# Combined Clustering Approach for Privacy Preserving Data Mining using Fuzzy Logic

J. Paranthaman
Department of Computer Science and Engineering,
University College of Engineering Panruti,
Cuddalore, Tamil Nadu, India.

***Abstract*— Data mining technology is more significant in identifying patterns and trends from large collections of data. It provides Business intelligent supports, solutions and decisions. Data mining provides large benefits to the individual, commercial and government security sectors, but the aggregation and storage of huge amounts of data leads to an erosion of privacy. we present Combined Clustering approach for a number of non trivial tasks related to privacy preserving advanced data mining. The advantages of all clustering techniques are combined with the K-Means Clustering and Expectation Maximization clustering (EM-clustering) for privacy preserving advanced data mining.**

***Keywords—Combined Clustering Approach; Privacy Preserving Data Mining; Fuzzy Logic.***

## I. INTRODUCTION

Privacy preserving data mining has become an important problem because of the large amount of personal data which is tracked by many business applications. In many cases, users are unwilling to provide personal details unless the privacy of sensitive information is guaranteed. Privacy-preserving data mining has been an active area of research since it was introduced by Agrawal and Srikant [7] and Lindell and Pinkas [3].

### *A. Data mining*

Data mining is a recently emerging field, connecting the three worlds of Databases, Artificial Intelligence and Statistics. The information age has enabled many organizations to gather large volumes of data. However, the usefulness of this data is negligible if "meaningful information" or "knowledge" cannot be extracted from it. Data mining, otherwise known as knowledge discovery, attempts to answer this need. In contrast to standard statistical methods, data mining techniques search for interesting information without demanding a priori hypotheses. As a field, it has introduced new concepts and algorithms such as association rule learning. It has also applied known machine-learning algorithms such as inductive-rule learning (e.g., by decision trees) to the setting where very large databases are involved. Data mining techniques are used in business and research and are becoming more and more popular with time [1-3] [18].

### *B. Confidentiality issues in data mining*

A key problem that arises in any e-mass collection of data is that of confidentiality. The need for privacy is sometimes due to law (e.g., for medical databases) or can be motivated by business interests. However, there are situations where the sharing of data can lead to mutual gain. A key utility of large databases today is research, whether it be scientific, or economic and market oriented [4] [18].

### *C. Very large databases and efficient secure computation*

We have described a model which is exactly that of multi-party computation. Therefore, there exists a secure protocol for any probabilistic polynomial time functionality [10]. However these generic solutions are very inefficient, especially when large inputs and complex algorithms are involved. Thus, in the case of private data mining, more efficient solutions are required. It is clear that any reasonable solution must have the individual parties do the majority of the computation independently. Our solution is based on this guiding principle and in fact, the number of bits communicated is dependent on the number of transactions by a logarithmic factor only. We remark that a necessary condition for obtaining such a private protocol is the existence of a (non-private) distributed protocol with low communication complexity [5-7] [18].

Privacy also takes many different forms. Some of the more relevant ones to event correlation include:

1. Source anonymity refers, in particular, to the producer of an event. A source that is anonymous cannot be traced by recipients of the event. there is no explicit identifier linking the event to a known producer and the data in the event cannot reliably be linked to the producer.

2. Data privacy is related to, but not equivalent to, source anonymity: it specifically refers to the semantics of the data in the event and whether they contain information that may be deemed sensitive by the producer of the event.

3. Physical privacy refers to the access of sensitive information or resources via direct access to the repositories or interference with servers of data. This includes intruders, malicious insiders, and resource starvation (e.g., denial of service) mechanisms.

4. Time privacy corresponds to the fact that this thesis considers an event as being time stamped. The distribution of event arrival times could yield some aggregate

information; more interestingly, the correlation of curious or insidious activities with event arrival times could potentially violate the source anonymity stated above [5].

