DATA MINING IN DATABASES STORED OVER THE INTERNET

Benard Mapako
Computer Science Department
University of Zimbabwe
P. O. Box Mt 167
Mt Pleasant
Harare
Zimbabwe
bemapako@gmail.com

Mike Abia
Computer Science Department
University of Zimbabwe
P. O. Box Mt 167
Mt Pleasant
Harare
Zimbabwe
abiamaike@gmail.com

Cross Gombiro
Computer Science Department
Bindura Univ. of Scie. Edu.
P. O. Box 1020
Bindura
Zimbabwe
cgombiro@gmail.com

ABSTRACT

Data mining is a part of knowledge discovery (KDD). Data mining may be described as the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cut costs, or both. Data mining of databases stored over the internet is the application of data mining concepts on databases stored over the internet. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases. One of the biggest challenges of data mining databases stored over the internet is the heterogeneity of data formats in these databases. Emerging forms of data mining are able to perform multidimensional mining on a wide variety of heterogeneous data sources, to provide solutions to many problems. These emerging forms accommodate data mining of databases stored over the internet. In this paper we examine some of the techniques that can be used to mine data/information from databases stored on the internet. We then, propose a technique for efficient and effective data mining of databases stored over the internet.

Key words: Data mining, Data warehouse, Federated Databases, Distributed Database

1. INTRODUCTION

1.1 Background

Data mining in databases stored on the internet deals with the problem of finding data patterns in an environment with databases stored in disparate locations over the internet. The databases may be homogeneous but generally they are heterogeneous. In heterogeneous cases, each database site maintains databases with different kinds of information. Thus, the feature sets observed at different sites are different. This is sometimes called *vertically partitioned dataset* as described in [1]. In this paper we assume that access to the different databases is not inhibited by security measures of the database owners. Data mining is an important technique that organizations wish to use for the acquisition of knowledge on the internet. We therefore, in this paper, examine strategies that may be used to effect data mining on databases stored over the internet.

1.2 Basic Concepts

Data mining, *the extraction of hidden information from large databases*, is a technology whose goals are prediction, identification of the existence of items, classification of data and optimization of resource usage. Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. The automated, prospective analyses offered by data mining move beyond the analyses of past events provided by retrospective tools typical of decision support systems. Data mining tools can answer business questions that traditionally were too time-consuming to resolve. They scour databases for hidden patterns, finding predictive information that experts may miss because it lies outside their expectations.

Data mining can also be done on databases stored over the internet for the same reasons of extracting information. Data mining is part of the knowledge discovery process as illustrated in Fig 1.

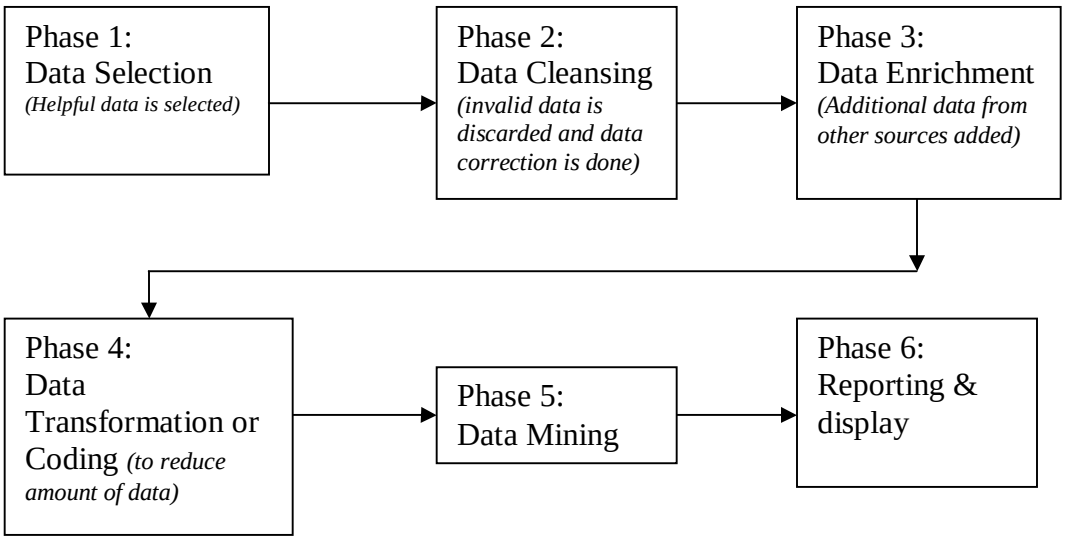

Fig 1: Knowledge discovery process

The result of data mining may be to discover:

