Mohamed A. Ouda, Sameh A. Salem, Privacy-preserving data mining of distributed databases usingnaïve bayes classifier, pp. 1581 - 1594

PRIVACY-PRESERVING DATA MINING OF DISTRIBUTED DATABASES USING NAÏVE BAYES CLASSIFIER

Mohamed A. Ouda [1,*] , Sameh A. Salem [2]

Depart. of Communication and Computer, Faculty of Engineering Helwan University, Cairo – Egypt

Received 21 March 2013, accepted 29 April 2013

ABSTRACT

Privacy-preserving data mining is discovering accurate patterns and rules without precise access to the original data. In this paper, we propose a novel algorithm for privacy preserving data mining. The proposed algorithm is based on the integration of RSA public key cryptosystem and homomorphic encryption scheme. No data is shared between distributed parties except the final result. Data mining algorithm is performed locally for each party. The final result of all parties is compared to get the target value. Previous solution for privacy preserving data mining of Naive Bayes classifier is based on secure sum that may permit collusion between parties, which is not here in proposed solution. Theoretical analysis and experimental results show that the proposed algorithm can provide good capability of privacy preserving, accuracy and efficiency.

Keywords: privacy preserving, Naive Bayes classifier, distributed databases, secure multiparty computation.

1. Introduction

Data mining is a well-known technique for extracting information or knowledge in a form of classification patterns and association rules as well as cluster analysis from large amount of data. One of the main tasks of data mining techniques is classification and prediction. The goal of classification as a predictive model is to predict the value of a single nominal variable based on the known values of other variables. Loosely speaking classification is the task of assigning objects to one of predefined categories. There are many applications which use classification model such as medical diagnosis, credit approval and identification of high risk customers for insurance companies.

Pattern classification [1] can be described generally as follows: given N training instances (or samples) with known class labels C, e.g., $C = \{c_1, \dots, c_m\}$, how to predict the class label of an unknown instance? That is to say, the purpose of pattern classification is mainly to predict or mark unknown instances with predefined labels in the light of the historical behaviors.

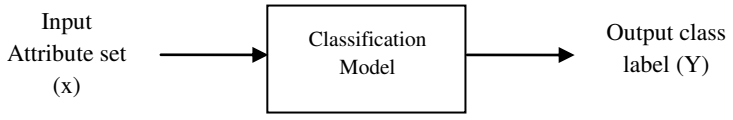

Fig. 1: Classification model: an illustration

Pattern classification has been extensively studied and many outstanding and different kinds of classification algorithms, such as k-nearest neighbor learning algorithm (KNN)

* Corresponding Author.
E-mail address: mohamed_awad099@yhoo.com

[2-3], ID3 decision tree[4], and C4.5[5], have been developed. Among them, the Naïve Bayes classifier [6] which is simple but efficient base line classifier that based on applying Bayes' theorem and uses the simplifying assumption of attribute independence. It is simple to implement and use while giving surprisingly good results.

Privacy concern for classification pattern of distributed data among different parties is an important issue. Privacy concern may prevent different parties from sharing their data and conduct data mining model. So, privacy preserving data mining has becoming an increasing important field of research. The task of running data mining algorithms over multiple data sources without revealing any information other than the output of the algorithm to other sources is often referred to as privacy preserving data mining. In this paper we develop a solution for privacy preserving Naïve Bayes classifier based on RSA public key cryptosystem and homomorphic encryption scheme. The goal of this solution is to have a simple, efficient and privacy preserving classifier and overcome the problem of collusion for privacy preserving Naïve Bayes classifier based on secure sum which is introduced in [7].

The paper is organized as follows: Literature survey in Section 2 presents a related work and the building blocks for privacy preserving which are used. Section 3 introduces the proposed algorithm. Section 4 shows the experimental results and discussion on the work done for three different data sets applied to the proposed algorithm and finally the conclusion.

2. Related work

At present privacy-preserving data mining methods can be roughly divided into two approaches. One approach is called distortion technique or random perturbation technique [8]. The second approach uses cryptographic tools to build data miner pattern and the data are distributed between two or more sites. This approach was first introduced to the data mining community by Lindell and Pinkas[9], with a method that enabled two parties to build a decision tree without either party learning anything about the other party's data, except what might be revealed through the final decision tree.

So far many secure protocols have been developed for data mining and machine learning techniques such as[10-11] for decision tree classification, [12-14] for clustering, [15], [16] for association rule mining, [17-19] for Neural Networks, and [20-21] for Bayesian Networks.

2.1. Naïve bayes classifiers

Bayesian classifier is a popular technique used in many real world applications such as medicine and financial systems. Many applications such as medical symptoms and diagnosis, fraud detection and financial systems use this model to predict class events.

