ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 10, March 2016

NEURAL NETWORK CLASSIFER OVER ENCRYPTED DATA IN CLOUD ENVIRONMENT

*K.R.Raji **Ms.Archana.V.Nair.S*** Ms.K.R.Raghi Ponjesly College of Engineering
rajikrnair0@gmail.com, archananair@gmail.com,raghinair012@gmail.com

Abstract -- Data Mining has wide applications in many areas such as banking, medicine, scientific research and among government agencies. Classification is one of the commonly used tasks in data mining applications. The cloud computing, users have the opportunity to outsource their data, in encrypted form, as well as the data mining tasks to the cloud. Since the data on the cloud is in encrypted form. With the recent popularity of cloud computing, users now have the opportunity to outsource their data as well as the data management tasks to the cloud. However, due to the rise of various privacy issues, sensitive data need to be encrypted before outsourcing to the cloud. To achieve high accuracy, a neural network classifier which has been shown to outperform other learning models for many tasks. To achieve the privacy requirements, we use encryption in the following protocol: the data owner encrypts the data and sends the cipher texts to the third party to obtain a prediction from a trained model. The model operates on these cipher texts and sends back the encrypted prediction. In this protocol, not only the data remains private, even the values predicted are available only to the data owner. This method paves the way to build a secure cloud-based neural network prediction services without invading users' privacy.

Keywords - Security, NN Classifier, Outsourced Databases, Encryption

I.INTRODUCTION

Cloud computing paradigm 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.. The advantages that the cloud offers, 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. The 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 privacy/security requirements of the DMED problem on a cloud are threefold: (1) confidentiality of the encrypted data, (2) confidentiality of a user's query record, and (3) hiding data access patterns. Existing work on Privacy-Preserving Data Mining (either perturbation or secure multi-party computation based approach) 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. As a result, a novel methods to effectively solve the DMED problem assuming that the encrypted data are outsourced to a cloud. Specifically, we focus on the classification problem since it is one of the most common data mining tasks. Because each classification technique has their own advantage, to be concrete, this paper concentrates on executing the k-nearest neighbor classification method over encrypted data in the cloud computing environment.

II. EXISTING SYSTEM

The existing secret sharing techniques in SMC develop a PPkNN protocol. However, our work is different from the secret sharing based solution from the following two aspects. (i) Solutions based on the secret sharing schemes require at least three parties whereas our work requires only two parties. (ii) Hiding data access patterns is still an unsolved problem in the secret sharing based schemes, whereas our work protects data access patterns from both participating parties.

II.1 Privacy-Preserving Data Mining (PPDM)

Privacy Preserving Data Mining (PPDM) is defined as the process of extracting/deriving the knowledge about data without compromising the privacy of data. In the past decade, many privacy-preserving classification techniques have been proposed in the literature in order to protect user privacy. Agrawal and Srikant, Lindell and Pinkas introduced the notion of privacy-preserving under data mining applications. In particular to privacy preserving classification, the goal is to build a classifier in order to predict the class label of input data record based on the distributed training dataset without compromising the privacy of data.

A. Data Perturbation Method

In these methods, values of individual data records are perturbed by adding random noise in such way that the distribution of perturbed data look very different from that of actual data. After such a transformation, the perturbed data is sent to the miner to perform the desired data mining tasks. Data perturbation techniques cannot be applicable for semantically secure encrypted data. Also, they do not produce accurate data mining results due to the addition of statistical noises to the data.

B. Data Distribution Methods

Assume the dataset is partitioned either horizontally or vertically and distributed across different parties. During this process, data owned by individual parties is not revealed to other parties. Classification is one important task in many applications of data mining such as health-care and business. In cloud computing, data owner outsources his/her data to the cloud. The direct way to guard the outsourced data is to apply encryption on the data before outsourcing. The hosted data on the cloud is in encrypted form in our problem domain, the existing privacy preserving classification techniques are not sufficient and applicable to PPkNN due to the following reasons. (i) In existing methods, the data are partitioned among at least two parties, whereas in our case encrypted data are hosted on the cloud. (ii) Since some amount of information is loss due to the addition of statistical noises in order to hide the sensitive attributes, the existing methods are not accurate. (iii) Leakage of data access patterns: the cloud can easily derive useful and sensitive information about users' data items by simply observing the database access patterns.