- Association rules - for example whenever a student requests a textbook, he/she also requests another literary item.
- Sequential patterns - for example, suppose a customer buys a camera, and within three months he/she buys photographic supplies, then within six months he is likely to buy an accessory item.
- Classification trees – for example, customers may be classified by frequency of visits, by type of financing used, by amount of purchase, or by affinity for types of items and some revealing statistics may be generated for such classes.

The most commonly used techniques in data mining are:

- **Artificial neural networks**: Non-linear predictive models that learn through training and resemble biological neural networks in structure.

- **Decision trees**: Tree-shaped structures that represent sets of decisions. These decisions generate rules for the classification of a dataset. Specific decision tree methods include Classification and Regression Trees (CART) and Chi Square Automatic Interaction Detection (CHAID) .

- **Genetic algorithms**: Optimization techniques that use processes such as genetic combination, mutation, and natural selection in a design based on the concepts of evolution.

- **Nearest neighbor method**: A technique that classifies each record in a dataset based on a combination of the classes of the k record(s) most similar to it in a historical dataset . Sometimes called the k-nearest neighbor technique.

- **Rule induction**: The extraction of useful if-then rules from data based on statistical significance.

Many of these technologies have been in use for many years in specialized analysis tools that work with relatively small volumes of data. These capabilities are now evolving to integrate directly with industry-standard data warehouse and OLAP (On-line Analytical Processing) platforms.

2. RESULTS OF DATA MINING

2.1 Classification Tree

A Hybrid Decision Tree/Genetic Algorithm Method for Data Mining

In the classification task of data mining, the discovered knowledge is often expressed as a set of rules of the form:

IF <conditions> THEN <prediction (class)>.

This knowledge representation has the advantage of being intuitively comprehensible for the user. From a logical viewpoint, typically the discovered rules are expressed in disjunctive normal form, where each rule represents a disjunct and each rule condition represents a conjunct. In this context, a small disjunct can be defined as a rule which covers a small number of training examples .

The concept of small disjunct is illustrated in Fig 2. This figure shows a part of

a decision tree. In this figure we indicate, beside each tree node, the number of examples belonging to that node. Hence, the two leaf nodes at the right bottom can be considered small disjuncts, since they have just one and three examples (instances).