Bayesian classifier is based on Bayes' theorem. Bayesian classifiers classification model is obtained by applying a relatively simple method to a training dataset [22]. To understand how the Bayesian Classifier works, let us consider $x_1,...,x_n$ are attributes with discrete values used to predict discrete class $C = \{v_1, v_2, \dots, v_k\}$. The learner asked to predict classification value for new instance with attribute values a1 through an, the optimal prediction is class value v_j within finite set C such that

$V_{NB} = \text{argmax}_{v_j \in C} (Pr(C=v_j \mid x_1= a_1, x_2= a_2,\dots, x_n= a_n)$.

Using Bayes' theorem, $VNB = \text{argmax}_{v_j \in C} \frac{\Pr(x_1 = a_1, x_2 = a_2, \ldots, x_n = a_n | C = v_j)}{\Pr(x_1 = a_1, x_2 = a_2, \ldots, x_n = a_n)} \Pr(C = v_j)$ (1)

Since $\Pr(x_1 = a_1, x_2 = a_2, \ldots, x_n = a_n)$ is invariant for each class value v_j, then it can be dropped from equation (1) leading to the expression used by Naïve Bayes (NB) classifiers, assuming that attribute values are conditionally independent:

$$V_{NB} = \text{argmax}_{v_j \in C} (Pr(C = v_j \mid x_1 = a_1, x_2 = a_2, \ldots, x_n = a_n). \quad (2)$$

Where, VNB denote the target value output by Naïve Bayes classifier. Probabilities are computed differently for nominal and numeric attributes. For nominal attribute X, that has possible attribute r values $x_1, \ldots, x_r$ the probability $P(X = x_k | C = v_j) = \frac{n_j}{n}$ (3)

where n is the total number of training examples for which $C = v_j$ and n_j is the number of those training examples which also have $X = x_k$. For a numeric attribute, in the simplest case the attribute is assumed to have a normal or Gaussian probability distribution,

$$N(\mu, \sigma^2) = f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \quad (4)$$

The mean μ and variance σ^2 are calculated for each class and each numeric attribute from the training data set. The required probability that the instance is of the class c_j $P(X = x' \mid c_j)$, can be estimated by substituting $x = x'$ in Eqn.4.

An instance is classified as in Eqn.2. Thus the conditional probability of a class given the instance is calculated for all classes, and the class with the highest relative probability is chosen as the class of query instance.

2.2. Building blocks for privacy preserving

2.2.1. Notion of security

An encryption system is called secure if knowing the encrypted message does not give any partial information about message that is not known beforehand. Since the adversary is assumed to run in polynomial time, public-key encryption system [23] is secured due to existence of trapdoor permutations which has the hardness of solving some problem (e.g. factoring large integers). However Goldwasser and Micali [24] introduced the notion of semantic security in formal way. Then Goldreich [25] refined the notion of semantic security.

2.2.2. Secure multi- party computation (SMC)[26]

The aim of a secure multiparty computation task is for the participating parties to securely compute some functions of their distributed and private inputs. There are two properties for secure computation in SMC, privacy and correctness.

Privacy would be to require that each party learns nothing about the other parties' inputs, even if it behaves maliciously, the only information learned by the parties is that specified by the function output. Correctness means that each party is guaranteed that the output it receives is correct.

For a general case, let a set of n parties with private inputs $x_1, \ldots, x_n$ wish to jointly compute a function f of their inputs, this joint computation should have the property that the parties learn the correct output $y = f(x_1, \ldots, x_n)$ and nothing else, and this should hold even if some of the parties maliciously attempt to obtain more information.

The term privacy-preserving under the context of this paper is related to the security definition of Secure Multi-party Computation (SMC). Details of the security definitions and underlying models can be found in [27].

2.2.3 Digital envelope

A digital envelope [28] is a method to hide the private data. That is by conducting a set of mathematical operations between a random number (or a set of random numbers) and the private data. The mathematical operations could be addition, subtraction, multiplication, etc. For example let À be a private data, and R is a random which is only known by the owner of À. The owner can hide À by adding this random number, e.g., À + R.

2.2.4. Homomorphic encryption and decryption scheme

Homomorphic encryption [29] ensures that the computation result on two or more encrypted values is exactly the same as the encrypted result of the same computation on two or more unencrypted values e.g.$E(x) * E(y) = E(x *' y)$, where $*$ and $*'$are two different algebraic operations. In this part, we proposed an additively homomorphic encryption and decryption scheme, which is in[30].

3. The proposed algorithm

The proposed model in this paper consists of two levels (i.e. local and global) and three steps which are demonstrated in figure 2. Steps are as follows:

1- Local classifier (at distributed data in each site).
2- Extraction of local properties.
3- Determining global model based on the local models.