This thesis focuses on the first two forms of privacy. It is possible to maintain data privacy without maintaining source anonymity (e.g., an event came from source X but it is free of what X deems sensitive), as well as vice-versa (e.g., it is unknown exactly who the event came from, but it contains classified information privy to only a small number of organizations). Of course, both can exist in tandem. With both, I argue that recipients cannot trace the source or information for relevant applications [8-10].

As for the latter two forms, physical privacy poses a unique set of challenges on its own most systems secure from remote access have physical backdoors and is considered outside the scope of this thesis. Meanwhile, the definition of events and event correlation assume an ordering amongst events. Some of the data privacy approaches in the proposal do indirectly provide time privacy, but full time privacy poses its own unique correlation challenges; a complete discussion is outside the scope of this work. Finally, a privacy policy is both a promise by an organization to originators of data contained within the organization, as well as a compliance statement to consumers of data produced by the organization. It may contain one or both of the first two privacy requirements, as well as other additional requirements.

## II. DATA MINING TECHNIQUES

Data mining techniques include the following:

- Decision Trees/Rules
- Clustering
- Statistics
- Neural networks
- Logistic regression
- Visualization
- Association rules
- Nearest neighbor
- Text mining
- Web mining
- Bayesian nets / Naive Bayes
- Sequence analysis
- SVM (Support Vector Machine)
- Hybrid methods
- Genetic algorithms

In the following, we will discuss some of these techniques briefly. The above Data mining techniques are divided into the following three categories data mining techniques. They are classification, prediction and estimation which are observed from the references [11-14].

### *A. Rule induction*

A data mine system has to infer a model from the database; that is, it may define classes such that the database contains one or more attributes that denote the class of a tuple. The class can then be defined by the condition of the attributes. When the classes are defined, the system should be able to infer the rules that govern classification. In other words, the system should find the description of each class. Production rules have been widely used to represent knowledge in expert systems and they have the advantage of being easily interpreted by human experts because of their modularity, i.e. a single rule can be understood in isolation and does not need reference to other rules [18].

### *B. Association rules*

Association rule mining finds interesting associations and/or correlation relationships among large sets of data items. Association rules show attributes value conditions that occur frequently together in a given dataset. A typical and widely-used example of association rule mining is Market Basket Analysis. For example, the data are collected using bar-code scanners in supermarkets. Such market basket databases consist of a large number of transaction records. Each record lists all items bought by a customer on a single transaction. Managers would be interested to know if certain groups of items are consistently purchased together. They could use this data for adjusting store layouts (placing items optimally with respect to each other), for cross-selling, for promotions, for catalog design and to identify customer segments based on buying patterns. Association rules provide information of this type in the form of "if-then" statements. These rules are computed from the data and, unlike the if-then rules of logic, association rules are probabilistic in nature [18].

### *C. Clustering*

In an unsupervised learning environment, the system has to discover its own classes and one way in which it does this is to cluster the data in the database. The first step is to find subsets of related objects and then find descriptions which identify each of these subsets. Clustering and segmentation essentially partition the database so that each partition or group is similar according to some criteria or metric. Clustering according to similarity is a concept which appears in many disciplines. If a measure of similarity is available, there are a number of techniques for forming clusters. Membership of groups can be based on the degree of similarity between members and from this the rules of membership can be defined. Another approach is to construct a set of functions that measure some property of partitions; that is, groups or subsets as functions of some parameter of the partition. This latter approach achieves what is known as optimal partitioning. Many data mining applications make use of clustering according to similarity for example to segment a client/customer base. Clustering according to optimization of set functions is used in data analysis, e.g. when setting insurance tariffs, the customers can be divided according to a number of parameters and the optimal tariff segmentation achieved [13-17].

### *D. Decision trees*

Decision trees are an easy knowledge representation technique and they divide examples to a limited number of classes. The nodes are labeled with dimension names, the edges are labeled with potential values for this dimension and the leaves labeled with distinct classes. Objects are classified by following a route down the tree, by taking the edges, proportionate to the values of the attributes in a target [18].