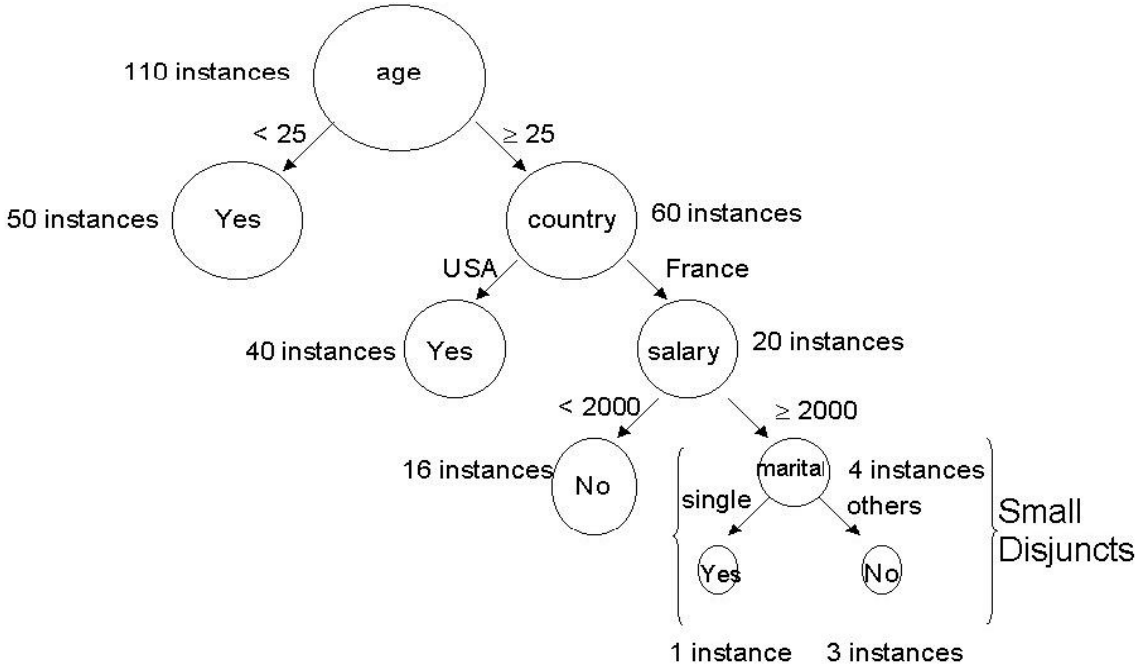

Fig 2: Example of a small disjunct in a decision tree induced from a population data set

The vast majority of rule induction algorithms have a bias that favors the discovery of large disjuncts, rather than small disjuncts. This preference is due to the belief that it is better to capture generalizations rather than specializations in the training set, since the latter are unlikely to be valid in the test set.

Note that classifying examples belonging to large disjuncts is relatively easy. For instance, in Fig 2, consider the leaf node predicting class "Yes" for examples having Age < 25, at the left top of the figure. Presumably, we can be confident about this prediction, since it is based on 50 examples. By contrast, in the case of small disjuncts, we have a small number of examples, and so the prediction is much less reliable. The challenge is to accurately predict the class of small disjunct examples.

At first glance, perhaps one could ignore small disjuncts, since they tend to be error prone and seem to have a small impact on predictive accuracy. However, small disjuncts are actually quite important in data mining and should not be ignored. The main reason is that, even though each small disjunct covers a small number of examples, the set of all small disjuncts can cover a large number of examples. In such cases we need to discover accurate small-disjunct rules in order to achieve a good classification accuracy rate.

Our approach for coping with small disjuncts consists of a hybrid decision tree/genetic algorithm method. The basic idea is that examples belonging to large disjuncts are classified by rules produced by a decision-tree algorithm, while examples belonging to small disjuncts are classified by a genetic algorithm (GA) designed for discovering small-disjunct rules. This approach is justified by the fact that GAs tend to cope better with attribute interaction than conventional greedy decision-tree/rule induction algorithms , and attribute interaction can be considered one of the main causes of the problem of small disjuncts.

2.2 Association Rules

Discovery of Association Rules

In the standard form of this task (ignoring variations proposed in the literature) each data instance (or "record") consists of a set of binary attributes called items. Each instance usually corresponds to a customer transaction, where a given item has a true or false value depending on whether or not the corresponding customer bought that item in that transaction.

An association rule is a relationship of the form IF *X* THEN *Y*, where *X* and *Y* are sets of items. An example is the association rule:

IF *fried_potatoes* THEN *soft_drink*, *ketchup* .

Although both classification and association rules have an IF-THEN structure, there are important differences between them. We briefly mention here two of these differences.

First, association rules can have more than one item in the rule consequent, whereas classification rules always have one attribute (the goal one) in the consequent. Second, unlike the association task, the classification task is asymmetric with respect to the predicting attributes and the goal attribute. Predicting attributes can occur only in the rule antecedent, whereas the goal attribute occurs only in the rule consequent.

2.3 Sequential Patterns

Discovery of Sequential Patterns

The discovery of sequential patterns is based on the concept of a sequence of itemsets [2]. The ordering yields a sequence of itemsets. The problem of identifying sequential patterns is to find all subsequences from the given sets of sequences that have a user defined minimum support. The sequence $S_1, S_2, S_3, \ldots$ is a predictor of the fact that, say a client who requests itemset S_1 is likely to request S_2 and then S_3 and so on. This prediction is based on the frequency (support) of this sequence in the past.

3. CHALLENGES OF DATA MINING INTERNET BASED DATABASES

The effective use of data from disparate databases presents several challenges in practice. These challenges are explained in the following sections.

3.1 Size of Databases

Data repositories are large in size, dynamic, and physically distributed. Consequently, it is neither desirable nor feasible to gather all of the data in a centralized location for analysis. Hence, there is a need for algorithms that can efficiently extract the relevant information from disparate databases on demand.

3.2 Autonomous Ownership

Databases of interest are autonomously owned and operated. Consequently, the range of operations that can be performed on the database (e.g., the types of queries allowed), and the precise mode of allowed interactions can be quite diverse. Hence, strategies for

obtaining the necessary information (e.g., statistics needed by data mining algorithms) within the operational constraints imposed by the database are needed.