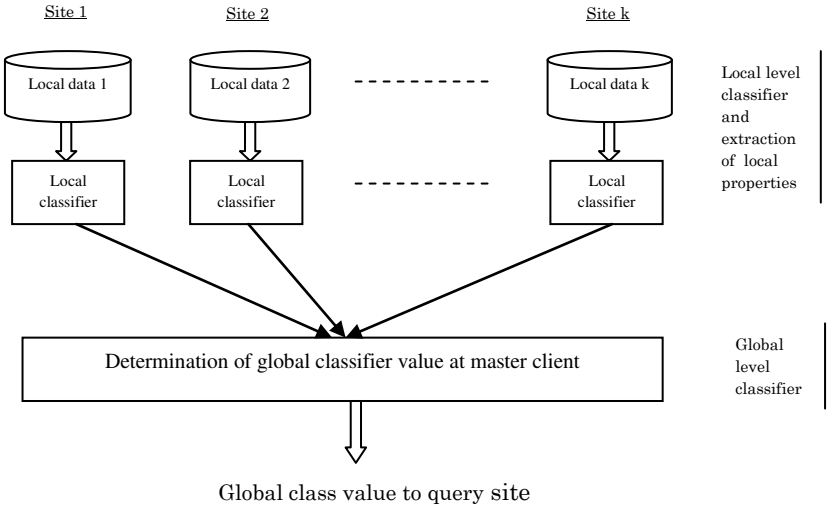

Fig. 2. Steps of classification at distributed databases

The mathematical model of distributed data sets over horizontal partition is as follows: Suppose DB_i $(1 \le i \le k)$ among the data sets DB_1, DB_2, … , DB_k located at different sites P_1,P_2,…,P_k (i.e. k- divisions) as the partial database and $DB = DB_1 \cup DB_2 \cup \ldots \cup DB_k$ as

the overall situation database. Suppose that the database DB_i has l attributes and r class values. Each DB_i has different number of entities. A preprocessing work is done for normalization of training and test data before implementation the proposed algorithm.

We also assume that the adversary model is semi- honest in which parties follow the execution requirement of the protocol but may use what they see during the execution to compute more than they need to know.

A key result which is also used in this work is the composition theorem. We state it for the semi-honest model: Suppose that function g is privately reducible to f and that there exists a protocol for privately computing f. Then there exists a protocol for privately computing g. Loosely speaking the composition theorem states if a protocol consists of several sub-protocols, and can be shown to be secure other than the invocations of the sub-protocols, if the sub-protocols are themselves secure, then the protocol itself is also secure. A detailed discussion of this theorem, as well as the proof, can be found in [31].

The proposed algorithm presents a method for privately computing data mining process from distributed sources without disclosing any information about the sources and their data except that revealed by final classification result. The proposed algorithm develops a solution for privacy–preserving Bayesian classifier [22] which is one of the data mining tasks.

We consider that each site/partyhas its own data mining process independently (all data in each site/party are assumed horizontally partitioned (homogenous data)). The distributed algorithm is determining which of the local results are the closest globally and finding the class of maximum weight of global Bayesian classifier. It is required to protect the privacy of the data sources i.e. a party P_i is not allowed to learn anything about any of the data of the other parties, also collusion with other parties to reveal information about the data is not allowed.

The basic idea is that public encryption key e_i is sent from the master client to every party/site. Each site finds its own Bayesian classifier, then scramble and encrypt the local Naïve Bayes probability with homomorphic encryption using the methodology presented in [30]. This methodology is given in equation 5,6 for encryption and decryption respectively.

3.1. Encryption algorithm

1) At master client the encryption and decryption keys (e_i, d_i) RSA algorithm are generated.
2) Each party /site uses a large number N_i , such that, $N_i = T_i \times Q_i$, where T_i and Q_i are large security prime numbers. In addition, a random number R_i is generated at each site within the uniform distribution $(1, Q_i)$.
3) Given V_{NBi}, which is a plaintext message, the encrypted value is computed as:

$$E_i(V_{NBi}) = mod\big((V_{NBi} + T_i \times R_i), N_i\big) \quad (5)$$

Where mod ( ) is a common modulo N_i – operation. In addition using public key encryption e_i, the parameter T_i and class label y_i are encrypted as follows:

$C_i = E_{ei}(T_i)$ and, $Cl_i = E_{ei}(y_i)$, where C_i and Cl_i are the ciphers of T_i and y_i respectively.

3.2. Decryption algorithm

Given $E_i(V_{NBi})$, which is a ciphertext message of V_{NBi}, we use the security key d_i to recover plaintext $T_i = D_{d_i}(C_i)$;

$$V_{NBi} = D_{T_i}(E_i(V_{NBi})) = mod(E_i(V_{NBi}), T_i) \quad (6)$$

And the corresponding class $y_i = D_{d_i}(Cl_i)$;

The results of V_{NBi} , $i = 1 \dots k$ parties from all sites are then combined to produce the permutation table as in table 1 at master client and the instance with maximum weight of V_{NBi} with its class is determined as the class of querying instance which is transferred to querying site. Each site learns nothing about other sites.