*E. Neural networks*

Neural networks are an access to computing that involves developing numerical structures with the ability to learn. The methods are the result of academic investigations to model nervous system learning. Neural networks have the extraordinary power to infer significance from complicated or inexact information and can be used to distill patterns and discover trends that are overly complicated to be noticed by either humans or new computer techniques. A skilled neural network can be thought of as an "expert" in the class of data it has been given to analyze. This expert can so be used to offer projections, given original situations of stake and respond to "what if" questions. Neural networks have broad applicability to real world business problems and have already been successfully applied in many industries. Since neural networks are best at identifying patterns or trends in data, they are well suited for prediction or forecasting needs, among them

- Sales forecasting
- Industrial process control
- Customer research
- Data validation
- Risk management
- Target marketing etc.

Neural networks take a lot of processing elements similar to neurons in the mind. These processing elements are interconnected in a web that can so describe patterns in information once it is exposed to the information. This distinguishes neural networks from conventional computation programs that merely follow instructions in a fixed sequential decree.

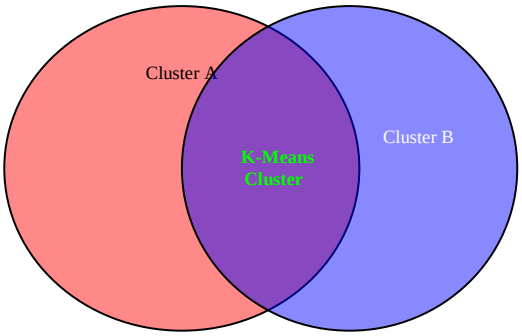


Fig. 1. K-means clustering

Cluster Analysis is the problem of decomposing or partitioning a (usually Multivariate) data set into groups so that the points in one group are similar to each other and are as different as possible from the points in other groups. There are many situations where clustering can lead to the discovery of important knowledge but privacy/security reasons restrict the sharing of data.

Imagine the following scenario. A law enforcement agency wants to cluster individuals based on their financial transactions, and study the differences between the clusters and known money laundering operations. Knowing the differences and similarities between normal individuals and known money launderers would enable better direction of investigations. Currently, an individual's financial transactions may be divided between banks, credit card companies, tax collection agencies, etc. Each of these (presumably) has effective controls governing release of the information. These controls are not perfect, but violating them (either technologically or through insider misuse) reveals only a subset of an individual's financial records. The law enforcement agency could promise to provide effective controls, but now overcoming those gives access to an individual's entire financial history. This raises justifiable concerns among privacy advocates. What is required is a privacy preserving way of doing clustering [13-17].

## III. MODIFIED K-CLUSTERING

We focus on k-means clustering which is a simple technique to group items into k clusters. k-means clustering is an iterative algorithm, which starts off with random cluster centers. A single iteration assigns all objects to the closest clusters based on their distances from the cluster means and then re-computes the cluster means. Iterations are repeated until the algorithm converges to a set of stable clusters. The basic k-means clustering algorithm is given below:

Initialize the k means $\mu_1, \mu_{2...} \mu_k$ to 0.
Arbitrarily select k starting points $\mu'_1, \mu'_{2...} \mu'_k$

Repeat
  Assign $\mu'_1, \mu'_{2...} \mu'_k$ to $\mu_1, \mu_{2...} \mu_k$ respectively
    for all points i do
      Assign point i to cluster j if distance d(i, $\mu_j$ ) is
      the minimum over all j.
    end for
  Calculate new means $\mu'_1, \mu'_{2...} \mu'_k$
until the difference between $\mu_1, \mu_{2,...,} \mu_k$ and $\mu'_1, \mu'_{2,...,} \mu'_k$ is acceptably low.

Algorithm 2. K-means clustering

The results come in two forms: Assignment of entities to clusters, and the cluster centers themselves. We assume that the cluster centers $\mu_i$ are semiprivate information, i.e., each site can learn only the components of $\mu$ that correspond to the attributes it holds. Thus, all information about a site's attributes (not just individual values) is kept private; if sharing the $\mu$ is desired, an evaluation of privacy/secrecy concerns can be performed after the values are known.