3.3 Heterogeneity of the Structure of Databases

Databases are heterogeneous in structure (e.g., relational databases, object databases, etc) and content. Each data source implicitly or explicitly uses its own ontology concepts, attributes and relations among attributes) [3] to represent data.

Effective integration of information from different sources bridging the syntactic and semantic mismatches among the data sources is needed.

3.4 User Views

In many applications (e.g., scientific discovery), because users often need to examine data in *different contexts from different perspectives*, there is no single universal ontology [3] that can serve all users, or for that matter, even a single user, in every context. Hence, methods for context-dependent dynamic information extraction and integration from distributed data based on user-specified ontologies are needed to support knowledge acquisition and decision making from heterogeneous databases on the internet.

4. REQUIREMENTS FOR DATA MINING ON THE INTERNET

The basic requirements that have to be met when data mining databases on the internet are explained in the sections that follow.

4.1 Compatibility with existing infrastructures

The internet has varied infrastructure that need to be considered in coming up with techniques for data mining databases on the internet. It is through compatibility that a technique can be integrated with existing techniques, infrastructure, protocols, etc.

4.2 Openness to tools and algorithms.

The techniques must be open to the integration of new data mining tools and algorithms. This enhances effective and comprehensive data mining in that it empowers the technique to use the powers vested in tools and algorithms that are not part of it.

4.3 System, network, and location transparency.

Users should be able to run their data mining applications in an easy and transparent way, without needing to know details of the network features and physical location of data sources.

4.4 Security and data privacy.

Security and privacy issues are vital features in a lot of data mining techniques. The technique must offer valid support to cope with user authentication, security and privacy of data.

4.4.1 Secure Multiparty data mining

The basic idea of multi party data mining is that data mining is secure if at the end of the computation no party knows anything except its own input and the results. A trusted party may be used or there has to be some communication between the databases on the internet. If the communication between the data sources is not to disclose anything then we have to allow non-determinism in the exact values send in the intermediate communication. (e.g. encrypt with a randomly chosen key) and demonstrate that a party with just its own input and the result can generate a predicted intermediate computation that is as likely as the actual values. This has been shown possible [4].

5. TECHNIQUES FOR DATA MINING ON THE INTERNET

Database Integration

Data mining techniques attempt to provide users with seamless and flexible access to information from multiple autonomous, distributed and heterogeneous data sources through a unified query interface. Ideally, a data mining technique should allow users to

specify *what* information is needed without having to provide detailed instructions on *how* or from where to obtain the information. Thus, in general, a data mining technique must provide mechanisms for the following:

a) Communications and interaction with each database on the internet as needed.

b) Specification of a query, expressed in terms of a user specified vocabulary (ontology), across multiple heterogeneous and autonomous databases

c) Specification of mappings between user ontology and the database specific ontologies.

d) Transformation of a query into a plan for extracting the needed information by interacting with the relevant database.

e) Integration and presentation of the results in terms of a vocabulary known to the user. This is not part of data mining as described in *Fig: 1* above, but it is necessary to have results presented to the user. There are two broad classes of approaches to data integration: *Data Warehousing* and *Database Federation* [4].

5.1 Data Warehousing

In the data warehousing approach, data from heterogeneous internet databases (databases stored over the internet) is gathered, mapped to a common structure and stored in a central location. In order to ensure that the information in the warehouse reflects the current contents of the individual sources, it is necessary to periodically update the warehouse. *Fig: 3* shows the very simplified layout of components in the data warehousing approach.

