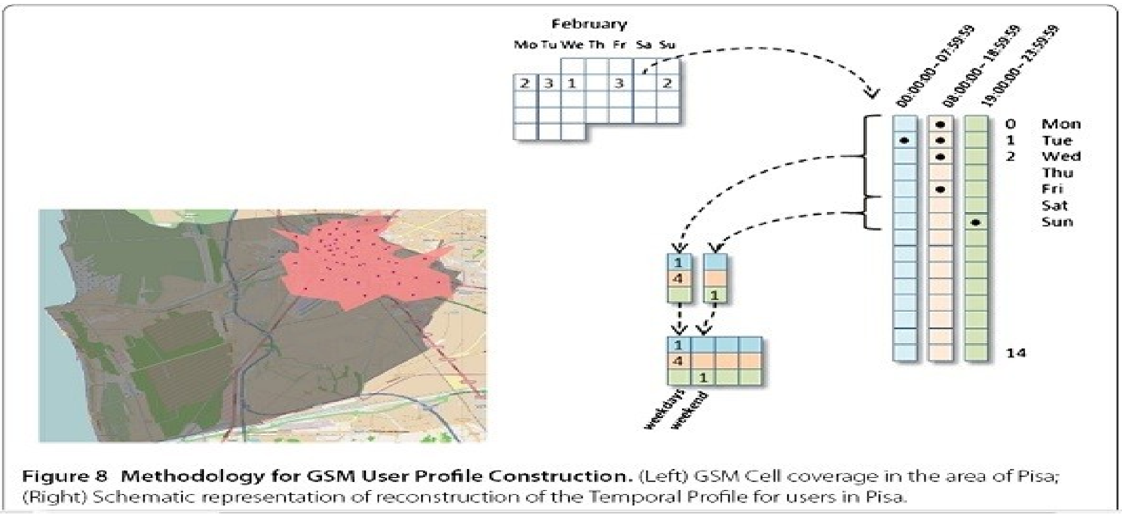


Figure 8 Methodology for GSM User Profile Construction. (Left) GSM Cell coverage in the area of Pisa; (Right) Schematic representation of reconstruction of the Temporal Profile for users in Pisa.

Figure-20: Methodology for GSM User Profile Construction

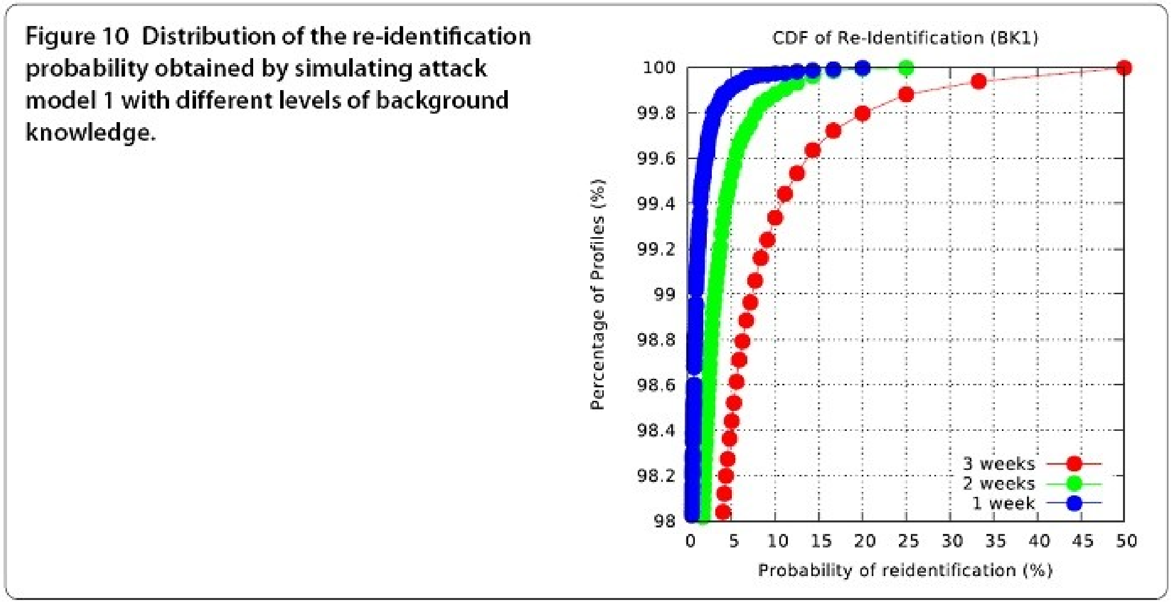


Figure 10 Distribution of the re-identification probability obtained by simulating attack model 1 with different levels of background knowledge.

Figure-21:Distribution of the re-identification probability obtained by simulating attack model with different level of background knowledge

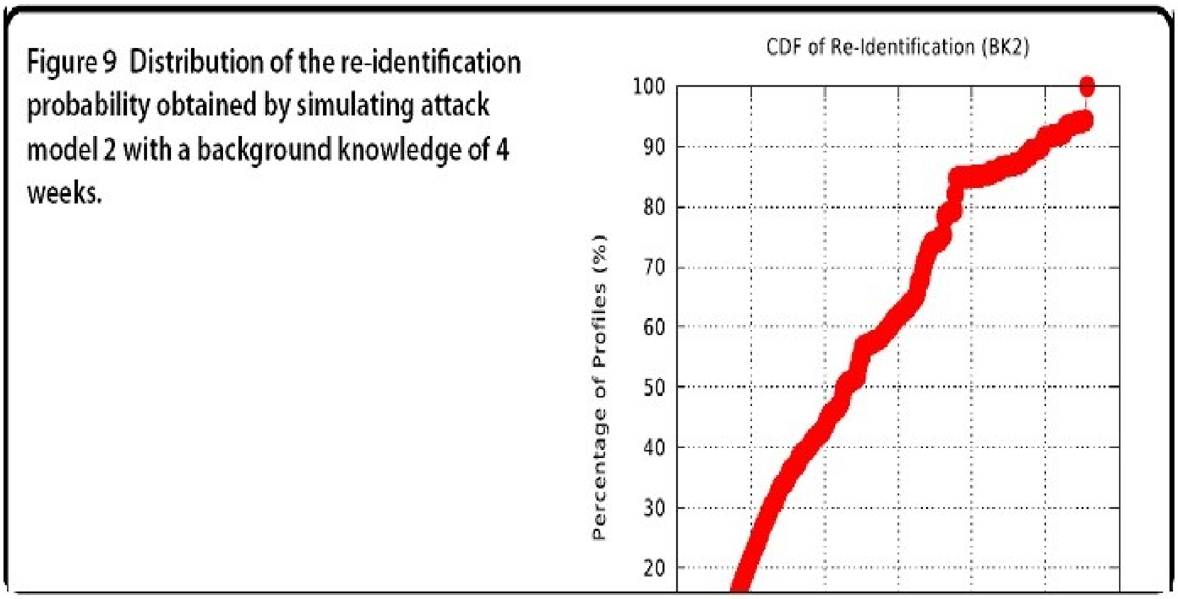


Figure 9 Distribution of the re-identification probability obtained by simulating attack model 2 with a background knowledge of 4 weeks.

Figure-22:Distribution of the re-identification probability obtained by simulating attack model 2 with a background knowledge of 4 weeks.

Algorithms and frame works

SMC problem is the problem of n parties to compute a private function of their inputs in a secure method, where security means the correct result computed by the TTPs for maintaining the privacy of the parties as some of the parties may want to misuse the other party's data. We assume that the inputs are *x1,x2,…xn* where xi is the data of party Pi and the TTP will compute a function f(x1,x2,….xn)=y and announce the result y [1]. Security is meant to achieve correctness of the result of computation and keeping the party's input private even if some of the parties are corrupted. In figure 1, trusted third party is used for doing the computation on the inputs provided by the parties. According to [2], the major problem with this approach is that it is difficult to find the third party which is trusted by all the parties providing the inputs and to control the function of adversaries.

Aiming at privacy preserving computing of statistical distribution, which is Frequently encountered in statistics, and based on the intractability of computing discrete logarithm and using rigorous logic, they proposed the solution. [17] Presented the protocols allowing the players to securely solve standard computational problems in linear algebra such as determinant of matrices product, rank of a matrix, and determine similarity between matrices. [18] Presented TASTY, a novel tool for automating, i.e., describing, generating, executing, benchmarking, and comparing, efficient secure two-party computation protocols. They used

TASTY to compare protocols for secure multiplication based on homomorphic encryption with those based on garbled circuits and highly efficient multiplication. [19] Presented a hybrid-secure MPC protocol that provides an optimal trade-off between IT robustness and computational privacy. [20] Presented a solution to the Secure Multi-party Computation (SMC) problem in the form of a protocol that ensures zero-hacking. The solution comprises of a protocol with several trusted third parties (TTPs).The protocol selects one TTP among all TTPs in the SMC architecture that owns the responsibility of all the computation in the system. This TTP is called the master TTP and it is different at different times. The procedure of selecting master TTP could be non-deterministic but it is made deterministic by randomization technique. This ensures that no single TTP controls the entire system all the times. At the same time, this also ensures that no TTP knows where the computation is taking place. This approach is having merit over the other one where only one TTP is given the responsibility to hold entire data of the system.