At first glance, this might appear simple - each site can simply run the k-means algorithm on its own data. This would preserve complete privacy. Figure 2 shows why this will not work. Assume we want to perform 2-means clustering on the data in the figure. From y's point of view (looking solely at the vertical axis), it appears that there are two clusters centered at about 2 and 5.5. However in two dimensions it is clear that the difference in the horizontal axis dominates. The clusters are actually "left" and "right", with both having a mean in the y dimension of about 3. The problem is exacerbated by higher dimensionality.

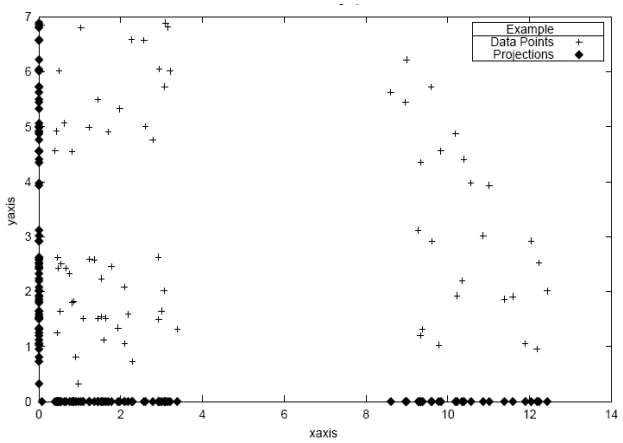


Fig. 2. Two dimensional problems

**Basic approach**

Given a mapping of points to clusters, each site can independently compute the components of $\mu_i$ corresponding to its attributes. Assigning points to clusters, Specifically computing which cluster gives the minimum d(i, $\mu_j$), requires cooperation between the sites. We show how to privately compute this in Section 3.3.2. Briefly, the idea is that site A generates a (different) vector (of length k) for every site (including itself) such that the vector sum of all the site vectors is $\vec{0}$ . Each site adds its local differences |point - $\mu_i$ | to its vector. At the same time, the vector is permuted in an order known only to A. Each site (except a single holdout) sends their permuted vector to site B. Site B sums the received vectors, then the holdout site and B perform a series of secure additions and comparisons to find the minimum i without learning distances. B now asks A the real index corresponding to i, giving the proper cluster for the point.
Securely Finding the Closest Cluster

This algorithm is used as a subroutine in the k-means clustering algorithm to privately find the cluster which is closest to the given point, i.e., which cluster should a point be assigned to. Thus, the algorithm is invoked for every single data point in each iteration. Each party has as its input the component of the distance corresponding to each of the k clusters. This is equivalent to having a matrix of distances of dimension k x r. For common distance metrics; such as Euclidean, Manhattan, or any other Minkowski; this translates to finding the cluster where the sum of the local distances is the minimum among all the clusters. It Require: r parties, k clusters, n points.

1: for all sites $j = 1 \ldots r$ do
2: for all clusters $i = 1 \ldots k$ do
3: initialize $\mu'_{ij}$ arbitrarily
4: end for
5: end for
6: repeat
7: for all $j = 1 \ldots r$ do
8: for $i = 1 \ldots k$ do
9: $\mu_{ij} \leftarrow \mu'_{ij}$
10: $Cluster[i] = \emptyset$
11: end for
12: end for
13: for $g = 1 \ldots n$ do
14: for all $j = 1 \ldots r$ do
15: {Compute the distance vector $\vec{X}_j$ for point $g$.}
16: for $i = 1 \ldots k$ do
17: $x_{ij} = |data_{gj} -_D \mu_{ij}|$
18: end for
19: end for
20: Each site puts $g$ into $Cluster[closest\_cluster]$
21: end for
22: for all $j = 1 \ldots r$ do
23: for $i = 1 \ldots k$ do
24: $\mu'_{ij} \leftarrow$ mean of $j$'s attributes for points in $Cluster[i]$
25: end for
26: end for
27: until $checkThreshold$

