HEMIN: A Cryptographic Approach for private k-NN classification

Pradeep Kumar, and Meena Singh, and Ashutosh Saxena Member, IEEE

Abstract—Data mining is frequently obstructed by privacy concerns. In many cases, data is shared with third party for the mining purpose. However, sharing of data for analysis is not possible due to prevailing privacy laws and/or policies. Privacy preserving data mining techniques have been developed to address this issue by providing mechanisms to mine the data while guaranteeing certain levels of privacy. In this paper, we address the issue of privacy preserving nearest neighbor search, which forms the basis of many data mining applications. We propose a scheme HEMIN for sharing the data to be mined by way of homomorphic encryption scheme at the data owner end. The nearest neighbors of a given data point is computed while the data point as well as the data points in training set are in encrypted form. This is being done using the third party which is assumed to be semi honest party. The efficiency of the model has been shown over three datasets namely, Iris, German credit database and Australian credit dataset. The misclassification ratio has been computed to show the efficiency of the k-NN classifier. The results show the effectiveness of the proposed method and suggests its applicability for various parametric data mining algorithms.

Keywords: Privacy Preserving Data Mining, k-NN classification, Cryptography, Homomorphic encryption schemes.

I. INTRODUCTION

Today’s digital era is experiencing exponential growth in the data collected as computer technology, network connectivity and disk storage space become increasingly affordable. The escalating increase in digital data have attracted the concerns about privacy of personal information globally [8]. These collected data contains various types of person-specific information. Such data may range from various domains including shopping habits, criminal records, medical history, credit records to name a few. Data holders, operating autonomously and with limited knowledge, are left with the difficulty of releasing information that does not compromise privacy and confidentiality. In many cases the survival of the database itself depends on the data holder’s ability to produce anonymous data because not releasing such information at all may diminish the need for the data, while on the other hand, failing to provide proper protection within a release may create circumstances that harm the public or others.

Data mining, with its promise to efficiently discover valuable, non-obvious information from large databases, is particularly vulnerable to misuse [14]. On the one hand, such data is an important asset to business organizations and governments both to decision making processes and to provide social benefits, such as medical research, crime reduction, national security, etc [9]. Analyzing such data opens new threats to privacy and autonomy of the individual if not done properly.

The threat to privacy becomes real since data mining techniques are able to derive highly sensitive knowledge from unclassified data that is not even known to database holders. Worse is the privacy invasion occasioned by secondary usage of data when individuals are unaware of behind the scenes use of data mining techniques [10]. Culnan [6] made a particular study of secondary information use which she defined as the use of personal information for other purposes subsequent to the original transaction between an individual and an organization when the information was collected. The key finding of this study was that concern over secondary use was correlated with the level of control the individual has over the secondary use. As a result, individuals are increasingly feeling that they are losing control over their own personal information that may reside on thousands of file servers largely beyond the control of existing privacy laws. This scenario has led to privacy invasion on a scale never before possible.

Rakesh and Agarwal were the first to raise the issue of privacy preserving in classification [2]. Private classification like ordinary classification comprises of two subtasks: learning a classifier from data with class labels often called a training data and predicting the class labels for unlabeled data using the learned classifier. Thus, the objective of privacy-preserving data classification is to build accurate classifiers without disclosing private information in the data being mined. The performance of privacy-preserving classification techniques should be analyzed and compared in terms of both the privacy protection of individual data and the predictive accuracy of the constructed classifiers. Here, the main emphasis is on privacy, i.e., how to disclose only the minimal amount of data.

The challenging problem that we address in this paper is: how can we protect against the abuse of the knowledge discovered from secondary usage of data and meet the business needs of an organizations to make a decision. The common practice is for organizations to release and receive person specific data with all explicit identifiers, such as name, address and telephone number, removed on the assumption that anonymity is maintained because the resulting data look anonymous. However, in most of these cases, the remaining data can be used to re-identify individuals by linking or matching the data to other data or by looking at unique
characters found in the released data. The solution for such a problem consists of two techniques, data anonymity i.e., to remove identifiers (e.g., names, social security numbers, addresses, etc.) followed by data transformation to protect some sensitive attributes (e.g., salary, age, etc.) since the released data, after removing identifiers, may contain other information that can be linked with other datasets to re-identify individuals or entities [15]. In this paper, we focus on the latter technique. Specifically, we consider the case in which confidential numerical attributes are encrypted in order to meet privacy protection in classification task, notably on $k$-NN Classification. In this paper, we present a $k$-NN classification technique in private environment. In the proposed technique, both the training dataset as well as unclassified dataset are encrypted using homomorphic encryption technique. The $k$-NN classification algorithm is modified in order to classify the unclassified dataset in encrypted form itself using the concept of semi honest party, which acts here as a third party.