Gaps in [20] are multiple TTPs are given the input but computation is performed by master TTP only. So even if this protocol defines multi TTP environment, but efficiency of TTP in the protocol is not utilized properly. The second gap in [20] is introducing packet layer. The responsibility of packet layer could be handled by parties itself and a virtual party can be used to make identity of party ambiguous. In our multi TTP computation protocol, same computation is performed by multiple TTPs selected at runtime and majority giving the same identical result is considered the right result of computation. Efficient SMC_Multi TTP algorithm designing was our previous work.

PROPOSED WORK

Informal description of the protocol

In this protocol all the hospitals involved in computation split their data into x packets and encrypt data through some pre-decided encryption method. The encrypted data Eij is send to inscrutablizers. This is an untrusted layer whose task is to forward the packets to TTPs selected at runtime for computation. Inscrutablizers cannot store the data, they just forward it. As inscrutablizers are untrusted, so they hold packets of the parties and not the entire data. After computation majority of TTPs giving the same result is considered as the right result of computation as correctness is a major parameter for computation which has been analyzed in previous work.

3 layer architectural framework:

1. n hospitals : H1, H2…Hn with data packets xij
2. Inscrutable layer
3. Multiple TTP layer: TTP1, TTP2…TTPn

Formal description

Algorithm: SMC_Split Multi TTP computation

Data Structure

Hi – Hospitals where i ranges from 1 to *n*

xij – Data of party *Hi* where *j* ranges from 1 to *x*

Rij – Random data of party *Hi* where *j* ranges from 1 to *q*

Dij – total data including the random and the original data

Eij – Encrypted data associated with party *Hi* where *j* ranges from 1 to *x+q*

ISp– untrusted inscrutablizers, where *p* ranges from 1 to *z*

TTP – third party.

Algorithm:

- Generate *xij* packets for every party *Hi*
- Generate random data *Rij* for every xij
- Group random data *Rij* with original data *xij* to get *Dij*
- Encrypt data *Dij* using pre-decided encryption method to get *Eij*.
- Distribute the encrypted data *Eij* among the inscrutablizer *Ap*
- Send the data from un-trusted inscrutablizer *Ap* to *TTPs*
- Calculate the result at *TTPs* using the encrypted data and the keys.
- Identify the TTPs at runtime for performing computation.
- The result is announced by TTPs
- Majority of TTPs giving same identical result is considered as correct result.

4 layer architectural framework:

1. n hospitals : H1, H2…Hn with data packets xij
2. Untrusted Inscrutable layer (Virtual Party)
3. Trusted Inscrutable layer (Packets are distributed))
4. Multiple TTP layer: TTP1, TTP2…TTPn

The advantage of designing four layer architectural frameworks is to increase the security level of inputs provided by the hospitals. In this framework an untrusted inscrutable layer is added to hide the identities of the hospitals. This layer is inscrutablizers' layer. The data from this layer is then send to trusted inscrutablizers who does not have any knowledge about input of the hospitals as the data arrives from virtual layer. In this protocol all the hospitals involved in computation split their data into x packets and encrypt data through some pre-decided encryption method. The encrypted data Eij is send to untrusted inscrutablizers. This is an untrusted layer whose task is to forward the packets to trusted inscrutablizers and then they forward packets to TTPs selected at runtime for computation. Inscrutablizers cannot store the data. Inscrutablizers hold packets of the parties and not the entire data for security and privacy of inputs. After computation majority of TTPs giving the same result is considered as the right result of computation as correctness is a major parameter for computation which has been analyzed in previous work.

Security Analysis

If the TTP is malicious then it can reveal the identity of the source of data. A set of inscrutablizers from the inscrutable layer will make the source of data ambiguous and will preserve the privacy of individual. The more the number of inscrutablizers in the inscrutable layer the less will be the possibility of hacking the privacy of the data. The inscrutablizers hide the identity of the bank. In the protocol there is one layer of inscrutablizers, consisting of p inscrutablizers $IS1, IS2, IS3\ldots, ISp$ Then the probability of revealing the source of the data at TTP is inversely proportional to the number of parties sending data. We can see that there is more security when there are large numbers of participants. The probability of hacking the data of a single hospital Hi:P_Hi_ = 1/n (1)

Where n is total number of hospitals involved in computation.

The probability of hacking data of r hospitals: P(Hr)=r/n (2)

Therefore, total Probability for leak of the packets = [r/n] * [r=1_r Xr)/ (r=1_n Xr] (3)

Where Xr are the packets of r hospitals.

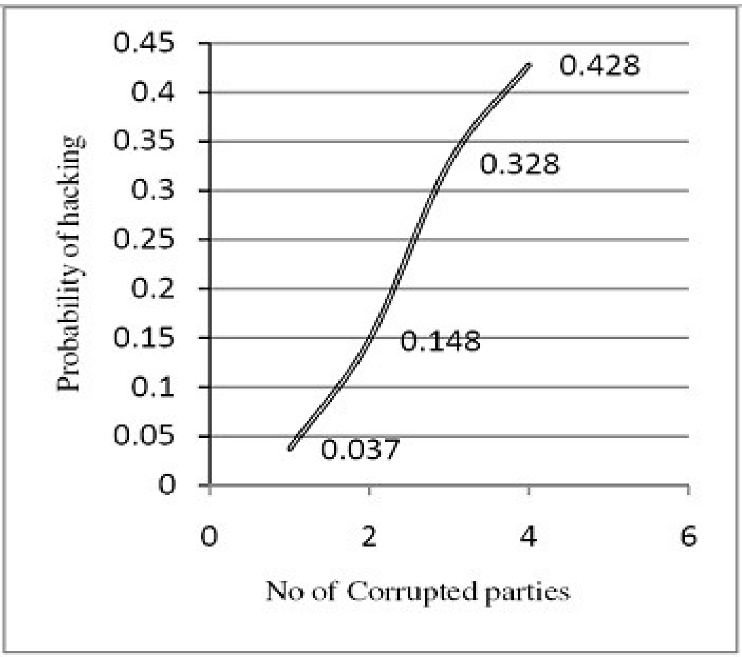


Figure 5. Security analysis with increased number of corrupted parties

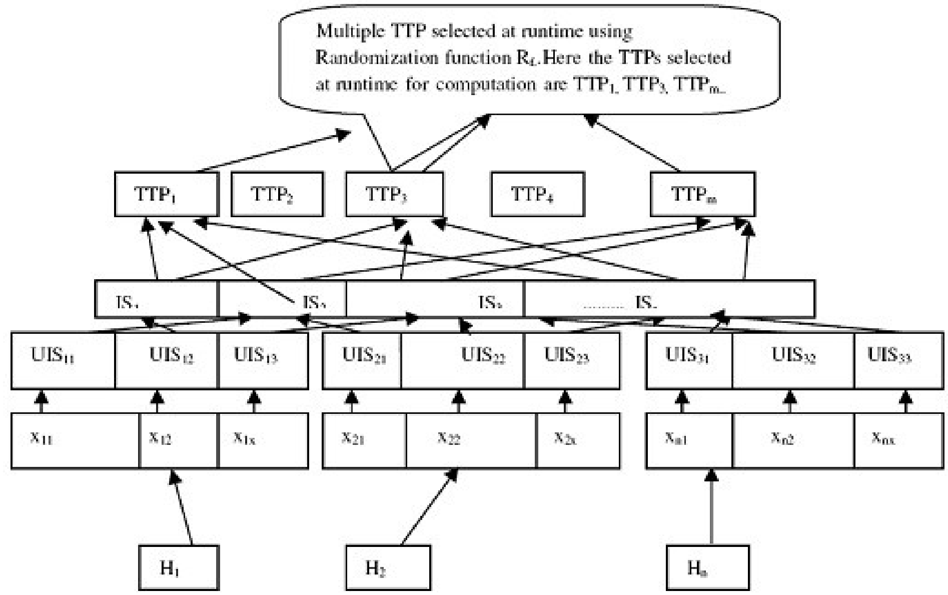

Multiple TTP selected at runtime using
Randomization function RL.Here the TTPs selected
at runtime for computation are TTP1, TTP3, TTPm.
TTP1
TTP2
TTP3
TTP4
TTPm
IS1
IS2
IS3
.... ISp
UIS11
UIS12
UIS13
UIS21
UIS22
UIS23
UIS31
UIS32
UIS33
x11
x12
x1x
x21
x22
x2x
xn1
xn2
xnx
H1
H2
Hn