II.2. Query processing over encrypted data

PPkNN is a more complex problem than the execution of simple kNN queries over encrypted data. One, the intermediate k-nearest neighbors in the classification process should not be disclosed to the cloud or any users. Secondly, even if we know the k-nearest neighbors, it is still very difficult to find the majority class label among these neighbors since they are encrypted at the first place to prevent the cloud from learning sensitive information. Third, the existing work did not address the access pattern issue which is a crucial privacy requirement from the user's perspective. Christo Ananth et al. [7] proposed a system which can achieve a higher throughput and higher energy efficiency. The S-BOX is designed by using Advanced Encryption Standard (AES). The AES is a symmetric key standard for encryption and decryption of blocks of data. In encryption, the AES accepts a plaintext input, which is limited to 128 bits, and a key that can be specified to be 128 bits to generate the Cipher text. In decryption, the cipher text is converted to original one. By using this AES technique the original text is highly secured and the information is not broken by the intruder. From that, the design of S-BOX is used to protect the message and also achieve a high throughput, high energy efficiency and occupy less area.

In this work, we only concentrate on the two party situations; thus, we adopted the Paillier cryptosystem. Two-party and multi- party (three or more parties) SMC protocols are complement to each other, and their applications mainly depend on the number of available participants. PPKNN support linear data only by using SRKNN and SCMC protocol. This protocol takes more time to train the data and high time computation. When dataset increases then the accuracy decreases.

III. THE PROPOSED PROTOCOL

A novel Probabilistic Neural Network classification protocol, denoted by PNN, which is constructed using the protocols as building blocks Neural networks are organized as layers one input layer, one or more hidden layers and an output layer. Hidden layers are made up of a number of neurons. Features/patterns are given to the network via the input layer, which are connected to one or more of the hidden layers. The actual processing is done in the hidden layers through a system of weighted connections. The hidden layers are connected to the output layer. The output layer provides the outcome of the processing or classification most neural networks contain some kind of learning function, which modifies the weights of the connections according to the training pattern presented to it. The proposed PPkNN protocol mainly consists of the following two stages:

•Stage 1 –Probabilistic Neural Network Classifier (PNN)
•Stage 2 - Secure Computation of Majority Class (SCMCk).

The Paillier cryptosystem is an additive homomorphic and probabilistic asymmetric encryption scheme. The encryption scheme is semantically secure. We simply use the well-known Paillier scheme in our implementations. However, to be more specific, in this paper we use the original Paillier cryptosystem.

Key generation : Randomly generate the prime numbers p and q.

Calculate : $n = p*q$ and $phi= (p-1)*(q-1)$

Find : $g=n+1$ and $\lambda=lcm\,(p-1,q-1)$

$\mu=(L(g^{\lambda} mod n^{2}))^{-1} \bmod n$

Where,

$L (u) = u-1/n$

$u = g^{\lambda} mod n^{2}$

Encryption : $C=g^{m}.r^{n} mod n^{2}$

Where,

C=cipher text

m=message

Decryption : $D=Lt*\mu \bmod n$

Where,

$t=c^{\lambda} \bmod n^{2}$

Lt= (t-1)/n

1. Data preprocessing Encryption
2. PNN classification
3. Secure computation
4. Security analysis

Data preprocessing and encryption:

The Car Evaluation dataset from the UCI KDD archive is used. It consists of 1,728 records and six attributes. Also, there is a separate class attribute and the dataset is categorized into four different classes. We encrypted this dataset attribute-wise, using the Paillier encryption. The Paillier cryptosystem is an additive homomorphic and probabilistic public-key encryption scheme whose security is based on the Decisional Composite Residuosity Assumption.

PNN classification:

Neural networks are organized as layers one input layer, one or more hidden layers and an output layer. Hidden layers are made up of a number of neurons. Features/patterns are given to the network via the input layer, which are connected to one or more of the hidden layers. The actual processing is done in the hidden layers through a system of weighted connections. The hidden layers are connected to the output layer. The output layer provides the outcome of the processing or classification most neural networks contain some kind of learning function, which modifies the weights of the connections according to the training pattern presented to it.