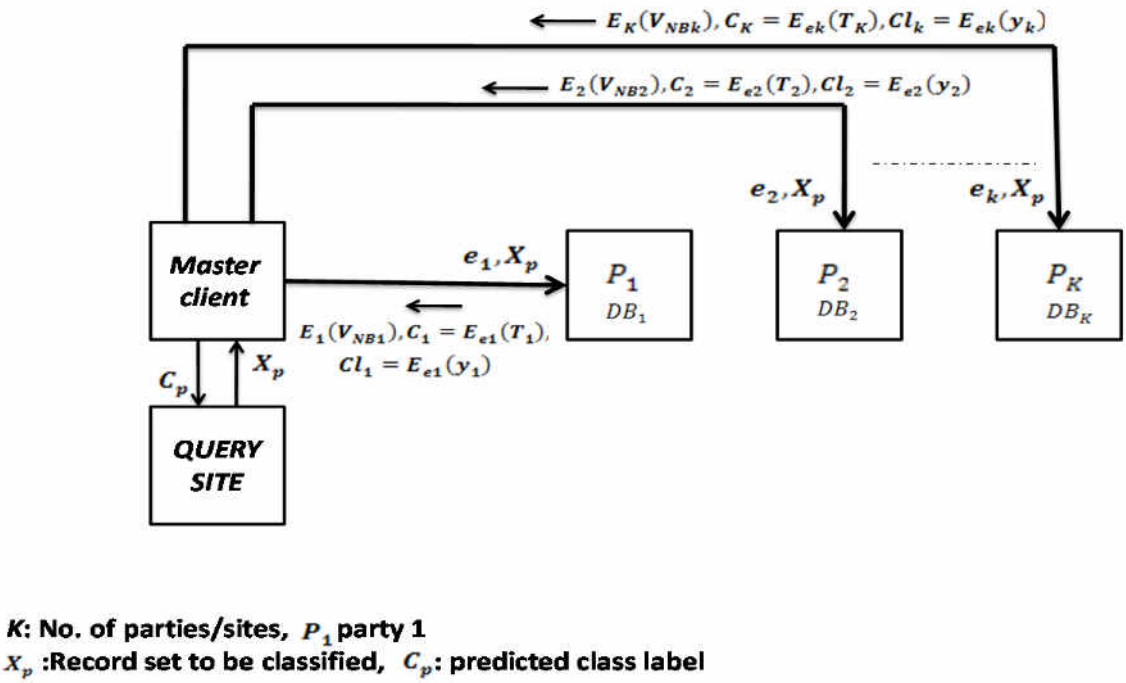

Fig. 3. Naive Bayes Classifier process using proposed algorithm: an illustration

3.3. Proof of encryption decryption scheme

Every party will generate a different value of R_i and N_i

1- $N_1 = T_1 * q_1$, $R_1 \in (1, T_1)$, T_1, q_1 are prime large numbers , R_1 is a random number

$E_1(x_1) = x_1 + mod((R_1 * q_1), N_1)$, $R_1 * q_1 < N_1$, $x_1 < q_1$ (x_1 is private data)

$D(E_1(x_1)) = mod(E_1(x_1), q_1)$

$$= mod((x_1 + mod((R_1 * q_1), N_1)), q_1)$$
$$= mod(x_1, q_1) + mod(mod((R_1 * q_1), N_1), q_1)$$
$$= x_1 + mod((R_1 * q_1), q_1) = x_1 + 0 = x_1$$

2- $N_2 = T_2 * q_2$, $R_2 \in (1, T_2)$, T_2, q_2 are prime large numbers, R_2 is a random number

$E_2(x_2) = x_2 + mod((R_2 * q_2), N_2)$, $R_2 * q_2 < N_2$, $x_2 < q_2$ (x_2 is private data)

$$D(E_2(x_2)) = mod(E_2(x_2), q_2) = x_2$$
$$= mod(x_2, q_2) + mod(mod((R_2 * q_2), N_2), q_2)$$
$$= x_2 + mod((R_2 * q_2), q_2) = x_2 + 0 = x_2$$

The same way if we have N_i,T_i,q_i,R_i are generated at local party P_i will get x_i at master client according to the encryption decryption technique used. Since q_i is generated at the local party and sent to the master client for decryption scheme.