Figure 4. Four Layer architectural SMC framework for hospitals using Multi TTP computation

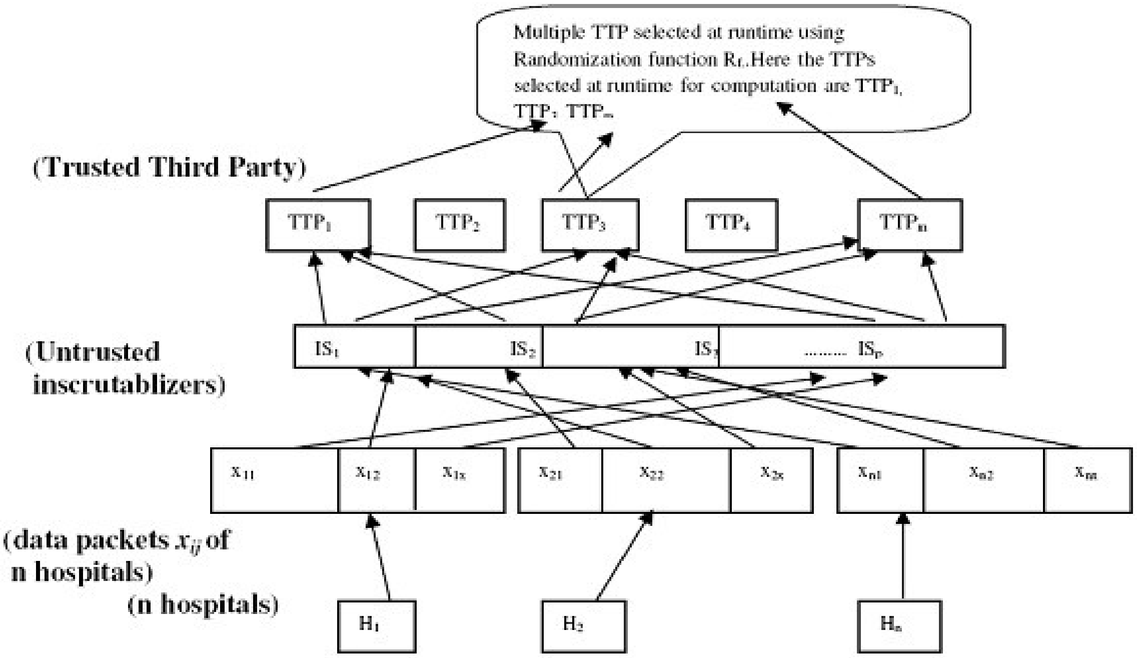


Figure 2. Three layer architectural SMC framework for hospitals using Multi TTP computation

5.2 Metrics to be calculated

Metrics for Quantifying Hiding Failure

The percentage of sensitive information that is still discovered, after the data has been sanitized, gives an estimate of the *hiding failure* parameter. Most of the developed privacy preserving algorithms are designed with the goal of obtaining zero hiding failure. Thus, they hide all the patterns considered sensitive. However, it is well known that the more sensitive information we hide, the more non-sensitive information we miss. Thus, some PPDM algorithms have been recently developed which allow one to choose the amount of sensitive data that should be hidden in order to find a balance between privacy and knowledge discovery. For example, Oliveira and Zaiane define the *hiding failure* (HF) as the percentage of restrictive patterns that are discovered from the sanitized database. It is measured as follows:

$$HF = \frac{\#R_P(D')}{\#R_P(D)}$$

------ [5.2.1]

where #*RP*(*D*) and #*RP*(*D*′) denote the number of restrictive patterns discovered from the original data base *D* and the sanitized database *D*′ respectively. Ideally, *HF*should be 0. In their framework, they give a specification of a *disclosure threshold* ϕ, representing the percentage of sensitive transactions that are not sanitized, which allows one to find a balance between the hiding failure and the number of misses. Note that ϕ does not control the *hiding failure* directly, but indirectly by controlling the proportion of sensitive transactions to be sanitized for each restrictive pattern. Moreover, it is important not to forget that intruders and data terrorists will try to compromise information by using various data mining algorithms. Therefore, a PPDM algorithm developed against a particular data mining techniques that assures privacy of information, may not attain similar protection against all possible data mining algorithms. In order to provide for a complete evaluation of a PPDM algorithm, we need to measure its hiding failure against data mining techniques which are different from the technique that the PPDM algorithm has been designed for. The evaluation needs the consideration of a class of data mining algorithms which are significant for our test. Alternatively, a formal framework can be developed that upon testing of a PPDM algorithm against pre-selected data sets, we can transitively prove privacy assurance for the whole class of PPDM algorithms.

Metrics for Quantifying Data Quality

The main feature of the most PPDM algorithms is that they usually modify the database through insertion of false information or through the blocking of data values in order to hide sensitive information. Such perturbation techniques cause the decrease of the data quality. It is obvious that the more the changes are made to the database, the less the database reflects the domain of interest. Therefore, data quality metrics are very important in the evaluation of PPDM techniques. Since the data is often sold for making profit, or shared with others in the hope of leading to innovation, data quality should have an acceptable level according also to the intended data usage. If data quality is too degraded, the released database is useless for the purpose of knowledge extraction. In existing works, several data quality metrics have been proposed that are either generic or data-use-specific. However, currently, there is no metric that is widely accepted by the research community. Here we try to identify a set of possible measures that can be used to evaluate different aspects of data quality. In evaluating the data quality metrics are very important in the evaluation of PPDM techniques. Since the data is often sold for making profit, or shared with others in the hope of leading to innovation, data quality should have an acceptable level according also to the intended data usage. If data quality is too degraded, the released database is useless for the purpose of knowledge extraction.

In existing works, several data quality metrics have been proposed that are either generic or data-use-specific. However, currently, there is no metric that is widely accepted by the research community. Here we try to identify a set of possible measures that can be used to evaluate different aspects of data quality. In evaluating the data quality after the privacy preserving process, it can be useful to assess both the *quality of the data* resulting from the PPDM process and the *quality of the data mining results*. The quality of the data themselves can be considered as a general measure evaluating the state of the individual items contained in the database after the enforcement of a privacy preserving technique. The quality of the data mining results evaluates the alteration in the information that is extracted from the database after the privacy preservation process, on the basis of the intended data use.

Quality of the Data Resulting from the PPDM Process

The main problem with data quality is that its evaluation is relative, in that it usually depends on the context in which data are used. In particular, there are some aspects related to data quality evaluation that are heavily related not only with the PPDM algorithm, but also with the structure of the database, and with the meaning and relevance of the information stored in the database with respect to a well-defined context. In the scientific literature data quality is generally considered a multi-dimensional concept that in certain contexts involves both objective and subjective parameters [3, 34]. Among the various possible parameters, the following ones are usually considered the most relevant:

- *Accuracy*: it measures the proximity of a sanitized value to the original value.
- *Completeness*: it evaluates the degree of missed data in the sanitized database.
- *Consistency*: it is related to the internal constraints, that is, the relationships that must hold among different fields of a data item or among data items in a database.

The accuracy is closely related to the *information loss* resulting from the hiding strategy: the less is the information loss, the better is the data quality. This measure largely depends on the specific class of PPDM algorithms. In what follows, we discuss how different approaches measure the accuracy. As for heuristic-based techniques, we distinguish the following cases based on the modification technique that is performed for the hiding process. If the algorithm adopts a perturbation or a blocking technique to hide both raw and aggregated data, the information loss can be measured in terms of the dissimilarity between the original dataset *D* and the sanitized one *D*′. Oliveira and Zaiane propose three different methods to measure the *dissimilarity* between the original and sanitized databases. The first method is based on the difference between the frequency histograms of the original and the sanitized databases. The second method is based on computing the difference between the sizes of the sanitized database and the original one. The third method is based on a comparison between the contents of two databases. A more detailed analysis on the definition of dissimilarity is presented by Bertino. They suggest to use the following formula in the case of transactional dataset perturbation:

$$Diss(D, D') = \frac{\sum_{i=1}^{n} | f_D(i) - f_{D'}(i)|}{\sum_{i=1}^{n} f_D(i)}$$

------- [5.2.2]

where *i* is a data item in the original database *D* and *fD*(*i*) is its frequency within the database, whereas *i'* is the given data item after the application of a privacy preservation technique and *fD*′ (*i*) is its new frequency within the transformed database *D*′. As we can see, the information loss is defined as the ratio between the sum of the absolute errors made in computing the frequencies of the items from a sanitized database and the sum of all the frequencies of items in the original database. The formula 5 can also be used for the PPDM algorithms which adopt a blocking technique for inserting into the dataset uncertainty about some sensitive data items or their correlations. The frequency of the item *i* belonging to the sanitized dataset *D′* is then given by the mean value between the minimum frequency of the data item *i*, computed by considering all the blocking values associated with it equal to zero, and the maximum frequency, obtained by considering all the question marks equal to one.