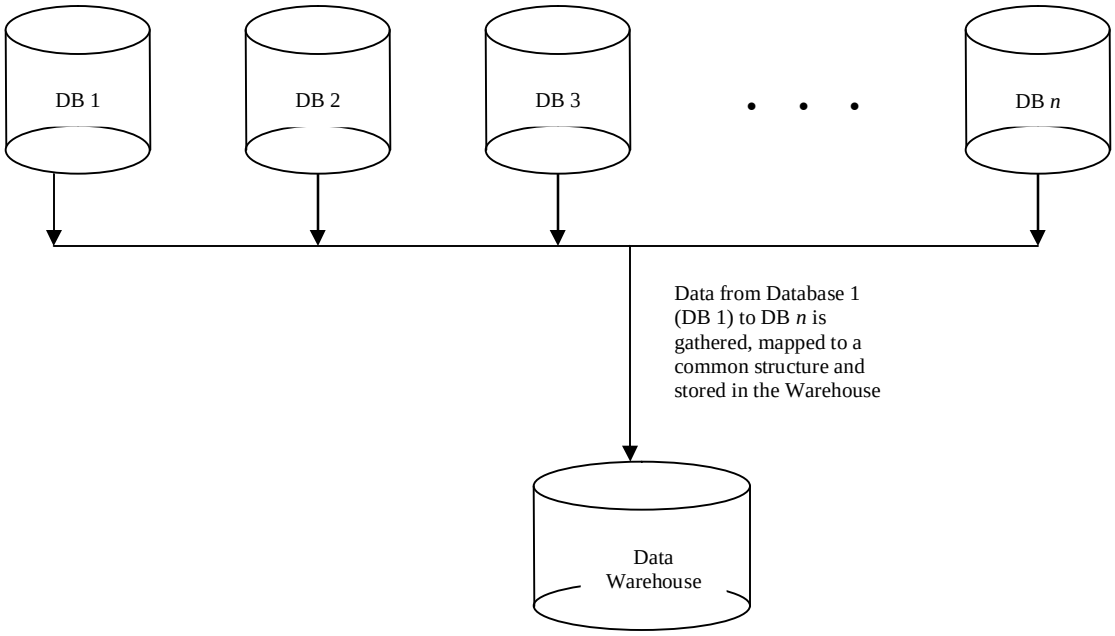

Fig 3: Data Warehousing approach to data mining databases stored over the internet.

In the case of large information repositories, this is not feasible unless the individual information sources support mechanisms for detecting and retrieving changes in their contents. This is often an unreasonable expectation in the case of autonomous databases. The warehousing approach to data integration has another important drawback in the case of applications such as scientific discovery in which users often need to analyze the same data from multiple points of view: The data warehouse relies on a single common ontology for all users of the system. This ontology is typically specified as part of the design of the data warehouse. Each user queries the warehouse using a common vocabulary and a common query interface.

5.2 Database Federation

The federated database architecture interconnect databases to minimize central authority yet supports partial sharing and coordination among database systems. A federated database is sometimes called a virtual database [5] and also [6]. In the case of Database Federation, information needed to answer a query is gathered directly from the data sources in response to the posted query. Hence, the results are up-to-date with respect to

the contents of the data sources at the time the query is posted. More importantly, the database federation approach lends itself to be more readily adapted to applications that require users to be able to impose their own ontologies on data from distributed autonomous databases [7].

Typically, a query posted by the user must be decomposed into a set of operations corresponding to the information that needs to be gathered from each database and the form in which this information must be returned to the querying system. To accomplish this, data integration techniques must support two basic set of operations in one form or the other:

• *get*( ) to query the database; and

• *transform*( ) for mapping the results in the desired

form.

5.2.1 Data Federation Technique (Example)

In order to demonstrate the Data Federation Technique in data mining, we are going to use an example that is important in that it clarifies the concept and not that it is of grand importance.

Federated data mining algorithms frequently calculate the sum of values from individual databases. Assuming three or more parties and no collusion, the following method computes such a sum.

Assume that the value $v = \sum_{i=1}^{s} v_i$ to be computed is known to lie in the range [*0..n*].

One database site is designated the master site, number 1. The remaining database sites are numbered *2..s*. Database site 1 generates a random number X, uniformly chosen from [*0..n*]. Site 1 adds this to its local value v_1 and sends the sum $X + v_1 \bmod n$ to site 2. Since the value X is chosen uniformly from [*0..n*] the value $X + v_1 \bmod n$ is also distributed uniformly across this region. So site 2 learns nothing about the actual value of v_1.

For the remaining sites $i = 2..s-1$, the algorithm is as follows.

Site i receives $V = X + \sum_{j=1}^{i-1} v_j \bmod n$ since this value is uniformly distributed over [1..n], i learns nothing. Site i then computes $X + \sum_{j=1}^{i} v_j \bmod n = (v_j + V) \bmod n$ and passes it to site $i+1$. Site s performs the above step and sends the result to site 1. Site 1 knowing X can subtract X to get the actual result. Note that site 1 can also determine $\sum_{i=2}^{s} v_i$ by subtracting v_1. This is possible from a global result regardless of how it is computed. So site 1 has learned nothing from the computation. *Fig: 4* depicts how this method operates.