Secure computation:

Without loss of generality, let us assume that Alice's dataset D consists of w unique class labels. We assume that Alice outsources her list of encrypted classes to C1. That is, Alice outsources to C1 along with her encrypted data-base during the data outsourcing step. To start with, C1 and C2 jointly compute the encrypted frequencies of each class label. Upon receiving the data C2 decrypts it to get the randomized majority class label and sends it to Bob. Finally, upon receiving rq from C1and gq fromC2, Bob computes the output class label.

Security analysis:

The output data of Stage 1 which are passed as input to Stage 2 are in encrypted format. Therefore, the sequential composition of the two stages leads to PNN protocol and we claim it to be secure under the semi-honest model. It is clear that the proposed protocol protects the confidentiality of the data, the user's input query, and also hides data access patterns from Alice, C1 andC2.

Stage 2: Secure Computation of Majority Class (SCMCk)

Without loss of generality, suppose Alice's dataset D consists of w unique class labels denoted by $c = (c_1, . . c_w)$. We assume that Alice outsources her list of encrypted classes to C1. That is, Alice outsources $(E_{pk} (c_1), . . ,E_{pk} (c_w))$ to C1 along with her encrypted database D′ during the data outsourcing step. Note that, for security reasons, Alice may add dummy categories into the list to protect the number of class labels, i.e., w from C1 and C2. However, for simplicity, we assume that Alice does not add any dummy categories to c. During Stage 2, C1 with private inputs $\Lambda=(E_{pk}(c_1), . . . ,E_{pk}(c_w))$ and $\Lambda' = (E_{pk}(c_1'), . . . ,E_{pk}(c_k'))$, and C2 with sk securely compute $E_{pk}(c_q)$. Here c_q denotes the majority class label among $c_1', . . . , c_{k'}$. At the end of stage 2, only Bob knows the class label c_q. The overall steps involved in Stage 2 are shown in Algorithm. To start with, C1 and C2 jointly compute the encrypted frequencies of each class label using the k-nearest set as input. That is, they compute $E_{pk}(f (c_i))$ using(Λ,Λ') as C1's input to the secure frequency (SF) protocol, for $1 \leq i \leq w$. The output $(E_{pk} (f(c_1)), . . . ,E_{pk}(f(c_w)))$ is known only to C1. Then, C1 with $E_{pk}(f (c_i))$ andC2 with sk involve in the secure bit decomposition (SBD) protocol to compute $[f (c_i)]$, that is, vector of encryptions of the individual bits of $f (c_i)$, for $1 \leq I$

$\leq w$. After this, C1 and C2 jointly involve in the SMAXw protocol. Briefly, SMAXw utilizes the sub-routine SMAX to eventually compute $([f_{max}], E_{pk} (c_q))$ in an iterative fashion. Here $[f_{max}] =[max(f(c1), . . . , f(c_w))]$ and cq denotes the majority class out of Λ'. At the end, the output $([f_{max}], E_{pk} (c_q))$ is known only to C1. After this,C1 computes $q = E_{pk}(c_q + r_q)$,where r_q is a random number in Z_N known only to C1. Then, C1 sends q to C2 and r_q to Bob. Upon receiving q, C2 decrypts it to get the randomized majority class label $'q = D_{sk}(q)$ and sends it to Bob. Finally, upon receiving r_q from C1 and $'q$ from C2, Bob computes the output class label corresponding to q as $c_q = q - r_q \bmod N$.

Security Analysis of PNN under the Semi-honest Model

Here we provide a formal security proof for the proposed PNN protocol under the semi-honest model. Due to the encryption of q and by semantic security of the Paillier cryptosystem, Bob's input query q is protected from Alice, C1 and C2 in our PNN protocol.

Probabilistic Neural Network (PNN) algorithm:

Step 1: pre-processing of data

- Collect the data for the PNN based prediction algorithm.
- Define the set of values for the training and testing purposes. Here from the literature, the collected. And then transform it to the format of PNN.