In case of data swapping, the information loss caused by an heuristic-based Algorithm can be evaluated by a parameter measuring the *data confusion* introduced by the value swappings. If there is no correlation among the different database records, the *data confusion* can be estimated by the percentage of value replacements executed in order to hide specific information. For the multiplicative-noise-based approaches, the quality of the perturbed data depends on the size of the random projection matrix. In general, the error bound of the inner product matrix produce by this perturbation technique is 0 on average and the variance is bounded by the inverse of the dimensionality of the reduced space. In other words, when the dimensionality of the random projection matrix is close to that of the original data, the result of computing the inner product matrix based on the transformed or projected data is also close to the actual value. Since inner product is closely related to many distance-based metrics (e.g., Euclidean distance, cosine angle of two vectors, correlation coefficient of two vectors, etc), the analysis on error bound has direct impact on the mining results if these data mining tasks adopt certain distance-based metrics. If the data modification consists of aggregating some data values, the information loss is given by the loss of detail in the data. Intuitively, in this case, in order to perform the hiding operation, the PPDM algorithms use some type of “Generalization or Aggregation Scheme” that can be ideally modeled as a tree scheme. Each cell modification applied during the sanitization phase using the Generalization tree introduces a data perturbation that reduces the general accuracy of the database. As in the case of the *k*-anonymity algorithm, we can use the following formula. Given a database *T* with *NA* fields and *N* transactions, if we identify as generalization scheme a domain generalization hierarchy *GT* with a depth *h*, it is possible to measure the *information loss* (*IL*) of a sanitized database *T*∗ as:

$$IL(T^*) = \frac{\sum_{i=1}^{i=N_A} \sum_{j=1}^{i=N} \frac{h}{|GT_{Ai}|}}{|T| * |N_A|} \qquad \text{----------- [5.2.3]}$$

where *h/ |GTAi|* represent the detail loss for each cell sanitized. For hiding techniques based on sampling approach, the quality is obviously related to the size of the considered sample and, more generally, on its features. There are some other precision metrics specifically designed for k-anonymization approaches. One of the earliest data quality metrics is based on the height of generalization hierarchies [25]. The height is the number of times the original data value has been generalized. This metric assumes that a generalization on the data represents an information loss on the original data value. Therefore, data should be generalized as fewer steps as possible to preserve maximum utility. However, this metric does not take into account that not every generalization steps are equal in the sense of information loss. Later, Iyengar [13] proposes a general *loss metric* (*LM*). Suppose *T* is a data table with *n* attributes. The *LM* metric is thought as the average information loss of all data cells of a given dataset, defined as follows:

$$LM(T^*) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{|T|} \frac{f(T^*[i][j])-1}{g(A_i)-1}}{|T| \cdot n}$$ --------- [5.2.4]

In equation 7, *T*∗ is the anonymized table of *T*, *f* is a function that given a data cell value *T*∗[*i*][*j*], returns the number of distinct values that can be generalized to *T*∗[*i*][*j*], and *g* is a function that given an attribute *Ai*, returns the number of distinct values of *Ai*.

The next metric, *classification metric* (*CM*), is introduced by Iyengar to optimize a *k*-anonymous dataset for training a classifier. It is defined as the sum of the individual penalties for each row in the table normalized by the total number of rows *N*.

$$CM(T^*) = \frac{\sum_{all\ rows} penalty(row\ r)}{N}$$ ---------- [5.2.5]

The penalty value of row *r* is 1, i.e., row *r* is penalized, if it is suppressed or if its class label is not the majority class label of its group. Otherwise, the penalty value of row *r* is 0. This metric is particularly useful when we want to build a classifier over anonymous data. Another interesting metric is the *discernibility metric*(*DM*) proposed by Bayado and Agrawal [4]. This discernibility metric assigns a penalty to each tuple based on how many tuples in the transformed dataset are indistinguishable from it. Let *t* be a tuple from the original table *T*, and let *GT*∗ (*t*) be the set of tuples in an anonymized table *T*∗ indistinguishable from *t* or the set of tuples in *T*∗ equivalent to the anonymized value of *t*. Then *DM* is defined as follows:

$$DM(T^*) = \sum_{t \in T} |G_{T^*}(t)|$$

--------- [5.2.6]

Note that if a tuple *t* has been suppressed, the size of *GT*∗ (*t*) is the same as the size of *T*∗. In many situation, suppressions are considered to be most expensive in the sense of information loss. Thus, to maximize data utility, tuple suppression should be avoided whenever possible.

For any given metric *M*, if *M*(*T*) > *M*(*T*′), we say *T* has a higher information loss, or is less precise, than *b*. In other words, data quality of *T* is worse than that of *T*′. Is this true for all metrics? What is a good metric? It is not easy to answer these kinds of questions. *CM* works better than *LM* in classification application. In addition, *LM* is better for association rule mining. It is apparent that to judge how good a particular metric is, we need to associate our judgment with specific applications (e.g., classification, mining association rules).The *CM* metric and the information gain privacy loss ratio are more interesting measure of utility because it considers the possible application for the data. Nevertheless, it is unclear what to do if we want to build classifiers on various attributes. In addition, these two metrics only work well if the data are intended to be used for building classifiers. Is there a utility metric that works well for various applications? Having this in mind, Kifer [17] proposes a utility measure related to Kullback-Leibler divergence. In theory, using this measure, *better* anonymous datasets (for different applications) can be produced. Researchers have measured the utility of the resulting anonymous datasets. Preliminary results show that this metric works well in practical applications. For the statistical-based perturbation techniques which aim to hide the values of a confidential attribute, the information loss is basically the lack of precision in estimating the original distribution function of the given attribute. The information loss incurred during the reconstruction of estimating the density function *fX* (*x*) of the attribute *X*, is measured by computing the following value:

$$I(f_X, \widehat{f}_X) = \frac{1}{2} E\left[\int_{\Omega_X} \left| f_X(x) - \widehat{f}_X(x) \right| dx\right]$$

---------- [5.2.7]

That is, half of the expected value of L1 norm between *fX* (*x*) and b*fX* (*x*), which are the density distributions respectively before and after the application of the privacy preserving technique. When considering the cryptography-based techniques which are typically employed in distributed environments, we can observe that they do not use any kind of perturbation techniques for the purpose of privacy preserving. Instead, they use the cryptographic techniques to assure data privacy at each site by limiting the information shared by all the sites. Therefore, the quality of data stored at each site is not compromised at all.

Completeness and Consistency

While the accuracy is a relatively general parameter in that it can be measured Without strong assumptions on the dataset analyzed, the completeness is not so general. For example, in some PPDM strategies, e.g. blocking, the completeness evaluation is not significant. On the other hand, the consistency requires to determine all the relationships that are relevant for a given dataset. Bertino et al. propose a set of evaluation parameters including the completeness and consistency evaluation. Unlike other techniques, their approach takes into account two more important aspects: relevance of data and structure of database. They provide a formal description that can be used to magnify the aggregate information of interest for a target database and the relevance of data quality properties of each aggregate information and for each attribute involved in the aggregate information. Specifically, the completeness lack (denoted as CML) is measured as follows:

$$CML = \sum_{i=0}^{n} (DMG.N_i.CV \cdot DMG.N_i.CW)$$

---------- [5.2.8]

DMG is an oriented graph where each node *Ni* is an attribute class.

CV is the completeness value and*CW* is the consistency value. The consistency lack (denoted as CSL) is given by the number of constraint violations occurred in all the sanitized transaction multiplied by the weight associated with every constraints.

$$CSL = \sum_{i=0}^{n} (DMG.SC_i.csv \cdot DMG.SC_i.cw) + \sum_{j=0}^{m} (DMG.CC_j.csv \cdot DMG.CC_j.cw)$$

----- [5.2.9]

In equation 12, *csv* indicates the number of violations, *cw* is the weight of the constraint, *SCi* describes a simple constraint class, and *CCj* describes a complex constraint class.

TESTING

Introduction

Testing is essential for database applications to function correctly and with acceptable performance when deployed. Currently, two approaches dominate database application testing. With the first approach, application developers carry out their tests on their own local *development* databases. Obviously this approach cannot fulfill the requirements of all the testing phases, especially those pertinent to performance and scalability, due to the limitation of relatively small size of data and test cases. Furthermore, the data in *local development databases* may not be accurate or close to real data. With the second approach, new applications are tested over *live production* databases. This approach cannot be applied in most situations due to the high risks of disclosure and incorrect updating of confidential information.

