International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Adjacent Contiguous Categorization over Semantically Secured Records

C.Subathra[1], M.S.VijayKumar[2]
[1]*PG scholar,* [2]*Assistant Professor,*
[1,2]*Department of CSE, Tejaa Shakthi Institute of Technology for Women, Coimbatore*

Abstract- Data Mining has large applications in many areas such as banking, medicine, scientific research and among government agencies. But due to the rise of many security issues, many theoretical and practical solutions to the classification problem have been proposed under different security models. However, users now have the opportunity to outsource their data, in encrypted form, to the cloud. Since the data on the cloud is in encrypted form, existing privacy-preserving classification techniques are not applicable. This model focus on solving the classification problem over encrypted data. In particular, propose a secure k-NN classifier over encrypted data in the cloud. The proposed protocol protects the confidentiality of data, privacy of user's input query, and hides the data access patterns. The secure k-NN classifier over encrypted data is developed under the semi- honest model.

Keywords - Security, K-NN classifier, Outsourced databases, Encryption

I. INTRODUCTION

The cloud computing paradigm [2] is revolutionizing the organizations way of operating their data particularly in the way they store, access and process data. As an emerging computing paradigm, cloud computing attracts many organizations to consider seriously regarding cloud potential in terms of its cost-efficiency, flexibility, and offload of administrative overhead. Most often, organizations delegate their computational operations in addition to their data to the cloud. Even though cloud has many advantages, privacy and security issues in the cloud are preventing companies to utilize those advantages. When data are highly sensitive, the data need to be encrypted before outsourcing to the cloud. However, when data are encrypted, irrespective of the underlying encryption scheme, performing any data mining tasks becomes very challenging without ever decrypting the data. Moreover, cloud can also derive useful and sensitive information about the actual data items by observing the data access patterns even if the data are encrypted .Therefore, the security requirements of the DMED problem on a cloud are threefold confidentiality of the encrypted data, confidentiality of a user's query record, and hiding data access patterns.Existing work on privacy-preserving data mining either perturbation or secure multi-party computation cannot solve the DMED problem. Perturbed data do not possess semantic security, so data perturbation techniques cannot be used to encrypt highly sensitive data. Also the perturbed data do not produce very accurate data mining results. Secure multi-party computation based approach assumes data are distributed and not encrypted at each participating party. In addition, many intermediate computations are performed based on non-encrypted data. As a result, proposed a methods to effectively solve the DMED problem assuming that the encrypted data are outsourced to a cloud. This model concentrates on executing the k-nearest neighbor classification method over encrypted data in the cloud computing environment.

A. Problem Definition

Suppose AA owns database D of n records and m + 1 attributes. Let $t_{i,j}$ denote the jth attribute value of record t_i. Initially, AA encrypts database that is, computes $E_{pk}(t_{i,j})$, for $1 \leq i \leq n$ and $1 \leq j \leq m + 1$, where column (m + 1) contains the class labels. The encryption scheme is semantically secure [4]. Let the encrypted database be denoted by D′. Assume that AA outsources D′ as well as the future classification process to the cloud. AA becomes the data owner.

Let BB be an authorized user who wants to classify his input record $q = (q_1,\ldots,q_m)$ by applying the k-NN classification method based on D′. Refer to such a process as privacy-preserving k-NN (PPkNN) classification over encrypted data in the cloud. BB becomes the client. The PPkNN protocol is defined as:

$$PPkNN(D',q) \rightarrow c_q$$

where c_q denotes the class label for q after applying k-NN cla ssification method on D′ and q.

II. LITERATURE REVIEW

The Paillier cryptosystem is an additive homomorphic and probabilistic public-key encryption scheme whose security is based on the Decisional Composite Residuosity Assumption. Let E_{pk} be the encryption function with public key $_{pk}$ given by (N,g), where N is a product of two large primes of similar bit length and g is a generator in $Z^*_{N^2}$. Also, let D_{sk} be the decryption function with secret key $_{sk}$ [3]. For any given two plaintexts a, b € Z_N, the Paillier encryption has the following properties:

Homomorphic addition.

$$D_{sk}(E_{pk}(a + b)) = D_{sk}(E_{pk}(a) * E_{pk}(b) \bmod N^2).$$

Homomorphic multiplication

$$D_{sk}(E_{pk}(a * b)) = D_{sk}(E_{pk}(a)^b \bmod N^2).$$

Semantically security

The encryption scheme is semantically secure. Briefly, given a set of ciper-texts, an adversary cannot deduce any additional information about the plaintext(s).