The rest of the paper is organized as follows. In Section II, we present the related work in the area of privacy preserving data mining. Section III describes in brief about the homomorphic encryption scheme. Section IV, presents the $k$-NN classification algorithm in private environment. Section V, presents the experimental results and discussion and finally we conclude in section VI.

II. RELATED WORK

Much work has been done in the field of Privacy Preserving Data Mining (PPDM). There are two main approaches to this field: the randomization approach, and the cryptographic approach. The randomization method was initially proposed by Agrawal and Srikant [2], with their work on reconstructing approximations of distribution of the original dataset from randomly perturbed values. Here the database values are perturbed by adding noise. During the mining process the noise values will be balanced and the result of the process will be the original database values. The advantage of this approach is that it will provide privacy but the utility of the data is being messed up because of the noise values. The randomization approach has also been applied to association rule mining [1].

The cryptographic approach primarily makes use of Secure Multiparty Computation ideas from the field of Cryptography. A number of cryptography-based approaches have also been developed in the context of Privacy Preserving Data Mining algorithms to solve secure multiparty computation [18], [16], [17]. In secure multi party computation, two or more parties want to jointly compute the function from the combination of their inputs. The issue here is how to conduct such a computation while preserving the privacy of the inputs. Secure computation was first introduced by Yao [18].

A cryptography-based approach is described in Vaidya and Clifton [16]. Such approach addresses the problem of association rule mining in vertically partitioned data. In other words, its aim is to determine the item frequency when transactions are split across different sites, without revealing the contents of individual transactions. In particular, the security of the protocol for computing the scalar product is analyzed. The total communication cost depends on the number of candidate item sets and can best be expressed as a constant multiple of the I/O cost of the apriori algorithm.

Several schemes for designing various operations in PPDM have been proposed in [5], [4]. The authors have suggested that the Secure Multiparty Computation schemes are useful primitives in various PPDM operations. Similarly in [7], the authors suggested that one of the applications of the secure multiparty computations is the PPDM operations. These schemes guarantee privacy and they are scalable i.e. they can involve many parties into the operations. The multiparty computation schemes offer privacy and utility to the database, however they suffer from the drawback of large communication complexity values. Especially when we are handling databases, it is a costly affair.

Yet another scheme is suggested in [5], [4], which is based on Homomorphic Encryption Schemes. These schemes involve encryption and decryption process so they offer privacy and utility to the database values. Basically, the homomorphic encryption schemes are either Additive i.e. they support addition operation or Multiplicative i.e. they support multiplication operation. There are no homomorphic schemes that support both the operations.

Many parametric data mining operations can be expressed in terms of addition, multiplication, subtraction or division. $k$-NN classification algorithm, k-Means clustering algorithm falls under parametric class. Hence we need a scheme which supports all the operations needed to perform data mining tasks. In [3], a scheme has been suggested which support both the algebraic operations. But the scheme has been restricted only to the formulae which are in $2DNF$ form. However in practice, it is relatively difficult to reduce all the PPDM operations into $2DNF$ form. In our proposed approach we have come up with a scheme which includes both addition and multiplicative homomorphic encryption scheme. In our approach we encrypted the data with additive homomorphic encryption scheme. Hence the addition and subtraction operations can be supported under this scheme. When a multiplication operation is called the corresponding ciphertext is decrypted and again encrypted using a multiplicative homomorphic scheme. Then the corresponding multiplication and division operation is applied. Finally the result is decrypted and again encrypted using additive homomorphic scheme. In the scheme, we have used the paillier scheme for additive homomorphic scheme and RSA for multiplicative homomorphic scheme.