3.4. Proposed algorithm

Require: k parties, r class values, l attribute values, X query instance $X= x_1, x_2, \ldots, x_l$

1: {(e_i, d_i) represent the encryption and decryption keys of RSA algorithm are generated at master client}

2: Master client generates encryption-decryption keys (e_i, d_i) using RSA Algorithm and Transport e_i to each Party P_i;

3: ***for*** *each party* P_i $i = 1 \ldots k$ ***do*** // scan k parties

4: ***for*** each class value y_j , j =1…r in each party P_i do // scan class values in dataset of party P_i

for each attribute value x_m m=1…l of query instance X having class value y_j in a dataset of party P_i

do

Calculate the probability $P_r(x_m|y_j)$ as per Eqn. (3), or Eqn. (4) for nominal and numeric attribute

respectively;

Calculate the probability in a dataset that a class C has value y_j: $P_r(C = y_j)$;

Calculate the probability of party $P_i$having class y_j :

$P_{iy_j} = P_r(C = y_j) \prod_{m=1}^{l} P_r(x_m|y_j)$;

end for // attributes values

end for // class values

5: calculate Naïve Bayes Probability $V_{NBi}= Max\ P_{iy_j}$, then determine class label y_i corresponding to Naïve Bayes probability V_{NBi}

6: **Encryption scheme of V_{NBi} and its class label** y_i

- Choose two large security primes T_i, Q_i, $N_i = T_i \times Q_i$
- Generate a random number R_i within the uniform distribution (1, Q_i)
- Encrypt V_{NBi} as per equation (5) and get $E_i(V_{NBi})$
- RSA encryption for T_i & class label y_i: $C_i=E_{ei}(T_i)$, $Cl_i=E_{ei}(y_i)$

- Transport $E_i(V_{NBi})$, C_i, and Cl_i to master client

7: ***end for*** //k parties

8: **At master client: from k parties, a decryption scheme is performed to get T_i , V_{NBi} and y_i**

for each party P_i , $i = 1$ ... k do //k parties

- $T_i = D_{di}(C_i)$, $y_{ij} = D_{di}(Cl_i)$
- Decrypt $E_i(V_{NBi})$ as per equation (6) and get V_{NBi}

end for //k parties

- Construct the mapping table that maps the relative difference between V_{NBi} with all V_{NBj} $\{i \neq j \ \& \ i, j \in (1, k)\}$ to $1, -1$
- Calculate the weight for each row in the mapping table by adding the row elements and get the sum.
- Determine the global max probability which corresponds to max weight in the mapping table.
- Get the predicted class that match global max probability (max weight in the mapping table).

3.5. How to compute global value through Naïve Bayes Classifier

Every client/site broadcast its encrypted local Naïve Bayes probability V_{NBi} , and corresponding class label y_{ij} to master client (as a third trusted party TTP). Master client after decryption each V_{NBi} and its class label, has a sequence of V_{NBi} , $1 \leq i \leq k$ (for k parties) which uses to construct the permutation mapping table. To construct a mapping table we compare every value in the sequence with other values and if the result is equal or greater than zero the result inserted in the permutation mapping table will be $+1$ otherwise will be -1, e.g if $V_{NB1} - V_{NB2} \geq 0$ the value in the mapping table is +1 otherwise is -1. As an example, let us have the sequence $V_{NB1}, V_{NB2}, V_{NB3}$ and V_{NB4} of four parties/sites and $V_{NB2} < V_{NB1} < V_{NB4} < V_{NB3}$ then $(V_{NB1-} V_{NB2})$, $(V_{NB1-} V_{NB3})$, $(V_{NB1-} V_{NB4})$, $(V_{NB2-} V_{NB3})$, $(V_{NB2-} V_{NB4})$, $(V_{NB3-} V_{NB4})$ will be $\{+1, -1, -1, -1, -1, +1\}$ as presented in Table 1. The weight for any element in the sequence relative to the others is the algebraic sum of the row corresponding to that element. Since V_{NB3} has the largest weight +4, then its corresponding class label will be the predicted value of query instance.

Table 1.
An example of a permutation mapping table

	V_{NB1}	V_{NB2}	V_{NB3}	V_{NB4}	weight
V_{NB1}	+1	+1	-1	-1	0
V_{NB2}	-1	+1	-1	-1	-2
V_{NB3}	+1	+1	+1	+1	+4
V_{NB4}	+1	+1	-1	+1	+2

The proposed algorithm can also be analyzed in many different aspects as follows:

3.5.1 Privacy preserving analysis

Since all the model parameters are completely present with all parties then evaluation can be performed easily. The party that wants to classify an instance using Naïve Bayes evaluation procedure can do that locally, so no interactions between parties. Thus there is no question of privacy being revealed or compromised. We reduce the problem to that of privately computing smaller sub problems and show how to compose them together in order to obtain a complete Naive Bayes solution. This composition is shown to be secure in [24].

Each Party P_i encrypts its output probability V_{NBi} as per equation (5) and the other parameters are encrypted in RSA public key encryption.

$Cl_i = E_{ei}(y_i)$; Cl_i is a cipher encryption of class label, $C_i = E_{e_i}(T_i)$; C_i is a cipher encryption of prime number T_i, e_i is a public key Encryption of party P_i . These encrypted values are transmitted to the master client for global classification. So the output of each party is securely transmitted to the master client to compute the global classification without leaking any information about the private data of a party except its output.