These are sub-protocols that used in constructing proposed k-NN protocol. All of the below protocols are considered under two party semi-honest setting. In particular, assume the exist of two semi-honest parties P_1 and P_2 such that the Paillier's secret key sk is known only to P_2 whereas pk is treated as public [1].

A. Secure Multiplication (SM) Protocol: This protocol considers P_1 with input ($E_{pk}(a)$, $E_{pk}(b)$) and outputs $E_{pk}(a*b)$ to P_1, where a and b are not known to P_1 and P_2. During this process, no information regarding a and b is revealed to P_1 and P_2

B. Secure Squared Euclidean Distance (SSED) Protocol: In this protocol, P_1 with input ($E_{pk}(X)$, $E_{pk}(Y)$) and P_2 with sk securely compute the encryption of squared Euclidean distance between vectors X and Y. Here X and Y are m dimensional vectors where $E_{pk}(X)=(E_{pk}(x_1),..., E_{pk}(x_m))$ and $E_{pk}(Y) = (E_{pk}(y_1),...,E_{pk}(y_m))$.The output of the SSED protocol is $E_{pk}(|X-Y|^2)$ which is known only to P_1.

C. Secure Bit Decomposition (SBD) Protocol: P_1 with input $E_{pk}(z)$ and P_2 securely compute the encryptions of the individual bits of z, where $0 \le z < 2^l$. The output $[z]=(E_{pk}(z_1), E_{pk}(z_1))$ is known only to P_1. Here z_1 and z_l are the most and least significant bits of integer z, respectively.

D. Secure Minimum (SMIN) Protocol: In this protocol, P_1 holds private input (u′, v′) and P_2 holds sk, where $u' = ([u], E_{pk}(s_u))$ and $v'=([v], E_{pk}(s_v))$. Here s_u (resp., s_v) denotes the secret associated with u (resp., v). The goal of SMIN is for P_1 andP_2 to jointly compute the encryptions of the individual bits of minimum number between u and v. In addition, they compute $E_{pk}(s_{min(u,v)})$. That is, the output is ($[min(u, v)]$, $E_{pk}(s_{min(u,v)})$) which will be known only to P_1.

E. Secure Minimum out of n Numbers (SMINn) Protocol: In this protocol, consider P_1 with n encrypted vectors ($[d_1],...,[d_n]$) along with their respective encrypted secrets and P_2 with sk. Here $[d_i] = (E_{pk}(d_{i,1}),..., E_{pk}(d_{i,l}))$ where $d_{i,1}$ and $d_{i,l}$ are the most and least significant bits of integer d_i respectively, for $1 \le i \le n$. The secret of d_i is given by s_{di}. P_1 and P_2 jointly compute $[min(d_1,...,d_n)]$. In addition, compute $E_{pk}(s_{min(d1,...,dn)})$). At the end of this protocol, the output ($[min(d_1,.....,d_n)]$, $E_{pk}(s_{min(d1,...,dn)})$) is known only to P_1. During the $SMIN_n$ protocol, no information regarding any of d_i`s and their secrets is revealed to P_1 and P_2.

The ability of databases to organize and share data often raises privacy concerns. Data warehousing combined with data mining, bringing data from multiple sources under a single authority, increases the risk of privacy violations. Privacy preserving data mining provides a means of addressing this issue, particularly if data mining is done in a way that doesn't disclose information beyond the result [5]. The method for privately computing knn classification from distributed sources without revealing any information about the sources or their data, other than that revealed by the final classification result.The cloud has gained increasing popularity for its flexibility and scalability, which motivates cloud service providers to offer accesses to cloud databases, such as Amazon Relational Database Service (Amazon RDS), Google Cloud SQL, and Microsoft SQL Azure. Data owners outsource their databases to the cloud service providers and rely on their services for storage, management, and query processing of the databases. Clearly, this framework offers great flexibility and scalability to data owners and their clients, and it is especially useful for users with stringent local resources. However, the remote placement of the data brings security concerns [8]. A data owner may prefer to prevent the service provider from learning the content of database D or the contents of queries to D, while still requiring the server to provide database functionality for D in the cloud. For this purpose, the data owner needs to encrypt D with an encryption scheme E and only publishes to the server an encrypted version of D, denoted as E(D). The clients also need to encrypt their queries q and send only E(q) to the server. The server needs to identify the cipher text in E(D) that corresponds to the answer of q on D, using only E(q) and E(D).