Algorithm 2. Privacy preserving k-means clustering

## IV. MODIFIED EXPECTATION MAXIMIZATION (E. M.) CLUSTERING

We present a privacy preserving EM algorithm for secure clustering. Only the one dimensional case is shown; extension to multiple dimensions is straight forward. The convention for notations is given below:

| | |
|---|---|
| k | Total number of mixture components (clusters). |
| s | Total number of distributed sites. |
| n | Total number of data points. |
| nl | Total number of data points for site L. |
| yj | Observed data points. |
| μi | Mean for cluster i. |
| $\sigma^2$i | Variance for cluster i. |
| πi | Estimate of proportion of items in cluster i. |
| zij | Cluster membership. If yj = 1 for cluster i, zij ≈ 1, else zij ≈ 0. |

i, j, l are the indexes for the mixture component, data points and distributed sites respectively. t denotes the iteration step. From conventional EM mixture models for clustering, we assume that data yj are partitioned across s sites ($1 \leq l \leq s$). Each site has nl data items, where summation over all the sites gives n. To obtain a global estimation for $\mu_i^{(t+1)}$, $\sigma_i^{2(t+1)}$, and $\pi_i^{(t+1)}$ the step requires only the global values n and

$$\sum_{j=1}^{n} z_{ij}^{(t)} y_j = \sum_{l=1}^{s} \sum_{j=1}^{n_l} z_{ijl}^{(t)} y_j$$

$$\sum_{j=1}^{n} z_{ij}^{(t)} = \sum_{l=1}^{s} \sum_{j=1}^{n_l} z_{ijl}^{(t)}$$

$$\sum_{j=1}^{n} z_{ij}^{(t)} (y_j - \mu_i^{(t+1)})^2 = \sum_{l=1}^{s} \sum_{j=1}^{n_l} z_{ijl}^{(t)} (y_j - \mu_i^{(t+1)})^2$$

Observe that the second summation in each of the above equations is local. Using secure sum, we can compute the global values securely, without revealing yj. The estimation step giving z can be partitioned and computed locally given global μi, $\sigma^2$i , and πi:

$$z_{ijl}^{(t+1)} = \frac{\pi_i^{(t)} f_i(y_j; \mu_i^{(t)}, \sigma_i^{2(t)})}{\sum_i \pi_i^{(t)} f_i(y_j; \mu_i^{(t)}, \sigma_i^{2(t)})}$$

where yj is a data point at site l. The E-step and M-step iterate until

$$|L^{(t+1)} - L^{(t)}| \leq \epsilon, \text{ where}$$

$$L^{(t)}(\theta^{(t)}, \mathbf{z}^{(t)}|y) = \sum_{j=1}^{n} \sum_{i=1}^{k} \{z_{ij}^{(t)} [\log \pi_i f_i(y_j^{(t)}|\theta^{(t)})]\}$$

Again, this can be computed using a secure sum of locally computed partitions of z.

*A. Quantifying Privacy*

One focus of this project will be to understand and define privacy and security in ways that make sense for data mining. The secure multiparty computation approach has two limitations:

1. It is too restrictive; truly secure solutions may be inefficient. (E.g., for set intersection to be completely secure, each site must send enough data to represent all possible values, even if much is just "dummy" data).
2. It doesn't guarantee privacy. It only guarantees that nothing is disclosed beyond the result, but what if the result itself violates privacy?

We need ways to define and measure privacy to ensure that privacy preserving data mining results do meet actual privacy constraints. Sketches of several approaches are given below.

*B. Knowledge query*

Bounded Knowledge Approaches that alter the data generally use a bounded knowledge definition of privacy, perhaps the best method to date is the entropy based metric of [1]. While secure multiparty computation appears to achieve "perfect" privacy, in that nothing is shared but the results, even the results can provide bounded knowledge on the data sources.

Need to know is well established in controlling access to data. In the U.S., access to classified data requires both a security clearance and a justification of why the data should be accessed.