Our approach is more feasible than other approaches. First, our approach is to generate data specifically for the purpose of testing and to run the tests in an isolated environment. The databases can possibly be shared by all developers so they can run their applications and see how it work with either realistic amounts or any amounts of data, rather than a handful of records in a local *development* database. Second, the a-priori knowledge required is generally available from detailed entity-relation diagram (ER) or schema definitions in data definition language (DDL) with complex data integrity rules as well as statistical information as an organization implementing a complex database application usually has clear understanding of the nature of the data on which the system will operate. Third, our approach can achieve better *controllability, observability, and privacy*. Using synthetic data, we can put a database system into the desired state before executing test cases (controllability) and observe its state after the execution of the test cases (observability). Two further potential threats, namely direct disclosure of individual data and indirect interpretation from the disclosed data to confidential data, are prevented by using synthetic datasets. A further advantage over other approaches is that there is no need in our approach to access the confidential live data. A formidable challenge that will be addressed in this paper is to minimize indirect confidential information leakage (e.g., the inference of some deterministic or non-deterministic business rules of the original database). It is often necessary for a database software vendor to test their software on a live commercial database before selling or integrating their package to the database owner. The testing of database applications can be classified as: functional testing, performance testing (load and stress, scalability), environment and compatibility testing, and usability testing. In this paper, we focus on performance testing which identifies current bottlenecks in application and verifies whether it meets or exceeds key performance measures. The crucial point here is how we can design the testing environment so that all functions of the software package are tested while no confidential information of the real database is leaked.

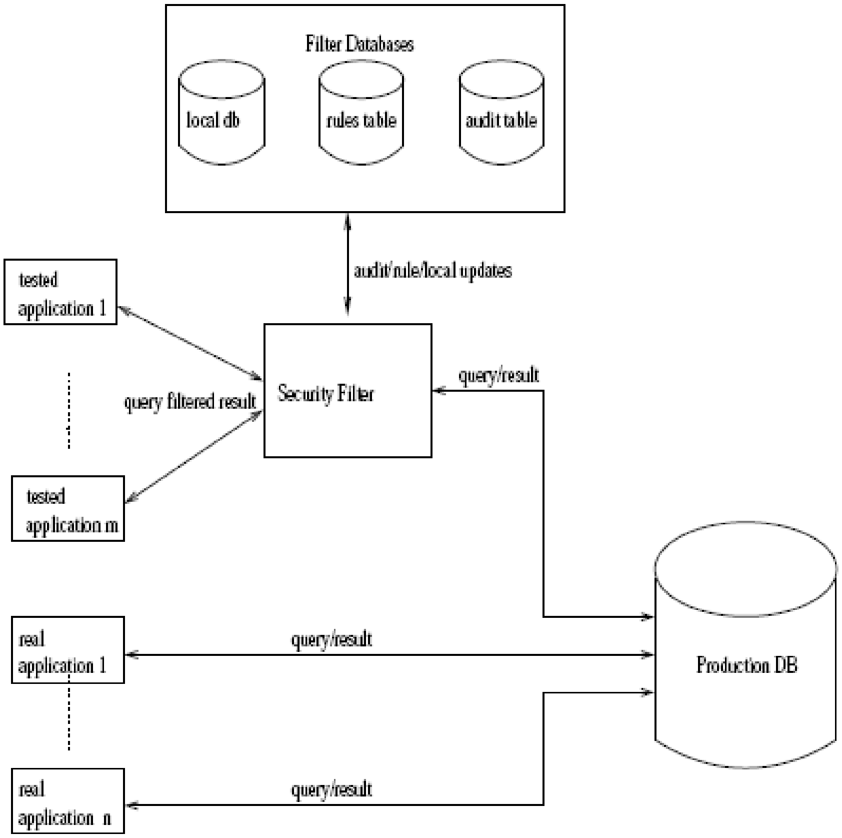


Figure 1: Architecture of live testing approach

Rule based data specification—generating the triplet *(R;N R; S)*

The specification of database testing involves characterizing data values, distributions, and relations. Thus, to achieve the goal of generating valid, close looking data, we expect the users to provide knowledge about the values, distribution, relations, and integrity constraints the data embodies Let *G* be a random process (i.e., a nondeterministic Turing machine) such that, for any consistent triplet *(R;NR; S)* and any random coin tosses *r* (i.e., a random binary sequence), *G* generates a mock database *DB* = *G*(*hR;NR; Si; r*)

CONCLUSION AND FUTURE WORK

CONCLUSION

How to protect sensitive information from the security threats brought by data mining has become a hot topic in recent years. In this paper we review the privacy issues related to data mining by using a user-role based methodology. We differentiate four different user roles that are commonly involved in data mining applications, i.e. data provider, data collector, data miner and decision maker. Each user role has its own privacy concerns, hence the privacy-preserving approaches adopted by one user role are generally different from those adopted by others:

- For data provider, his privacy-preserving objective is to effectively control the amount of sensitive data revealed to others. To achieve this goal, he can utilize security tools to limit other's access to his data, sell his data at auction to get enough compensations for privacy loss, or falsify his data to hide his true identity.
- For data collector, his privacy-preserving objective is to release useful data to data miners without disclosing data providers' identities and sensitive information about them. To achieve this goal, he needs to develop proper privacy models to quantify the possible loss of privacy under different attacks, and apply anonymization techniques to the data.
- For data miner, his privacy-preserving objective is to get correct data mining results while keep sensitive information undisclosed either in the process of data mining or in the mining results. To achieve this goal, he can choose a proper method to modify the data before certain mining algorithms are applied to, or utilize secure computation protocols to ensure the safety of private data and sensitive information contained in the learned model.
- For decision maker, his privacy-preserving objective is to make a correct judgement about the credibility of the data mining results he's got. To achieve this goal, he can utilize provenance techniques to trace back the history of the received information, or build classifier to discriminate true information from false information.

To achieve the privacy-preserving goals of different users roles, various methods from different research fields are required. We have reviewed recent progress in related studies, and discussed problems waiting to be further investigated. We hope that the review presented in this paper can offer researchers different insights into the issue of privacy-preserving data mining, and promote the exploration of new solutions to the security of sensitive information.

FUTURE RESEARCH DIRECTIONS

Although we have already pointed out some problems that need to be further investigated for each user role here in this section, we highlight some of the problems and consider them to be the major directions of future research.

A. Personalized Privacy Preserving

PPDP and PPDM provide methods to explore the utility of data while preserving privacy. However, most current studies only manage to achieve privacy preserving in a statistical sense. Considering that the definition of privacy is essentially personalized, developing methods that can support personalized privacy preserving is an important direction for the study of PPDP and PPDM. Researchers have already investigated the issue of personalized anonymization, but most current studies are still in the theoretical stage. Developing practical personalized anonymization methods is in urgent need. Besides, introducing personalized privacy into other types of PPDP/PPDM algorithms is also required. In addition, since complex socioeconomic and psychological factors are involved, quantifying individual's privacy preference is still an open question which expects more exploration.

B. Data Customization

By inverse data mining, we can "customize" the data to get the desired mining result. Alexandra introduced a concept called *reverse data management* (RDM) which is similar to our specification for inverse data mining. RDM consists of problems where one needs to compute a database input, or modify an existing database input, in order to achieve a desired effect in the output. RDM covers many database problems such as inversion mappings, provenance, data generation, view update, constraint-based repair, etc. We may consider RDM to be a family of data customization methods by which we can get the desired data from which sensitive information cannot be discovered. In a word, data customization can be seen as the inverse process of ordinary data processing. Whenever we have explicit requirements for the outcome of data processing, we may resort to data customization. Exploring ways to solve the inverse problem is an important task for future study.

C. Provenance for Data Mining

The complete process of data mining consists of multiple phases such as data collection, data preprocessing, data mining, analyzing the extracted information to get knowledge, and applying the knowledge. This process can be seen as an evolvement of data. If the provenance information corresponding to every phase in the process, such as the ownership of data and how the data is processed, can be clearly recorded, it will be much easier to find the origins of security incidents such as sensitive data breach and the distortion of sensitive information. We may say that provenance provides us a way to monitor the process of data mining and the use of mining result. Therefore, techniques and mechanisms that can support provenance in data mining context should receive more attention in future study. Glavic have discussed how traditional notions of provenance translated to data mining. They identified the need for new types of provenance that can be used to better interpret data mining results. In the context of privacy protection, we are more concerned with how to use provenance to better understand why and how "abnormal" mining result, e.g. result containing sensitive information or false result, appears. Different from provenance approaches that we have reviewed, approaches for data mining provenance are closely related to the mining algorithm. Therefore, it is necessary to develop new provenance models to specify what kind of provenance information is required and how to present, store, acquire and utilize the provenance information.