III. MATERIALS AND METHOD

Let the encrypted database be denoted by D`. The data owner outsources D` as well as the future classification process to the cloud, attribute values and their Euclidean distances lie in $[0,2^l]$. In addition, let w denote the number of unique class labels in D. The existence of two non-colluding semi-honest cloud service providers, denoted by C_1 and C_2, which together form a federated cloud. Under this setting, data owner outsources encrypted database D` to C_1 and the secret key sk to C_2. Here it is possible for the data owner to replace C_2 with her private server. The main purpose of using C_2 can be motivated by the following two reasons.(i) With limited computing resource and technical expertise, it is in the best interest of data owner to completely outsource its data management and operation all tasks to a cloud.

(ii) Suppose receiver wants to keep his input query and access patterns private from data owner. In this case, if data owner uses a private server, then she has to perform computations assumed by C_2 under which every purpose of outsourcing the encrypted data to C_1 is negated. The goal of the PPkNN protocol is to classify users' query records using D` in a privacy preserving manner. Consider an authorized user who wants to classify his query record $q=(q_1,...,q_m)$ based on D' in C_1. The proposed PPkNN protocol mainly consists of the following two stages:

A. Secure Retrieval of k-Nearest Neighbors (SRkNN).

In this stage, receiver initially sends his query q to C_1. After this, C_1 and C_2 involve in a set of sub-protocols to securely retrieve the class labels corresponding to the k-nearest neighbor so the input query q. At the end of this step, encrypted class labels of k nearest neighbors are known only to C_1.

Algorithm PPkNN(D`,q) → c_q

Require C_1 has D` and л; C_2 has sk; Client has q

1) Client:
- a) Compute $E_{pk}(q_j)$, for $1 \leq j \leq m$
- b) Send $E_{pk}(q) = (E_{pk}(q_1,...,E_{pk}(q_m))$ to C_1

2) C_1 and C_2:
- a) C_1 receives $E_{pk}(q)$ from Client
- b) for i=1 to n do:

  $E_{pk}(d_i) \leftarrow SSED(E_{pk}(q),E_{pk}(t_i))$

  $[d_i] \leftarrow SBD(E_{pk}(d_i))$

3) for s = 1 to k do:
- a) C_1 and C_2:

  $([d_{min}]E_{pk}(I),E_{pk}(c`)) \leftarrow SMIN_n(k_1,\ldots,k_n)$,where

  $k_i=([d_i],E_{pk}(I_{ti}), E_{pk}(t_{i,m+1})$

  $E_{pk}(c`s) \leftarrow Epk(c`)$
- b) C_1:

  $\Delta \leftarrow E_{pk}(I)^{N-1}$

  for i=1 to n do:

  $\tau_i \leftarrow E_{pk}(i) * \Delta$

  $\tau`_i \tau_i^i$,where $r_i \in_R Z_N$

  $ß \leftarrow л(\tau`)$; send ß to C_2
- c) C_2:

  $ß`_i \leftarrow D_{sk}(ß_i)$, for $1 \leq i \leq n$

  Compute U`, for $1 \leq i \leq n$:

  if $ß`_i=0$, then $U`=E_{pk}(1)$

  otherwise,$U`_i=E_{pk}(0)$

Send U` to C_1
- d) C_1:$V \leftarrow л^{-1}(U`)$
- e) C_1 and C_2, for $1 \leq i \leq n$ and $1 \leq \gamma \leq l$:

$E_{pk}(d_{i;\gamma}) \leftarrow SBOR(V_i,E_{pk}(d_{i;\gamma}))$

4) $SCMC_k(E_{pk}(c`_1),\ldots,E_{pk}(c`_k))$

Volume 4 Issue III, March 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

B. Secure Computation of Majority Class ($SCMC_k$).

Following from stage1, C_1 and C_2 jointly compute the class label with a majority voting among the k-nearest neighbors of q. At the end of this step, only client knows the class label corresponding to his input query record q.

Algorithm $SCMC_K(E_{pk}(c`_1),\ldots,E_{pk}(c`_k)) \rightarrow c_q$

Require $(E_{pk}(c_1),\ldots, E_{pk}(c_w))$, $(E_{pk}(c`_1),\ldots, E_{pk}(c`_k))$ are known only to C_1 ; sk is known only to C_2.