III. HOMOMORPHIC ENCRYPTION SCHEMES

The Homomorphic Encryption is a cryptographic scheme, which computes the sum or the product of plaintexts from their corresponding encrypted text without decrypting them i.e. $E(m_1 \# m_2) = E(m_1) || E(m_2)$ where $m_1, m_2$ are messages and $\#$, $||$ (may be same as $\oplus$ or different) represents some operation on the group containing $m_1$ and
We say encryption scheme $E$ is $\#$ homomorphic. The Homomorphic encryption systems can be very useful because they allow a third party to get the encrypted text of the result of operation $E(m_1 \# m_2)$ without knowledge of the corresponding plain text. Thus, it can provide a setting for operations on encrypted values by someone else such that only the party or entity who knows the key can decrypt the result. If $\#$ is addition then we call the homomorphism as Additive Homomorphism and if the symbol represents multiplication then we call the scheme as Multiplicative Homomorphism. If a scheme is both multiplicative and additive then we call such scheme as Algebraic Homomorphism. To the best of our knowledge there are no schemes suggested in the literature for algebraic homomorphism till now, except a scheme by Boneh [3] for evaluating the formula in 2-DNF. However, this scheme is not flexible enough for applying to all data mining algorithms. Hence designing an algebraic homomorphic scheme remains an open problem. The homomorphic encryption schemes are used in Privacy Preserving Database Operations [11] like select, project and aggregate operations. The RSA and ElGamal encryption schemes are examples of Multiplicative Homomorphic Encryption schemes. The RSA is multiplicative homomorphic because:

$$E(x) = x^e \mod n.$$  
$$E(y) = y^e \mod n.$$  

Now

$$E(x \cdot y) = (x \cdot y)^e \mod n$$  
$$= (x^e \cdot y^e) \mod n$$  
$$= E(x) \cdot E(y).$$  

i.e. $E(x \cdot y) = E(x) \ast E(y)$.

Similarly Goldwasser Micalli scheme, Benolah scheme and Paillier scheme are additive homomorphic encryption schemes. The Paillier scheme is additive homomorphic because:

$$E(x) \ast E(y) = (g^{xy}r^{n \cdot y}) \ast (g^{x \cdot y}r^{n \cdot y})$$  
$$= g^{x \cdot y}r^{n \cdot y} \cdot (g^{x \cdot y})r^{n \cdot y}$$  
$$= E(x \cdot y)$$  

i.e. $E(x + y) = E(x) + E(y)$

Since the above homomorphic schemes will not support both addition and multiplication operations. We propose to use Paillier scheme [12] to support addition and subtraction operations and RSA scheme for multiplication and division operations. In an elaborate fashion the data is initially encrypted using RSA scheme [13], whenever the addition operation is called the data is decrypted and subsequently encrypted using paillier scheme. After the operation is performed the data is again encrypted using RSA scheme. On the similar lines the subtraction operation is performed.

Usually the size of $n$, $|n| = 512, \ldots, 2048$ bits. The base $g$ can be randomly among elements of order divisible by $n$. Taking $g$ as small value like 2 will allow an immediate speed up factor, provided the chosen value fulfills the requirement, $g \in B$ imposed by the setting. Optionally, $g$ could even be fixed to a constant value if the key generation process includes a specific adjustment.

IV. PRIVACY PRESERVING KNN CLASSIFICATION

KNN classification is highly applicable algorithm in various domains. The basic problem of k-nearest neighbor search is as follows. Given a training dataset $S$, and a particular point $x \in S$, find the set of points $N_k(x) \subseteq S$ of size $k$, such that for every point $n \in N_k(x)$ and for every point $y \in S$, $y \ni N_k(x) \Rightarrow d(x,y) \leq d(x,n), \text{where } d(x,y)$ represents the distance between the points $x$ and $y$.

Whenever a organization wish to go for a data mining task, it has two options. Firstly, the organization can open a new analytic division or secondly the organization can out source his data to the analytic company. In both the cases, organization has to trust the other party. An organization assumes that the people engaged in analytic or the second party to whom the data is out sourced are semi honest parties. Hence we can say that in Privacy Preserving Data Mining(PPDM) operation, there are two entities involved, namely, Data Owner and Data Miner. The owner will mask or perturb/encrypt the data in order to keep the data secure. The owner of the database will perturb/encrypt the data such that the sensitive information is hidden. Consequently, a data miner has to operate on the perturbted/encrypted data to find the hidden knowledge from it.