Step 2: training of PNN

- Train the network, with the help of PNN training algorithm, input and output matrices.

- Identify the suitable value of spread constant s. The value of s cannot be selected arbitrarily. A too small s value can result in a solution that does not generalize from the input/ target vectors used in the design. In contrast, if the spread constant is large enough, the radial basis neurons will output large values for all the inputs used to design a network.

- Step 3: testing of PNN
  - Define the matrix for the testing of the PNN network.
  - Verify the predicted and actual values for the efficiency check of the network.

SCMC Algorithm:

```
SCMCk ($E_{pk}(c_1'), \ldots, E_{pk}(c_k')) \rightarrow c_q$
Require: $(E_{pk}(c1), \ldots, E_{ok}(c_w)), (E_{pk}(c_1'), \ldots, E_{pk}(c_k'))$are known only to C1; sk is known only to C2
1: C1 and C2:
  (a). $(E_{pk}(f(c_1)), \ldots, E(f(c_w))) \leftarrow SF(\Lambda,\Lambda')$, where $\Lambda = (E_{pk}(c_1), \ldots, E_{pk}(c_w)), \Lambda' = (E_{pk}(c_1'), \ldots, E_{pk}(c_k'))$
  (b). for $i = 1$ to w do:
    • $[f(c_i)] \leftarrow SBD(E_{pk}(f(c_i)))$
  (c). $([fmax], E_{pk}(c_q)) \leftarrow SMAXw(\Psi_1, \ldots, \Psi_w)$, where $\Psi_i = ([f(ci)], E_{pk}(c_i))$, for $1 \le i \le w$
2: C1:
  (a). $\gamma_q \leftarrow E_{pk}(c_q) * E_{pk}(r_q)$, where $r_q \in R\ Z_N$
  (b). Send $\gamma_q$ to C2 and $r_q$ to Bob
3: C2:
  (a). Receive $\gamma_q$ from C1
  (b). $\gamma'_q \leftarrow D_{sk}(\gamma_q)$; send $\gamma'_q$ to Bob
4: Bob:
  (a). Receive $r_q$ from C1 and $\gamma'_q$ from C2
  (b). $c_q \leftarrow \gamma'_q - r_q \bmod N$
```

The goal of PNN is to protect data confidentiality and hide data access patterns. In this paper, to prove a protocol's security under the semi-honest model, we adopted the well-known security definitions from the literature of secure multiparty computation (SMC).

IV.SYSTEM OVERVIEW

Read the car evaluation dataset from KDD. It consists of 1,728 records and six attributes. There is a separate class attribute and the dataset is categorized into four different classes. Encrypt the dataset attribute-wise, using the Bitwise XOR operation. The Bitwise XOR operation split the string data by removing comas. Strings are converted into binary format using ASCII keyword. Convert the binary values into encrypted format by using Paillier cryptosystem. The Paillier cryptosystem is a public-key encryption scheme. It consists of 3 schemes:

- Key generation
- Encryption
- Decryption

The encrypted data send to cloud. Then the cloud will search the similar data. That data's are grouped together, weightage value and training is given to the data by using PNN. From that data, find the majority class by means of SCMC and produce the classify data. Then analyze the performance of our solution through various experiments.