REFERENCES

[1]. J. Han, M. Kamber and J. Pei, Data Mining: Concepts and Techniques *2006, Morgan Kaufmann*

[2]. L. Brankovic and V. Estivill-Castro "Privacy issues in knowledge discovery and data mining" *Proc. Austral. Inst. Comput. Ethics Conf., pp. 89-99,*

[3]. R. Agrawal and R. Srikant "Privacy-preserving data mining" *ACM SIGMOD Rec., vol. 29, pp. 439-450, 2000* Quick Abstract | | Full Text: Access at ACM

[4]. Y. Lindell and B. Pinkas "Privacy preserving data mining" *Springer-Verlag, pp. 36-54,*

[**5**]. C. C. Aggarwal and S. Y. Philip A General Survey of Privacy-Preserving Data Mining Models and Algorithms *2008, Springer-Verlag*

[6]. M. B. Malik, M. A. Ghazi and R. Ali "Privacy preserving data mining techniques: Current scenario and future prospects" *Proc. 3rd Int. Conf. Comput. Commun. Technol. (ICCCT), pp. 26-32,* Quick Abstract | | Full Text: PDF

[7]. S. Matwin "Privacy-preserving data mining techniques: Survey and challenges" *Springer-Verlag, pp. 209-221,*

[8]. E. Rasmusen Games and Information: An Introduction to Game Theory *vol. 2, 1994, Blackwell*

[9]. V. Ciriani, S. De Capitani di Vimercati, S. Foresti and P. Samarati "Microdata protection" *Springer-Verlag, pp. 291-321,*

[10]. O. Tene and J. Polenetsky "To track or ???do not track???: Advancing transparency and individual control in online behavioral advertising" *Minnesota J. Law, Sci. Technol., no. 1, pp. 281-357, 2012*

[11]. R. T. Fielding and D. Singer Tracking Preference Expression (DNT). W3C Working Draft *2014, [online] Available:*

[12]. R. Gibbons A Primer in Game Theory *1992, Harvester Wheatsheaf*

[13]. D. C. Parkes "Iterative combinatorial auctions: Achieving economic and computational efficiency" *2001*

[14]. S. Carter "Techniques to pollute electronic profiling" *2007, [online] Available:https://www.google.com/patents/US20070094738*

[15]. 2013 Data Breach Investigations Report *2013, [online] Available: http://www.verizonenterprise.com/resources/reports/rp_data-breach-investigations-report-2013_en_xg.pdf*

[16]. A. Narayanan and V. Shmatikov "Robust de-anonymization of large sparse datasets" *Proc. IEEE Symp. Secur. Privacy (SP), pp. 111-125,* Quick Abstract | | Full Text: PDF

[17]. B. C. M. Fung, K. Wang, R. Chen and P. S. Yu "Privacy-preserving data publishing: A survey of recent developments" *ACM Comput. Surv., vol. 42, no. 4, 2010* Quick Abstract | | Full Text: Access at ACM

[18]. R. C.-W. Wong and A. W.-C. Fu "Privacy-preserving data publishing: An overview" *Synthesis Lectures Data Manage., vol. 2, no. 1, pp. 1-138, 2010*

[19]. L. Sweeney "-anonymity: A model for protecting privacy" *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst., vol. 10, no. 5, pp. 557-570, 2002*

[20]. R. J. Bayardo and R. Agrawal "Data privacy through optimal k-anonymization" *Proc. 21st Int. Conf. Data Eng. (ICDE), pp. 217-228,* Quick Abstract | | Full Text: PDF

[21]. K. LeFevre, D. J. DeWitt and R. Ramakrishnan "Mondrian multidimensional k-anonymity" *Proc. 22nd Int. Conf. Data Eng. (ICDE), p. 25,* Quick Abstract | | Full Text: PDF

[22]. J. Xu, W. Wang, J. Pei, X. Wang, B. Shi and A. W.-C. Fu "Utility-based anonymization for privacy preservation with less information loss" *ACM SIGKDD Explorations Newslett., vol. 8, no. 2, pp. 21-30, 2006* Quick Abstract | | Full Text: Access at ACM

[23]. A. Gionis and T. Tassa "k-anonymization with minimal loss of information" *IEEE Trans. Knowl. Data Eng., vol. 21, no. 2, pp. 206-219, 2009* Quick Abstract | | Full Text: PDF

[24]. B. Zhou, J. Pei and W. Luk "A brief survey on anonymization techniques for privacy preserving publishing of social network data" *ACM SIGKDD Explorations Newslett., vol. 10, no. 2, pp. 12-22, 2008* Quick Abstract | | Full Text: Access at ACM

[25]. X. Wu, X. Ying, K. Liu and L. Chen "A survey of privacy-preservation of graphs and social networks" *Springer-Verlag, pp. 421-453,*

[26]. S. Sharma, P. Gupta and V. Bhatnagar "Anonymisation in social network: A literature survey and classification" *Int. J. Soc. Netw. Mining, vol. 1, no. 1, pp. 51-66, 2012*

[27]. W. Peng, F. Li, X. Zou and J. Wu "A two-stage deanonymization attack against anonymized social networks" *IEEE Trans. Comput., vol. 63, no. 2, pp. 290-303, 2014* Quick Abstract | | Full Text: PDF

[28]. T. Zhu, S. Wang, X. Li, Z. Zhou and R. Zhang "Structural attack to anonymous graph of social networks" *Math. Problems Eng., vol. 2013, 2013*

[29]. C. Sun, P. S. Yu, X. Kong and Y. Fu *2013, [online] Available:http://arxiv.org/abs/1401.3201* Quick Abstract | | Full Text: PDF

[30]. C.-H. Tai, P. S. Yu, D.-N. Yang and M.-S. Chen "Privacy-preserving social network publication against friendship attacks" *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, pp. 1262-1270,*

[31]. C.-H. Tai, P. S. Yu, D.-N. Yang and M.-S. Chen "Structural diversity for resisting community identification in published social networks" *IEEE Trans. Knowl. Data Eng., vol. 26, no. 1, pp. 235-252, 2013* Quick Abstract | | Full Text: PDF

[32]. M. I. Hafez Ninggal and J. Abawajy "Attack vector analysis and privacy-preserving social network data publishing" *Proc. IEEE 10th Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom), pp. 847-852,*

[33]. Y. Wang, L. Xie, B. Zheng and K. C. K. Lee "High utility k-anonymization for social network publishing" *Knowl. Inf. Syst., vol. 36, no. 1, pp. 1-29, 2013*

[34]. N. Medforth and K. Wang "Privacy risk in graph stream publishing for social network data" *Proc. IEEE 11th Int. Conf. Data Mining (ICDM), pp. 437-446,* Quick Abstract | | Full Text: PDF

[35]. C.-H. Tai, P.-J. Tseng, P. S. Yu and M.-S. Chen "Identity protection in sequential releases of dynamic networks" *IEEE Trans. Knowl. Data Eng., vol. 26, no. 3, pp. 635-651, 2014* Quick Abstract | | Full Text: PDF

[36]. G. Ghinita Privacy for Location-Based Services (Synthesis Lectures on Information Security, Privacy, and Trust) *2013, Morgan & Claypool*

[37]. M. Wernke, P. Skvortsov, F. D??rr and K. Rothermel "A classification of location privacy attacks and approaches" *Pers. Ubious Comput., vol. 18, no. 1, pp. 163-175, 2014*

[38]. M. Terrovitis and N. Mamoulis "Privacy preservation in the publication of trajectories" *Proc. 9th Int. Conf. Mobile Data Manage. (MDM), pp. 65-72,* Quick Abstract | | Full Text: PDF

[39]. M. E. Nergiz, M. Atzori and Y. Saygin "Towards trajectory anonymization: A generalization-based approach" *Proc. SIGSPATIAL ACM GIS Int. Workshop Secur. Privacy GIS LBS, pp. 52-61*, Quick Abstract | | Full Text: Access at ACM

[40]. O. Abul, F. Bonchi and M. Nanni "Never walk alone: Uncertainty for anonymity in moving objects databases" *Proc. IEEE 24th Int. Conf. Data Eng. (ICDE), pp. 376-385*, Quick Abstract | | Full Text: PDF

[41]. R. Yarovoy, F. Bonchi, L. V. S. Lakshmanan and W. H. Wang "Anonmizing moving objects: How to hide a MOB in a crowd?" *Proc. 12th Int. Conf. Extending Database Technol., Adv. Database Technol., pp. 72-83*, Quick Abstract | | Full Text: Access at ACM

[42]. R. Chen, B. C. M. Fung, N. Mohammed, B. C. Desai and K. Wang "Privacy-preserving trajectory data publishing by local suppression" *Inf. Sci., vol. 231, pp. 83-97, 2013*