3.5.2. Accuracy of proposed algorithm

Master client, which decrypts, V_{NBi} and its class label y_i produce accurate results with RAS and homomorphic cryptosystem. As shown in Tables 3,4, and 5 the accuracy of the classifier for parties between 2 to 6 is 75.4 – 92.4% . As shown in Fig. 3 the accuracy is varied according to data set size and number of parties but accuracy range is still accepted and as long as the number of parties increases the accuracy gets better.

3.5.3. Efficiency of proposed algorithm

Raising efficiency of the algorithm is mainly shown the decreases in time complexity. Proposed-Bayesian classifier algorithm reduces the time complexity mainly in two aspects.

- **First**, global V_{NBi}probabilities are quickly generated, since the Bayesian classifier algorithm executed locally for every party P_i, this enables solutions where the communication cost is independent of the size of the database and greatly cut down communication costs comparing with centralized data mining which needs to transfer all data into data warehouse to perform data mining algorithm.
- **Second,** the party P_i only has to encrypt encryption parameter T_i of homomorphic encryption system and class label y_i of V_{NBi} with public key e_i of RSA. So, the algorithm avoids numerous exponent operations and improves the speed of operation greatly.

3.5.4. The complexity analysis of the protocol

a- **The communication cost**

Let us use α to denote the number of bits of each ciphertext and k is the total number of parties. The total communication cost is the cost of $3\alpha.k$ from step 6 in the proposed algorithm.

b- **The computational cost is affected by:**

- The generation of k random numbers and k cryptographic key pair.
- The total number of $3k$ encryptions and $3k$ decryptions.

- Complexity for local naïve Bayes algorithm is O(*lqr*), where *l* is the number of features/attributes, *q* is values for each feature, and *r* is alternative values for the class.
- Additional computations as k^2 additions, *k(k-1)* subtractions and *k log(k)* sorting *k* numbers.

Therefore, the complexity of naïve Bayes classifier for k parties is dominant for not only the other computational costs but also for communication costs too. Consequently, the overall complexity of the proposed model = O(klqr).

4. Experimental results and discussion

Three different real-world data sets have been used from UCI machine learning repositories [32] which are Adult, Breast Cancer and Heart Spect. Bayesian classifier algorithm is calculated for each party and the mapping table is prepared for the transferred attributes. This is implemented in C# standard Edition 2010 and made to run on Intel® Core2 Duo, 2.0 GHz, 3 GB RAM system. The accuracy of the classifier algorithm for the three different data sets is shown in Fig. 3.

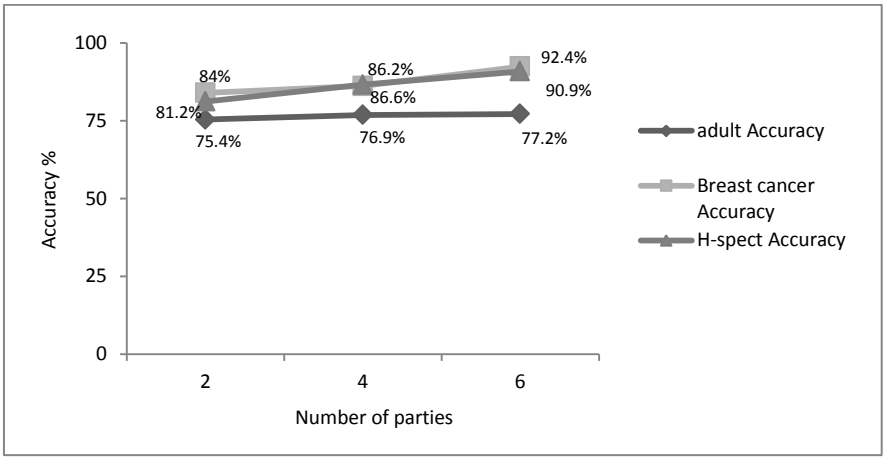

Fig. 3. The accuracy of distributed Naïve Bayes Classifier for three data sets

Table 2.
Adult data set

	Naïve Bayes Classifier			k-Nearest Neighbor Classifier		
No. of Parties	2	4	6	2	4	6
Accuracy % of distributed data	75.4	76.9	77.2	79.5	80.20	81.83
Records size(training set)	2000	4000	6000	2000	4000	6000
Number of attributes	13					
Accuracy % of centralized data	83.88			79.65		

Table 3.
Breast Cancer data set

	Naïve Bayes Classifier			k-Nearest Neighbor Classifier		
No. of Parties	2	4	6	2	4	6
Accuracy % of distributed data	84	86.2	92.4	93.0	93.9	94.5
Records size(training set)	200	400	600	200	400	600
Number of attributes	10					
Accuracy % of centralized data	93.2			93.7		