For Protected from disclosure, we want to protect specific items: individual data items, or specific rules. The problem becomes more difficult when we want to protect against disclosure of classes of information – in the limit this prevents data mining altogether [12]. We will develop privacy measures to address this issue; a likely starting point is ability to learn a classifier for a protected attribute from the results.

## V. PRIVACY-PRESERVING COLLABORATI APPROACH USING FUZZY LOGIC

The works referenced here are most similar in nature to the proposed work. In particular, they make privacy preservation one of the key requirements, and support it to varying degrees.

Anonymity is an established measure of privacy, including concepts such as k-anonymity. We have proposed a p-in distinguish ability metric that extends this concept to data mining, allowing results that reveal information about an individual as long as the results reveal equivalent information for all individuals.The proposed project will formalize these measures and use them to analyze the developed privacy preserving data mining constructs. We will also investigate the applicability of these measures, and identify and formalize new measures as appropriate [15] [18].

Lincoln et al. describe a privacy-preserving mechanism for sharing security alerts, and addresses several techniques to sanitize alert data, including scrubbing and hashing. They propose the use of multiple hash functions, some keyed, to build solutions that avoid dictionary attacks. They also employ multiple repositories that randomly forward alerts to each other to obfuscate event sources. There appears to be no implementation of the above model; however, they conducted some small performance tests of hashing and correlation overhead. As opposed to Lincoln, et al., this thesis is application agnostic. the infrastructure behind Worminator can be applied to other forms of intrusion detection or software fault correlation. It suggests approaches to support privacy preserving collaborative payload anomaly detection. Additionally, this thesis introduces the notion of a framework to enable scalable, heterogeneous privacy preserving mechanisms, while focuses on a fixed basket of techniques. Finally,

Worminator has several significant differences to enable practical deployment, including the use of Bloom filters, fast Bloom filter correlation techniques, and publish-subscribe infrastructures. To the best of my knowledge, the

work proposed in remains unimplemented, and in fact postdates much of the Worminator work [13].

Kissner just completed a thesis proposal titled "Privacy-Preserving Distributed Information Sharing" [14], and some of the results are published in [15]. In the thesis proposal, she outlines two different privacy-preserving mechanisms: a polynomial set representation that supports not only privacy-preserving intersection, but also union and element reduction and a pair of hot item algorithms, one defining an identification mechanism and the other defining a publication mechanism.

Huang et al. describe Privacy-Preserving Friends Troubleshooting Network [16], which extends Wang et al.'s Peer Pressure research a collaborative model for software configuration diagnosis with a privacy preserving architecture utilizing a "friend"-based neighbor approach to collaboration. The key relevant aspects of the paper include a variation of secure multi party computation problem to "vote" on the popularity of a configuration to determine the configuration outlier, and the use of hash functions to enable secure multiparty computation (SMC) to support an unknown set of values; the relation of this proposal to SMC is discussed further in the next section. Finally, as with the previous work, they do not address temporal constraints in their correlation mechanisms [17].

In data mining, the user looks for new knowledge from database, such as relations between variables rules for instance. Data mining in databases or data warehouses in fuzzy domain is not that much easy. The purpose is to find related homogeneous categories, prototypical behaviors, general associations, important features for the recognition of a class of data. In this case, using fuzzy sets brings flexibility in interpretability, knowledge representation in the obtained results.. Looking for strict a relation between variables may be impossible because of the variety of descriptions in the database, while looking for an imprecise relation between variables or to a crisp relation between approximate values of variables may lead to a solution. For Example educated people who saves nation can solved by fuzzy relation. The expressiveness of fuzzy rules and relations or fuzzy values of attributes in a simplified natural language is a major quality for the interaction with the final user in providing the needed information in secured manner.

## VI. CONCLUSION

A set of algorithms and techniques were proposed to solve privacy preserving data mining problems. The algorithms were implemented in java code. The experiments showed that the algorithms perform well on large databases. We introduced the notion of privacy preserving data mining with the primary goal of enabling collaboration of two clustering algorithms using fuzzy logic. It is very much useful in privacy preserving real world applications. This Combinatorial modified approach for secured date privacy is better than other approaches.