KNN classification problem can be divided into the following two sub-problems.

- **Nearest neighbor selection:** Given a query instance $x$ to be classified, the databases need to identify all points that are among the k nearest neighbors of $x$.
- **Classification:** Classify $x$ to the class of instance with majority value.

In this paper, we present a k-NN classification technique where the data holder gives the data miner after encrypting the database. Miner works over the encrypted data and classifies the new unseen record. Consider a training dataset which has a set of points $X = x_1, \ldots, x_n$, with their class labels. The dataset are encrypted using homomorphic encryption scheme. The data holder also provides the encrypted data points to the miner whose class labels are not known. What we intend to find is the set of points $\text{Nearest neighbor set} \subset X$ of size $k$, such that all points in $\text{Nearest neighbor set}$ are closer to $x$ than all points within $X - \text{Nearest neighbor set}$

The data holder encrypts the database using RSA and Paillier functions. Many RSA encrypted values do not support homomorphic addition hence data points are converted to their corresponding RSA values using RSAToPaillier() function. As the encrypted values cannot be mined we introduce a concept of a new entity called Third Party. The third party is an entity belonging to the owner and it is installed at the miner. In short, we can state that the third party is kind of a proxy of the data owner. Now the miner will
send all the operations that involve the decryption of the data and correction of the input values like, decrypt, encrypt, PaillierToRSA, RSAToPaillier, correct, nComplimentInRSA) to the third party. The result of the operation in encrypted form will be given back to the miner. Essentially, the third party has the following advantages, firstly we can ensure data privacy i.e. the miner has no means to retrieve plaintext from the encrypted text. Secondly, if there was no third party then the miner has to interact with the owner for each operation which would be an overhead especially when we are handling databases.

The security of our scheme is based on the RSA and Paillier cryptosystem. The paillier scheme is based on the composite degree residuosity assumption which is an intractable assumption. According to the results in [12], in a random oracle model it is not possible to guess the plaintext from the corresponding ciphertext with non-negligible success probability. Hence we can conclude that the paillier scheme is secure. In the RSA scheme[13], the Factorisation problem is the intractable assumption used in the scheme. This problem is NP-Complete, hence there is no polynomial time algorithm to solve this problem. Assume that in the random oracle model, there is a polynomial time algorithm \( A() \), that could guess the plaintext represented by the corresponding ciphertext. If that is the case, then we can say that the algorithm has solved the factoring problem in polynomial time which is a contradiction. So \( A() \) cannot exist and we can conclude that RSA scheme is secure. Finally, based on the proof of security of the above encryption schemes we can conclude that our scheme is secure.

V. EXPERIMENTAL RESULTS

This section briefs about the experiments performed for privacy preserving data mining using the homographic encryption scheme. The proposed scheme is implemented in Java 1.4. Experiments were conducted over three datasets. Three datasets namely, Iris, Australian Credit and German Australia credit datasets are obtained from UCI Machine Learning Repository. The data sets are illustrated in Table I, the Australian, German credit data sets and iris, are available from the UCI Repository of Machine Learning Databases and are adopted herein to evaluate the predictive accuracy. The Australian credit data consists of 307 instances of creditworthy applicants and 383 instances where credit is not creditworthy. Each instance contains 6 nominal, 8 numeric attributes, and 1 class attribute (accepted or rejected). This dataset is interesting because there is a good mixture of attributes: continuous, nominal with small numbers of values, and nominal with larger numbers of values. There are also a few missing values.

To protect the confidentiality of data, the attributes names and values have been changed to meaningless symbolic data. The German credit scoring data are more unbalanced, and it consists of 700 instances of creditworthy applicants and 300 instances where credit should not be extended. For each applicant, 24 input variables describe the credit history, account balances, loan purpose, loan amount, employment status, personal information, age, housing, and job title. This data set only consists of numeric attributes. Iris dataset consists of 150 instances with 4 numerical attributes. These 150 instances belong to three classes namely, Iris setosa, Iris versicolor and Iris virginica. This dataset is interesting because iris flower of class 2 and class 3 are not linearly separable. Only the iris flower samples of class 1 is separable from class 2 and class 3.