Table 4.
Spect Heart data set

	Naïve Bayes Classifier			k-Nearest Neighbor Classifier		
No. of Parties	2	4	6	2	4	6
Accuracy % of distributed data	81.2	86.6	90.9	73.6	84.2	89.4
Record size(training set)	40	80	120	40	80	120
Number of attributes	44					
Accuracy % of centralized data	90.4			91.2		

4.1. Comparison with other competitive algorithms

- Privacy preserving of sensitive data in proposed algorithm is based on the integration of RSA public key cryptosystem and homomorphic encryption scheme. But previous algorithms are based on secure sum technique or RSA encryption technique only.
- Naïve Bayes classifier is performed locally for each party in proposed algorithm. For previous algorithms partial computations from each party are performed to achieve global computations of Naïve Bayes classifier algorithm.
- In previous algorithms communication cost is dependent on the size of the database. But for proposed algorithm, communication cost is independent of the size of the database so no network overhead may happen due to large size of data sets.
- Collusion of parties is prohibited in proposed algorithm since no sharing of sensitive data except the final result of each party. In previous algorithms collusion may happen between parties due to exchange of partial computations.

- Computations of accuracy for previous algorithms in distributed databases are done as if it were the computations of centralized database, so the accuracy of distributed databases is comparable to centralized one. In proposed algorithm the experimental results show that the accuracy is as good as the centralized one**.**
- The accuracy of the Naïve Bayes Classifier and k-Nearest Neighbor Classifier for the same data sets is comparable as in Tables 2,3, and 4.

5. Conclusions

In this paper, a model of privacy-preserving distributed Bayesian classifier. The proposed algorithm uses semi-honest adversary model. The privacy preserving of the proposed algorithm is based on the technology of homomorphic and RSA encryption. Privacy preserving is achieved by performing the data mining algorithm locally. The result of each party is transferred in secure manner to the master client where to be processed to predict the class label of query instance in a way that network communication cost and performance are optimized. There are no partial computations of the algorithm of local parties that are transferred but the final result of each local party only. Experimental results show that it has good capability of privacy preserving, accuracy and efficiency. The collusion between parties is not permitted rather than the case that based on secure sum.

For future work, it is planned to extend the research to malicious model and test the scalability of the system and apply the methodology to another data mining task.

6. References

[1] R.O. Duda, P.E. Hart, D.G. Stork, "Pattern Classification", 2nd ed., New York: Wiley, (2001).

[2] Nitin Bhatia and Vandana, "Survey of Nearest Neighbor Techniques", in IJCSIS International Journal of Computer Science and Information Security, Vol. 8, No. 2, (2010)

[3] T. Cover and P. Hart. " Nearest neighbor pattern classification ". In IEEE Transaction of Information Theory, Vol. 13, pp. 21-27, (January 1968).

[4] Quinlan, J. R. "Induction of Decision Trees". Mach. Learn. 1, 1 (Mar. 1986), 81-106.

[5] Quinlan, J. R. "C4.5: Programs for Machine Learning". Morgan Kaufmann Publishers, (1993).

[6] George H. John and Pat Langley "Estimating Continuous Distributions in Bayesian Classifiers". Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, pp. 338 - 345, (1995).

[7] Jaideep Vaidya, Murat Kantarcioglu and Chris Clifton. " Privacy Preserving Naive Bayes Classification ". The VLDB Journal, VLDB Endowment, 17(4):879-898, (2008)

[8] S. Evfimievski. "Randomization techniques for privacy- preserving association rule mining". SIGKDD Explorations, 4(2), (Dec. 2002).

[9] Yehuda Lindell and Benny Pinkas. "Privacy Preserving Data Mining". In Proceedings of the 20th Annual International Cryptology Conference (CRYPTO), pages 36–54, Santa Barbara, CA, USA, (2000).

[10] Jaideep Vaidya and Chris Clifton. "Privacy-Preserving Decision Trees over Vertically Partitioned Data". In Proceedings of the 19th Annual IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec), pages 139–152, Storrs, CT, USA, (2005).

[11] Ming-Jun Xiao, Liu-Sheng Huang, Yong-Long Luo, and Hong Shen. "Privacy Preserving ID3 Algorithm over Horizontally Partitioned Data". In Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), pages 239–243, Dalian, China, (2005).

[12] Geetha Jagannathan and Rebecca N. Wright. "Privacy-Preserving Distributed k-Means Clustering over Arbitrarily Partitioned Data". In Proceeding of the eleventh ACM SIGKDD International conference on Knowledge discovery in data Mining (KDD), pages 593–599, Chicago, IL, USA, (2005).

[13] Somesh Jha, Louis Kruger, and Patrick Mc Daniel. " Privacy Preserving Clustering ". In Proceedings of the 10th European Symposium On Research In Computer Security (ESORICS), Pages 397–417, Milan, Italy, (2005).

[14] Jaideep Vaidya and Chris Clifton."Privacy-Preserving k-Means Clustering over Vertically Partitioned Data". In Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining (KDD), pages 206–215, Washington, DC, USA, (2003).