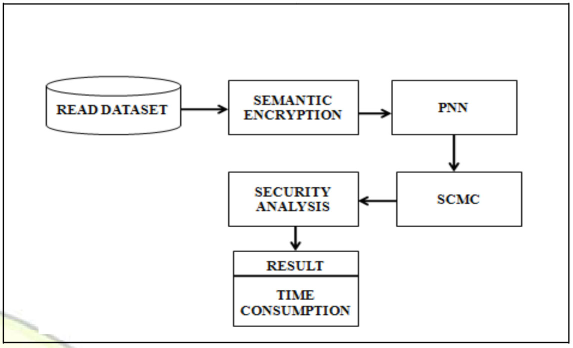

Fig: Architecture Design

V. Complexity Analysis

The computation complexity of Stage 1 in PPkNN is bounded by O (n) instantiations of SBD and SSED, O (k) instantiations of SMINn, and O(n * k * l) instantiations of SBOR. We emphasize that the computation complexity of the SBD protocol proposed in is bounded by O (l) encryptions and O (l) exponentiations (under the assumption that encryption and decryption operations based on Paillier cryptosystem take similar amount of time). Also, the computation complexity of SSED is bounded by O (m) encryptions and O (m) exponentiations. In addition, the computation complexity of SMINn is bounded by O (l * n * log2 n) encryptions and O (l * n * log2 n) exponentiations. Here the computation complexity of SF is bounded by O (k * w) encryptions and O (k * w) exponentiations. Therefore, the total computation complexity of Stage 2 is bounded by O (w* (l + k +l * log2 w)) encryptions and exponentiations. In general, w ≪ n, therefore, the computation cost of Stage 1 should be significantly higher than that of Stage 2.

VI. EMPIRICAL RESULTS

In this section, we discuss some experiments demonstrating the performance of our PNN protocol under different parameter settings. We used the Paillier cryptosystem as the underlying additive homomorphic encryption scheme and implemented the proposed PNN protocol in C. To the best of our knowledge, our work is the first effort to develop a secure PNN classifier under the semi honest model. Thus, there is no existing work to compare with our approach. Therefore, we evaluate the performance of our PNN protocol under different parameter settings.

VI.1. Dataset and Experimental Setup

For our experiments, used the Car Evaluation dataset from the UCI KDD archive. The dataset consists of 1728 data records (i.e., n = 1728) with 6 input attributes (i.e., m = 6). Also, there is a separate class attribute and the dataset is categorized into four different classes (i.e., w = 4). We encrypted this dataset attribute-wise, using the Paillier encryption whose key size is varied in our experiments, and the encrypted data were stored on our machine. Based on our PPkNN protocol, we then executed a random query over this encrypted data.

VII. CONCLUSION

Classification is an important task in many data mining applications. PPKNN support linear data only by using SRKNN and SCMC protocol. This protocol takes more time to train the data and high time consumption. When dataset increases then the accuracy decreases. To achieve high accuracy the PNN classifier is used. This classifier reduces the time consumption and it takes less time to train the dataset. Our protocol protects the confidentiality of the data, user's input query, and hides the data access patterns. We also evaluated the performance of our protocol under different parameter settings. Since improving the efficiency of SMINn is an important first step for improving the performance of our PNN protocol.

REFERENCES

[1] C. C. Agrawal and P. S. Yu. A general survey of privacy-preserving data mining models and algorithms. Privacy-preserving data mining, pages 11–52, 2008.

[2] Tope Komal, Rane Ashutosh, Rahate Roshan and S.M. Nalawade "Encryption and Decryption using ArtificialNeural Network", International Advanced Research Journal in Science, Engineering and Technology Vol. 2, Issue 4, April 2015.

[3] R. Agrawal and R. Srikant. Privacy-preserving data mining. In ACM Sigmod Record, volume 29, pages 439–450.ACM, 2000.

[4] Khalil Shihab, "A Back propagation Neural Network for Computer Network Security", Journal of Computer Science 2 (9): 710-715, 2006.

[5] R.S Kale and S.Y. Kute, " Five Layers Artificial Neural Network System [Ann] to Design a Concrete Mix Based on I.S.I. Method", CACE Volume 2, Issue 4 Oct. 2014.

[6] Khaled M.G. Noaman and Hamid Abdullah Jalab, "Data Security Based on Neural Networks" 2015.

[7] Christo Ananth, H. Anusuya Baby, "S-Box using AES Technique", International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 3, March – 2014, pp 285-290

[8] Abdelhadi LOTFI and Abdelkader BENYETTOU, "A reduced probabilistic neural network for the classification of large databases", Turk J Elec Eng & Comp Science 22: 979 - 989,2014.

[9] M. Bohanec and B. Zupan. The UCI KDD Archive. University of California, Department of Information and Computer Science, Irvine, CA, 1997.

[10] A.P. Mittal and Hasmat Malik , Vihan Talur and Saarang Rastogi, "Probabilistic Neural Network Based Identification of External faults Experienced by 3-Phase Induction motors", IJCEM International Journal of Computational Engineering & Management, Vol. 18 Issue 3, May 2015.