As the encryption algorithm can work over integer values in our experiments we have taken only those records which are integer and the nominal values records were dropped from the experiments. To compute the distance between two data points Manhattan distance is used. Training set and test-

---

**TABLE I**

<table>
<thead>
<tr>
<th>Number</th>
<th>Names</th>
<th>Number of Classes</th>
<th>Number of Records</th>
<th>Nominal Features</th>
<th>Numerical features</th>
<th>Total Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>German</td>
<td>2</td>
<td>1000</td>
<td>0</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>2</td>
<td>Australian</td>
<td>2</td>
<td>690</td>
<td>6</td>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>Iris</td>
<td>3</td>
<td>150</td>
<td>0</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

**TABLE II**

**EXPERIMENTAL RESULTS FOR IRIS DATASET**

<table>
<thead>
<tr>
<th>k=5</th>
<th>Iris Setosa</th>
<th>Iris Virginica</th>
<th>Iris Versicolor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris Setosa</td>
<td>15</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Iris Virginica</td>
<td>0</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>Iris Versicolor</td>
<td>0</td>
<td>0</td>
<td>15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>k=7</th>
<th>Iris Setosa</th>
<th>Iris Virginica</th>
<th>Iris Versicolor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris Setosa</td>
<td>15</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Iris Virginica</td>
<td>0</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>Iris Versicolor</td>
<td>0</td>
<td>0</td>
<td>15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>k=9</th>
<th>Iris Setosa</th>
<th>Iris Virginica</th>
<th>Iris Versicolor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris Setosa</td>
<td>15</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Iris Virginica</td>
<td>0</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>Iris Versicolor</td>
<td>0</td>
<td>0</td>
<td>15</td>
</tr>
</tbody>
</table>
In this paper, we propose a new scheme on privacy-preserving data classification. The classification task has been done using k-NN classification algorithm which forms the core of many data mining applications. In the current approach we designed a model to encrypt the database using homomorphic encryption scheme. Miner with the help of third party mines the encrypted database. It uses a two-way communication mechanism between the data miner and the third party with little overhead. As a result, our scheme has the benefit of a better tradeoff between accuracy and privacy.

We showed the viability of our proposed scheme over k-NN classification algorithm using three real time datasets. Our work is preliminary and many extensions can be made. In the current work we have only used the integer attributes. We are currently investigating how to apply our scheme to real attributes. Also the extension of the work includes looking for a scheme to adopt it for clustering problem. We would also like to investigate the integration of our scheme with different randomization techniques for the scalability issues.

VI. Conclusion

In this paper, we propose a new scheme on privacy-preserving data classification. The classification task has been done using k-NN classification algorithm which forms the core of many data mining applications. In the current approach we designed a model to encrypt the database using homomorphic encryption scheme. Miner with the help of third party mines the encrypted database. It uses a two-way communication mechanism between the data miner and the third party with little overhead. As a result, our scheme has the benefit of a better tradeoff between accuracy and privacy.

We showed the viability of our proposed scheme over k-NN classification algorithm using three real time datasets. Our work is preliminary and many extensions can be made. In the current work we have only used the integer attributes. We are currently investigating how to apply our scheme to real attributes. Also the extension of the work includes looking for a scheme to adopt it for clustering problem. We would also like to investigate the integration of our scheme with different randomization techniques for the scalability issues.

TABLE III

AUSTRALIAN CREDIT DATASET

<table>
<thead>
<tr>
<th>k=5</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>109</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
<td>82</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>k=7</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>109</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>81</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>k=9</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>109</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>13</td>
<td>78</td>
</tr>
</tbody>
</table>

TABLE IV

GERMAN CREDIT DATASET

<table>
<thead>
<tr>
<th>k=5</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>207</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>93</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>k=7</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>207</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>93</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>k=9</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>207</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>93</td>
</tr>
</tbody>
</table>

REFERENCES