[43]. M. Ghasemzadeh, B. C. M. Fung, R. Chen and A. Awasthi "Anonymizing trajectory data for passenger flow analysis" *Transp. Res. C, Emerg. Technol., vol. 39, pp. 63-79, 2014*

[44]. A. E. Cicek, M. E. Nergiz and Y. Saygin "Ensuring location diversity in privacy-preserving spatio-temporal data publishing" *VLDB J., vol. 23, no. 4, pp. 1-17, 2013*

[45]. G. Poulis, S. Skiadopoulos, G. Loukides and A. Gkoulalas-Divanis "Distance-based k^ m-anonymization of trajectory data" *Proc. IEEE 14th Int. Conf. Mobile Data Manage. (MDM), vol. 2, pp. 57-62*, Quick Abstract | | Full Text: PDF

[46]. F. Bonchi, L. V. S. Lakshmanan and H. W. Wang "Trajectory anonymity in publishing personal mobility data" *ACM SIGKDD Explorations Newslett., vol. 13, no. 1, pp. 30-42, 2011* Quick Abstract | | Full Text: Access at ACM

[47]. X. Xiao and Y. Tao "Personalized privacy preservation"*Proc. ACM SIGMOD Int. Conf. Manage. Data, pp. 229-240,*

[48]. K. Qing-Jiang, W. Xiao-Hao and Z. Jun “k- anonymity model for privacy protection of personal information in the social networks" *Proc. 6th IEEE Joint Int. Inf. Technol. Artif. Intell. Conf. (ITAIC), vol. 2, pp. 420-423,*

[49]. B. Wang and J. Yang “k-anonymity algorithm based on entropy classification" *J. Comput. Inf. Syst., vol. 8, no. 1, pp. 259-266, 2012*

[50]. Y. Xua, X. Qin, Z. Yang, Y. Yang and K. Li "A personalized k-anonymity privacy preserving method" *J. Inf. Comput. Sci., vol. 10, no. 1, pp. 139-155, 2013*

[51]. S. Yang, L. Lijie, Z. Jianpei and Y. Jing "Method for individualized privacy preservation" *Int. J. Secur. Appl., vol. 7, no. 6, p. 109, 2013*

[52]. A. Halevy, A. Rajaraman and J. Ordille "Data integration: The teenage years" *Proc. 32nd Int. Conf. Very Large Data Bases (VLDB), pp. 9-16,*

[53]. V. S. Verykios, E. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin and Y. Theodoridis "State-of-the-art in privacy preserving data mining" *ACM SIGMOD Rec., vol. 33, no. 1, pp. 50-57, 2004* Quick Abstract | | Full Text: Access at ACM

[54]. C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin and M. Y. Zhu "Tools for privacy preserving distributed data mining" *ACM SIGKDD Explorations Newslett., vol. 4, no. 2, pp. 28-34, 2002* Quick Abstract | | Full Text: Access at ACM

[55]. R. Agrawal, T. Imieli??ski and A. Swami "Mining association rules between sets of items in large databases" *Proc. ACM SIGMOD Rec., vol. 22, no. 2, pp. 207-216,* Quick Abstract | | Full Text: Access at ACM

[56]. V. S. Verykios "Association rule hiding methods" *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery, vol. 3, no. 1, pp. 28-36, 2013*

[57]. K. Sathiyapriya and G. S. Sadasivam "A survey on privacy preserving association rule mining" *Int. J. Data Mining Knowl. Manage. Process, vol. 3, no. 2, p. 119, 2013*

[58]. D. Jain, P. Khatri, R. Soni and B. K. Chaurasia "Hiding sensitive association rules without altering the support of sensitive item(s)" *Proc. 2nd Int. Conf. Adv. Comput. Sci. Inf. Technol. Netw. Commun., pp. 500-509,*

[59]. J.-M. Zhu, N. Zhang and Z.-Y. Li "A new privacy preserving association rule mining algorithm based on hybrid partial hiding strategy" *Cybern. Inf. Technol., vol. 13, pp. 41-50, 2013*

[60]. H. Q. Le, S. Arch-Int, H. X. Nguyen and N. Arch-Int "Association rule hiding in risk management for retail supply chain collaboration" *Comput. Ind., vol. 64, no. 7, pp. 776-784, 2013*

[61]. M. N. Dehkordi "A novel association rule hiding approach in OLAP data cubes" *Indian J. Sci. Technol., vol. 6, no. 2, pp. 4063-4075, 2013*

[62]. J. Bonam, A. R. Reddy and G. Kalyani "Privacy preserving in association rule mining by data distortion using PSO" *Proc. ICT Critical Infrastruct., Proc. 48th Annu. Conv. Comput. Soc. India, vol. 2, pp. 551-558,*

[63]. C. N. Modi, U. P. Rao and D. R. Patel "Maintaining privacy and data quality in privacy preserving association rule mining" *Proc. Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT), pp. 1-6,* Quick Abstract | | Full Text: PDF

[64]. N. R. Radadiya, N. B. Prajapati and K. H. Shah "Privacy preserving in association rule mining"*Int. J. Adv. Innovative Res., vol. 2, no. 4, pp. 203-213, 2013*

[65]. K. Pathak, N. S. Chaudhari and A. Tiwari "Privacy preserving association rule mining by introducing concept of impact factor" *Proc. 7th IEEE Conf. Ind. Electron. Appl. (ICIEA), pp. 1458-1461,*Quick Abstract | | Full Text: PDF

[66]. T. Mielik??inen "On inverse frequent set mining" *Proc. 2nd Workshop Privacy Preserving Data Mining, pp. 18-23,*

[67]. X. Chen and M. Orlowska "A further study on inverse frequent set mining" *Proc. 1st Int. Conf. Adv. Data Mining Appl., pp. 753-760,*

[68]. Y. Guo "Reconstruction-based association rule hiding" *Proc. SIGMOD Ph. D. Workshop Innovative Database Res., pp. 51-56,*

[69]. Y. Wang and X. Wu "Approximate inverse frequent itemset mining: Privacy, complexity, and approximation"*Proc. 5th IEEE Int. Conf. Data Mining, p. 8,*

[70]. Y. Guo, Y. Tong, S. Tang and D. Yang "A FP-tree-based method for inverse frequent set mining" *Proc. 23rd Brit. Nat. Conf. Flexible Efficient Inf. Handling, pp. 152-163,*

[71]. J. Dowd, S. Xu and W. Zhang "Privacy-preserving decision tree mining based on random substitutions" *Proc. Int. Conf. Emerg. Trends Inf. Commun. Security, pp. 145-159,*

[72]. J. Brickell and V. Shmatikov "Privacy-preserving classifier learning" *Proc. 13th Int. Conf. Financial Cryptogr. Data Security, pp. 128-147,*

[73]. P. K. Fong and J. H. Weber-Jahnke "Privacy preserving decision tree learning using unrealized data sets" *IEEE Trans. Knowl. Data Eng., vol. 24, no. 2, pp. 353-364, 2012* Quick Abstract | | Full Text: PDF

[74]. M. A. Sheela and K. Vijayalakshmi "A novel privacy preserving decision tree induction" *Proc. IEEE Conf. Inf. Commun. Technol. (ICT), pp. 1075-1079,* Quick Abstract | | Full Text: PDF

[75]. O. Goldreich Secure Multi-Party Computation *2002, [online] Available: http://www.wisdom.weizmann.ac.il/~oded/PS/prot.ps*

[76]. J. Vaidya, M. Kantarc??o??lu and C. Clifton "Privacy-preserving Na??ve Bayes classification" *Int. J. Very Large Data Bases, vol. 17, no. 4, pp. 879-898, 2008*

[77]. M. E. Skarkala, M. Maragoudakis, S. Gritzalis and L. Mitrou "Privacy preserving tree augmented Na??ve Bayesian multi-party implementation on horizontally partitioned databases" *Proc. 8th Int. Conf. Trust, Privacy, Security Digit. Bus., pp. 62-73,*

[78]. F. Zheng and G. I. Webb "Tree augmented Na??ve Bayes" *Proc. Encyclopedia Mach. Learn., pp. 990-991,*

[79]. J. Vaidya, B. Shafiq, A. Basu and Y. Hong "Differentially private Na??ve Bayes classification" *Proc. IEEE/WIC/ACM Int. Joint Conf. Web Intell. (WI) Intell. Agent Technol. (IAT), vol. 1, pp. 571-576,*

[80]. C. Dwork "Differential privacy" *Proc. 33rd Int. Conf. Autom., Lang., Program., pp. 1-12,*

[81]. J. Vaidya, H. Yu and X. Jiang "Privacy-preserving SVM classification" *Knowl. Inf. Syst., vol. 14, no. 2, pp. 161-178, 2008*