## REFERENCES

[1]. Qian Wang, Cong Wang, Kui Ren, Wenjing Lou, and Jin Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing,"IEEE Transactions on Parallel and Distributed Systems (TPDS), Vol. 22, No. 5, pp. 847-859, May, 2011
[2]. Anna Squicciarini, Barbara Carminati, Sushama Karumanchi., "Privacy-preserving Service Selection in Business Oriented Web Service Composition", IEEE International Conference on Web Services (ICWS), July 2011.
[3]. Kun Liu, Hillol Kargupta, and Jessica Ryan, "Random Projection-based Multiplicative Perturbation for Privacy Preserving Distributed Data Mining". IEEE Transactions on Knowledge and Data Engineering (TKDE), VOL. 18, NO. 1, pages 92--106, Piscataway, NJ, January 2006.
[4]. A. C. Squicciarini, E. Bertino, and E. Ferrari. "Achieving Privacy with an Ontology-Based approach in Trust Negotiations". IEEE Transaction on Dependable and Secure Computing (TDSC), Vol. 3 N. 1, pp. 13-30, January-March 2006.
[5]. N. Yvas, A. Squicciarini, Chih-Cheng Chang, D. Yao. "Towards Automatic Privacy Management in Web 2.0 with Semantic Analysis on Annotations". Collaborate Com Conference, IEEE. November 2009.
[6]. A. De Santis, G. Di Crescenzo, R. Ostrovsky, G. Persiano, and A. Sahai. "Robust Noninteractive Zero Knowledge", Advances in Cryptology - CRYPTO 2001, volume 2139 of Lecture Notes in Computer Science, pages 566-598, 2001.
[7]. R. Agrawal and R. Srikant. "Privacy-preserving data mining". In SIGMOD '00, pages 439-450. ACM Press, 2000.
[8]. Y. Lindell and B. Pinkas. "Privacy preserving data mining". Journal of cryptology, 15(3):177-206, 2002.
[9]. Gagan Aggarwal, Tom_as Feder, Krishnaram Kenthapadi, Rajeev Motwani, Rina Panigrahy, Dilys Thomas, and An Zhu. "Approximation algorithms for k-Anonymity". Journal of Privacy Technology (JOPT), 2005.
[10]. Y. Lindell and B. Pinkas, "Privacy Preserving Data Mining, Advances in Cryptology" - CRYPTO 2000. Lecture Notes in Computer Science, Vol. 1880, pp. 36–53Springer-Verlag, 2000.
[11]. Maurizio Atzori, Francesco Bonchi, Fosca Giannotti, and Dino Pedreschi. "k-anonymous patterns". In PKDD '05: Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases, pages 10-21, 2005.
[12]. Maurizio Atzori, Francesco Bonchi, Fosca Giannotti, and Dino Pedreschi. "Blocking anonymity threats raised by frequent itemset mining". In ICDM, pages 561 -564, 2005.
[13]. Patrick Lincoln, Phillip Porras, and Vitaly Shmatikov. "Privacy-Preserving Sharing and Correlation of Security Alerts". In USENIX Security, 2004.
[14]. Lea Kissner. Thesis Proposal: "Privacy Preserving Distributed Information Sharing. PhD thesis", CMU, 2005.
[15]. Lea Kissner and Dawn Song. Privacy-Preserving Set Operations. In CRYPTO, 2005.
[16]. Qiang Huang, Helen J. Wang, and Nikita Borisov. "Privacy-Preserving Friends Troubleshooting Network". In NDSS, San Diego, CA, 2005.
[17]. Helen J.Wang, John C. Platt, Yu Chen, Ruyun Zhang, and Yi-MinWang. „Automatic Misconfiguration Troubleshooting with PeerPressure". In OSDI, San Francisco, 2004.
[18]. Charu C. Aggarwal and Philip S. Yu. "Privacy-Preserving Data Mining: Models and Algorithms". Springer Publishing Company, Incorporated, July 2008.