1) C_1 and C_2:

a) $(E_{pk}(f(c_1)),\ldots, E_{pk}(f(c_w))) \leftarrow SF(v,v`)$,where

b) $v=(E_{pk}(c_1),\ldots, E_{pk}(c_w))$, $v`=(E_{pk}(c`_1),\ldots, E_{pk}(c`_k))$

c) for i = 1 to w do:
$[f(c_i)] \leftarrow SBD(E_{pk}(f(c_i)))$

d) $([f_{max}], E_{pk}(c_q)) \leftarrow SMAX_w(x_1 \ldots ;x_w)$, where
$x_i= [f(c_i),(E_{pk}(f(c_i)))$, for $1 \leq i \leq w$

2) C_1:

a) $.\gamma_q \leftarrow E_{pk}(c_q) * E_{pk}(r_q)$, where $r_q \in Z_N$

b) Send γ_q to C_2 and r_q to Client

3) C_2:

a) Receive γ_q from C_1

b) $\gamma`_q \leftarrow D_{sk}(\gamma_q)$; send γ_q to Client

4) Client:

a) Receive r_q from C_1 and γ_q from C_2

b) $c_q \leftarrow \gamma_q - r_q \mod N$

IV. RESULTS AND DISCUSSION

In fig.1 the data owner upload the file into the cloud. The file is stored in the cloud in encrypted form. Paillier cryptosystem is used for the encryption. Using private key file is encrypted. Data owner has the full rights on the file to update, delete and modify the data. Once the data are outsourced data owner does not involve in any other computation.

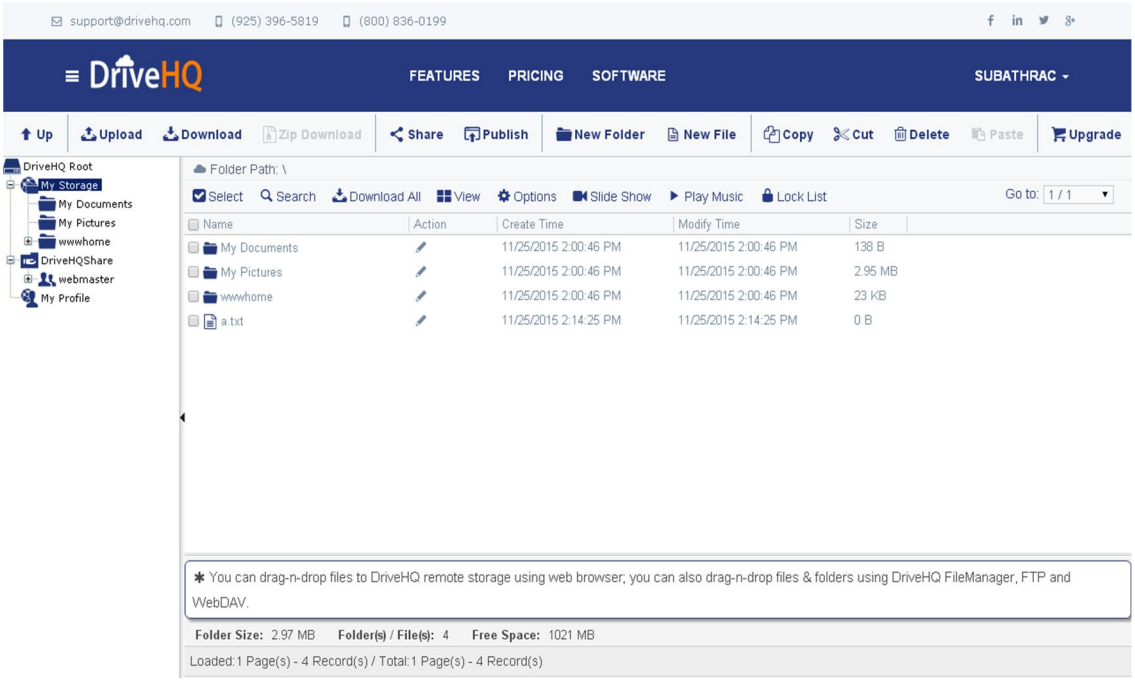

Fig. 1 File Uploaded

Volume 4 Issue III, March 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

The paillier cryptosystem is an additive homomorphic and probabilistic public key encryption system. The encryption scheme is semantically secured. The encrypted data are stored in cloud in semi honest cloud model. Two non-colluding semi honest cloud service providers, denoted by C_1 and C_2. The encrypted data are stored in the C_1 and secret key is stored in C_2.The client sends query to the cloud. Secure Minimum protocol is used, the goal of this protocol is to securely compute the encryption of the individual bit of data. At the end of the SMIN, the output is known only to C_1.The unauthorized user cannot view or change the data. Hence the data are secure in the cloud.C_1 and C_2 jointly compute the encryption of the individual bits, so no information regarding the contents is revealed to C_1 and C_2. With user's input query, c_1 with private input and c_2 with private key jointly involve in the SSED (Secure Squared Euclidean Distance) protocol and output is known only to C_1 . Then C_1 with SSED output and c_2 securely compute the encryption of the individual bits using SBD (Secure Bit-Decomposition) protocol and the output is known only to C_1. Finally C_1 and C_2 compute the encryption class based on the query and sends to the C_1. The data access pattern is hidden from the cloud. The client decrypt the file and receive the data by using private key.