[15] C. Clifton, M. Kantarcioglu, and J. Vaidya. " Defining Privacy for Data Mining ". In Proceedings of the National Science Foundation Work shop on Next Generation Data Mining (NGDM), pages 126–133, Baltimore, MD, USA, (2002).

[16] Murat Kantarcioglu and Chris Clifton. " Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data". IEEE Transactions on Knowledge and Data Engineering, 16 (9) : 1026–1037, (2004).

[17] M. Barni, C. Orlandi, and A. Piva. " A Privacy-Preserving Protocol for Neural-Network-Based Computation". In Proceeding of the 8th Workshop on Multi media and Security, pages 146–151, Geneva, Switzerland, (2006).

[18] SaeedSamet and Ali Miri. " Privacy-Preserving Protocols for Perception Learning Algorithm in Neural Networks". In Proceeding of the 4th IEEE International Conference on Intelligent Systems (IS), pages 10–65–10–70, Varna, Bulgaria, (2008).

[19] Jimmy Secretan, Michael Georgiopoulos, and Jose Castro. "A Privacy Preserving Probabilistic Neural Network for Horizontally Partitioned Databases". In Proceedings of the International Joint Conference on Neural Networks (IJCNN), pages 1554–1559, Orlando, FL, USA, (2007).

[20] Zhiqiang Yang and Rebecca N. Wright. " Privacy-Preserving Computation of Bayesian Networks on Vertically Partitioned Data". IEEE Transactions on Knowledge and Data Engineering, 18 (9): 1253–1264, (2006).

[21] Saeed Samet, Ali Miri, and Eric Granger. "Incremental Learning of Privacy-Preserving Bayesian Networks", Applied Soft Computing Journal,(2013).

[22] T. Mitchell. " Machine Learning". McGraw-Hill Science/Engineering/Math, 1st edition, (1997).

[23] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes", Advances in Cryptography - EUROCRYPT '99, pp 223-238, Prague, Czech Republic, (1999).

[24] S. Golwasser and S. Micali, " Probabilistic encryption," Journal of Computer and System Sciences, vol. 28, pp. 270–299, (1984).

[25] O. Goldreich. " Foundations of cryptography ". Class notes, Technion University, (Spring 1989).

[26] Yehuda Linedell and Benny Pinkas, "Secure Multiparty Computation for Privacy-Preserving Data Mining", The Journal of Privacy and Confidentiality , Number 1, pp. 59-98, (2009).

[27] O. Goldreich. "The Foundations of Cryptography", volume 2, Cambridge University Press, (2004).

[28] R. Rivest, A. Shamir and L. Adleman. " A Method for Obtaining Digital Signatures and Public-Key Cryptosystems ". Communications of the ACM, 21 (2), pp. 120-126, (February 1978).

[29] R. Rivest, L. Adleman, and M. Dertouzos. "On data banks And privacy homomorphisms". In Foundations of Secure Computation, eds. R.A. De Millo etal., Academic Press, pp. 169-179,(1978).
[30] Gui Qiong and Cheng Xiao-hui "A privacy-preserving distributed method for mining association rules", (2009) International conference on Artificial intelligence and computational intelligence.
[31] O. Goldreich. " Secure multi-party computation", (Sept.1998). (working draft).
[32] Ronny Kohavi and Barry Becker "UCI Repository of Machine Learning Databases", Available at http://archive.ics.uci.edu/ml/datasets.html , Data Mining and Visualization , Silicon Graphics, (1996)

استخدام أساليب التنقيب عن البيانات مع المحافظة على خصوصية البيانات الموزعة أثناء عمل خوارزمي

الملخص العربى

المحافظة على خصوصية البيانات الموزعة أثناء استخدام طرق التنقيب عن البيانات يمكننا من اكتشاف أنماط وقواعد صحيحة دون الوصول مباشرة إلى البيانات الخاصة. في هذا البحث نقترح خوارزمية جديدة للحفاظ على الخصوصية أثناء التنقيب عن البيانات. وتستند الخوارزمية الجديدة المقترحة على إدماج نظام التشفير بالمفتاح العمومي RSA ونظام التشفير المتماثل الشكل حيث أن مشاركة البيانات بين الأطراف الموزعة لا تتم عدا النتيجة النهائية. يتم تنفيذ خوارزمية التنقيب عن البيانات محليا بالنسبة لكل طرف ثم تتم مقارنة النتيجة النهائية من جميع الأطراف للحصول على القيمة المستهدفة. ويستند الحل السابق للحفاظ على الخصوصية على طرق تسمح بالتواطؤ بين الأطراف وهذا غير مسموح في الحل المقترح. كما ان التحليل النظري والنتائج التجريبية تبين أن الخوارزمية المقترحة يمكن أن توفر قدرة جيدة من الحفاظ على الخصوصية والدقة والكفاءة.