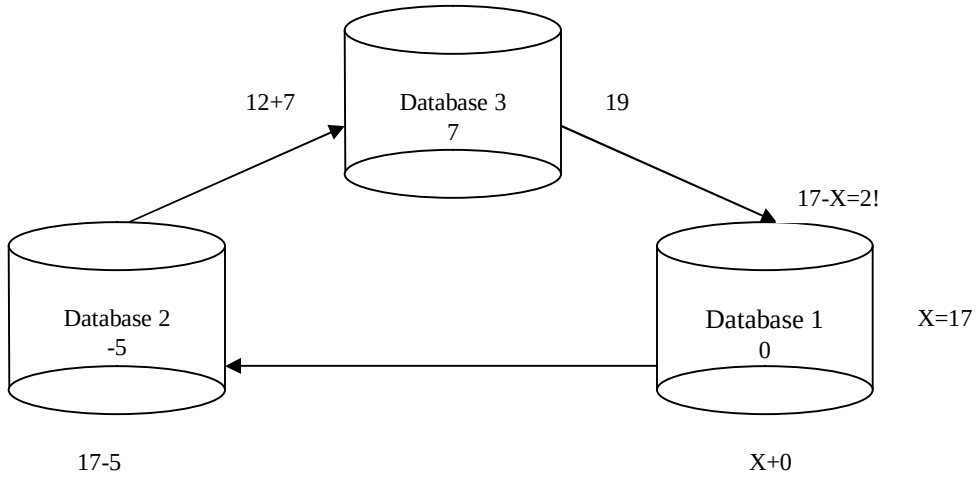

Fig 4: Sum from federated databases

This idea may be implemented in cases where a pattern, association or classification data mining result is suspected and each database site on the internet provides its share of the information needed. An example is that, banking details of a certain individual may be of interest and mining that information becomes a goal. Given a hypothetical case where banks have their databases stored on the internet, then each database is mined separately determining whether the individual of concern has an account with the bank. If an account exists, details about the account are obtained. The same is done with the other banks. It should be noted that this example does not highlight security issues that need to be taken into consideration. This is because it is possible to consider them according to [8] and also [9].

6. CONCLUSION AND FURTHER WORK

Heterogeneous data sites are common in many businesses, government, military and scientific information processing environments. Data mining of databases stored on the internet must develop a well grounded approach to have sound recognition and wide-spread effective use. The techniques presented here show possibilities that may be expanded on without compromising the quality of solutions or knowledge acquired. Data warehousing and Database federation are the techniques presented. Several approaches can be employed in line with these techniques. In [6] it is mentioned that a Database Administrator in the respective federated databases decided on what information is to be accessed during the knowledge discovery process hence data mining process. This may be inhibitive on the data mining quality of results obtained. Further work needs to be done to find a way of making federated databases provide as much information as needed for effective data mining without compromising the security of the database and its owners.

7. REFERENCES

[1] *Collective data mining: A New Perspective Towards Distributed Data Mining – Hillol Kargupta, Byung-Hoon Park, Daryl Hershberger and Erick Johnson (School of Electrical Engineering and Computer Science; Washington State University)*

[2] *R. Elmasri, S.B. Navathe -Fundamentals of Database Systems.*

[3] *Knowledge Representation: Logical, Philosophical, and Computational Foundations. New York: PWS Publishing Co. - Sowa, J. (1999)*

[4] *How to generate and exchange secrets IEEE Symposium-1986 - A.C. Yao*

[5] *www.en.wikipedia.org*

[6] *Towards Interoperability in Heterogeneous Database systems – A. Zisman, J. Kramer (Department of Computing; Imperial College; London; UK-1995)*

[7] *An Intelligent System For Identifying and Integrating Non-Local Objects In Federated Database Systems – Joachim Hammer, Dennis McLeod and Antonio Si (Computer Science Department; University of South California; Los Angeles; USA)*

[8] *Tools for Privacy Preserving Data Mining - C. Clifton, M. Kantacioghi, X. Lin, M. Y. Zhu – Purdue University*

[9] *How to Generate and Exchange Secrets - A.C. Yao*