In fig.2 the receiver login to the cloud in order to send query. The query is sent to the C_1. After that C_1 and C_2 jointly compute the data and then the result is sent to the C_1. Secure file is decrypted with the public key and file is downloaded to the client. The client can securely retrieve the data. The file name and key is sent to the client's mail.

Fig.2 Receiver Login

Fig.3 Data Download

In fig.3 the file can be downloaded using the key value and file name. The client can download the file using the key and file name which is sent to his mail. The client's query is confidentially and file can be viewed only by the requested client. The confidentiality of data, user input query and the data access pattern are achieved using the protocols.

V. CONCLUSION AND FUTURE WORK

To protect user privacy, various privacy-preserving classification techniques have been proposed over the past decade. The existing techniques are not applicable to outsourced database environments where the data resides in encrypted form on a third-party server. The proposed privacy-preserving k-NN classification protocol over encrypted data in the cloud, protects the confidentiality of the data, user's input query, and hides the data access patterns. The performance of the protocol is evaluated under different parameter settings. Since improving the efficiency of $SMIN_n$ is an important first step for improving the performance of PPkNN protocol, plan to investigate alternative and more efficient solutions to the $SMIN_n$ problem in future work.

REFERENCES

[1] K. Samanthula, Y. Elmehdwi, and W. Jiang, "knearest neighbour classification over semantically secure encrypted relational data," eprint arXiv: 1403. 5001, 2014.
[2] P. Melland T. Grance, "The NIST definition of cloud computing (draft)" NIST Special Publication, vol. 800, p. 145, 2011.
[3] P. Paillier, "Public key cryptosystems based on composite degree residuosity classes," in Proc. 17th Int. Conf. Theory Appl. Cryptographic Techn., 1999, pp. 223–238.
[4] R. J. Bayardo and R. Agrawal, "Data privacy through optimal k- anonymization," in Proc. IEEE 21st Int. Conf. Data Eng. ,2005, pp. 217–228.
[5] M. Kantarcioglu and C. Clifton, "Privately computing a distributed k-nn classifier," in Proc. 8th Eur.Conf. Principles Practice Knowl Discovery Databases, 2004, pp. 279-290.
[6] Y. Lindell and B. Pinkas, "Privacy preserving data mining," in Proc. 20th Annu. Int. Cryptol. Conf. Adv. Cryptol., 2000, pp. 36–54.
[7] L. Xiong, S. Chitti, and L. Liu, "K nearest neighbor classificationacross multiple private databases," in Proc. 15th ACM Int. Conf.Inform. Knowl. Manage., 2006, pp. 840–841.
[8] X. Xiao, F. Li, and B. Yao, "Secure nearest neighbor revisited," inProc. IEEE Int. Conf. Data Eng., 2013, pp. 733–744.
[9] A. C. Yao, "Protocols for secure computations," in Proc. 23rdAnnu. Symp. Found. Comput. Sci., 1982, pp. 160–164.
[10] C. Gentry, "Fully homomorphic encryption using ideal lattices," in Proc. 41st Annu. ACM Sympos. Theory Comput., 2009, pp. 169–178
[11] C. Gentry and S. Halevi, "Implementing gentry's fully-homomorphic encryption scheme," in Proc. 30th Annu. Int. Conf. TheoryAppl. Cryptographic Techn.: Adv. Cryptol., 2011, pp. 129–148.
[12] A. Shamir, "How to share a secret," Commun. ACM, vol. 22,pp. 612–613, 1979.
[13] D. Bogdanov, S. Laur, and J. Willemson, "Sharemind: A framework for fast privacy-preserving computations," in Proc. 13th Eur.Symp. Res. Comput. Security: Comput. Security, 2008, pp. 192–206.
[14] R. Agrawal and R. Srikant, "Privacy-preserving data mining," ACM Sigmod Rec., vol. 29, pp. 439–450, 2000.