[82]. H. Xia, Y. Fu, J. Zhou and Y. Fang "Privacy-preserving SVM classifier with hyperbolic tangent kernel" *J. Comput. Inf. Syst., vol. 6, no. 5, pp. 1415-1420, 2010*

[83]. K.-P. Lin and M.-S. Chen "On the design and analysis of the privacy-preserving SVM classifier"*IEEE Trans. Knowl. Data Eng., vol. 23, no. 11, pp. 1704-1717, 2011*Quick Abstract | | Full Text: PDF

[84]. R. R. Rajalaxmi and A. M. Natarajan "An effective data transformation approach for privacy preserving clustering" *J. Comput. Sci., vol. 4, no. 4, pp. 320-326, 2008*

[85]. M. N. Lakshmi and K. S. Rani "SVD based data transformation methods for privacy preserving clustering" *Int. J. Comput. Appl., vol. 78, no. 3, pp. 39-43, 2013*

[86]. J. Vaidya and C. Clifton "-means clustering over vertically partitioned data" *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, pp. 206-215,*

[87]. S. Jha, L. Kruger and P. McDaniel "Privacy preserving clustering" *Proc. 10th Eur. Symp. Res. Comput. Security (ESORICS), pp. 397-417,*

[88]. R. Akhter, R. J. Chowdhury, K. Emura, T. Islam, M. S. Rahman and N. Rubaiyat "-means clustering in malicious model" *Proc. IEEE 37th Annu. Comput. Softw. Appl. Conf. Workshops (COMPSACW), pp. 121-126,* Quick Abstract | | Full Text: PDF

[89]. X. Yi and Y. Zhang "-means clustering over vertically partitioned data" *Inf. Syst., vol. 38, no. 1, pp. 97-107, 2013*

[90]. I. De and A. Tripathy "A secure two party hierarchical clustering approach for vertically partitioned data set with accuracy measure" *Proc. 2nd Int. Symp. Recent Adv. Intell. Informat., pp. 153-162,*

[91]. Y. L. Simmhan, B. Plale and D. Gannon "A survey of data provenance in e-science" *ACM Sigmod Rec., vol. 34, no. 3, pp. 31-36, 2005* Quick Abstract | | Full Text: Access at ACM

[92]. B. Glavic and K. R. Dittrich "Data provenance: A categorization of existing approaches" *Proc. BTW, vol. 7, no. 12, pp. 227-241,*

[93]. S. B. Davidson and J. Freire "Provenance and scientific workflows: Challenges and opportunities" *Proc. ACM SIGMOD Int. Conf. Manage. Data, pp. 1345-1350,* Quick Abstract | | Full Text: Access at ACM

[94]. O. Hartig "Provenance information in the web of data" *Proc. LDOW, [online] Available: http://ceur-ws.org/Vol-538/ldow2009_paper18.pdf*

[95]. L. Moreau "The foundations for provenance on the web" *Found. Trends Web Sci., vol. 2, no. 3, pp. 99-241, 2010*

[96]. G. Barbier, Z. Feng, P. Gundecha and H. Liu "Provenance data in social media" *Synth. Lectures Data Mining Knowl. Discovery, vol. 4, no. 1, pp. 1-84, 2013*

[97]. M. Tudjman and N. Mikelic "Information science: Science about information, misinformation and disinformation" *Proc. Inf. Sci.+Inf. Technol. Edu., pp. 1513-1527,*

[98]. M. J. Metzger "Making sense of credibility on the Web: Models for evaluating online information and recommendations for future research" *J. Amer. Soc. Inf. Sci. Technol., vol. 58, no. 13, pp. 2078-2091, 2007*

[99]. C. Castillo, M. Mendoza and B. Poblete "Information credibility on Twitter" *Proc. 20th Int. Conf. World Wide Web, pp. 675-684,*

[100]. V. Qazvinian, E. Rosengren, D. R. Radev and Q. Mei "Rumor has it: Identifying misinformation in microblogs" *Proc. Conf. Empirical Methods Natural Lang. Process, pp. 1589-1599,*

[101]. F. Yang, Y. Liu, X. Yu and M. Yang "Automatic detection of rumor on Sina Weibo" *Proc. ACM SIGKDD Workshop Mining Data Semantics,*

[102]. S. Sun, H. Liu, J. He and X. Du "Detecting event rumors on Sina Weibo automatically" *Proc. Web Technol. Appl., pp. 120-131,*

[103]. R. K. Adl, M. Askari, K. Barker and R. Safavi-Naini "Privacy consensus in anonymization systems via game theory" *Proc. 26th Annu. Data Appl. Security Privacy, pp. 74-89,*

[104]. R. Karimi Adl, K. Barker and J. Denzinger "A negotiation game: Establishing stable privacy policies for aggregate reasoning" *2012, [online] Available:http://dspace.ucalgary.ca/jspui/bitstream/1880/49282/1/2012-1023-06.pdf*

[105]. H. Kargupta, K. Das and K. Liu "Multi-party, privacy-preserving distributed data mining using a game theoretic framework" *Proc. 11th Eur. Conf. Principles Pract. Knowl. Discovery Databases (PKDD), pp. 523-531,*

[106]. A. Miyaji and M. S. Rahman "Privacy-preserving data mining: A game-theoretic approach" *Proc. 25th Data Appl. Security Privacy, pp. 186-200,*

[107]. X. Ge, L. Yan, J. Zhu and W. Shi "Privacy-preserving distributed association rule mining based on the secret sharing technique" *Proc. 2nd Int. Conf. Softw. Eng. Data Mining (SEDM), pp. 345-350,*

[108]. N. R. Nanavati and D. C. Jinwala "A novel privacy preserving game theoretic repeated rational secret sharing scheme for distributed data mining" *vol. 91, 2013, [online] Available: http://www.researchgate.net/ publication/256765823_A_NOVEL_PRIVACY_PRESERVING_ GAME_THEORETIC_REPEATED_RATIONAL_ SECRET_SHARING_SCHEME_FOR_DISTRIBUTED_DATA_MINING*

[109]. M. Halkidi and I. Koutsopoulos "A game theoretic framework for data privacy preservation in recommender systems" *Proc. Mach. Learn. Knowl. Discovery Databases, pp. 629-644,*

[110]. S. Ioannidis and P. Loiseau "Linear regression as a non-cooperative game" *Proc. Web Internet Econ., pp. 277-290,*

[111]. S. L. Chakravarthy, V. V. Kumari and C. Sarojini "-anonymity" *Proc. Technol., vol. 6, pp. 889-896, 2012, [online] Available:http://www.sciencedirect.com/science/article/pii/S2212017312006536*

[112]. R. Nix and M. Kantarciouglu "Incentive compatible privacy-preserving distributed classification" *IEEE Trans. Dependable Secure Comput., vol. 9, no. 4, pp. 451-462, 2012* Quick Abstract | | Full Text: PDF

[113]. M. Kantarcioglu and W. Jiang "Incentive compatible privacy-preserving data analysis" *IEEE Trans. Knowl. Data Eng., vol. 25, no. 6, pp. 1323-1335, 2013* Quick Abstract | | Full Text: PDF

[114]. A. Panoui, S. Lambotharan and R. C.-W. Phan "Vickrey???Clarke???Groves for privacy-preserving collaborative classification" *Proc. Fed. Conf. Comput. Sci. Inf. Syst. (FedCSIS), pp. 123-128,*

[115]. A. Ghosh and A. Roth "Selling privacy at auction" *Proc. 12th ACM Conf. Electron. Commerce, pp. 199-208,*

[116]. L. K. Fleischer and Y.-H. Lyu "Approximately optimal auctions for selling privacy when costs are correlated with data" *Proc. 13th ACM Conf. Electron. Commerce, pp. 568-585,*

[117]. K. Ligett and A. Roth "Take it or leave it: Running a survey when privacy comes at a cost" *Proc. 8th Internet Netw. Econ., pp. 378-391,*

[118]. K. Nissim, S. Vadhan and D. Xiao "Redrawing the boundaries on purchasing data from privacy-sensitive individuals" *Proc. 5th Conf. Innov. Theoretical Comput. Sci., pp. 411-422,*

[119]. C. Riederer, V. Erramilli, A. Chaintreau, B. Krishnamurthy and P. Rodriguez "For sale: Your data: By: You" *Proc. 10th ACM Workshop Hot Topics Netw.,*

[120]. A. Meliou, W. Gatterbauer and D. Suciu "Reverse data management" *Proc. VLDB Endowment, vol. 4, no. 12, [online] Available: http://people.cs.umass.edu/~ameli/projects/reverse-data-management/papers/VLDB2011_vision.pdf*

[121]. B. Glavic, J. Siddique, P. Andritsos and R. J. Miller "Provenance for data mining" *Proc. 5th USENIX Workshop Theory Pract. Provenance, p. 5,*