IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 4, April 2017

Privacy Preservation Data Classification using Machine Learning Techniques

Rahul Bankar[1], Sudarshan Bute[2], Sujit Gadekar[3], Avinash Ghodake[4], Santosh Javheri[5]

Student, Department of Computer Engineering, JSPM’s Rajarshi Shahu College of Engineering, Pune, India [1,2,3,4]

Teacher, Department of Computer Engineering, JSPM’s Rajarshi Shahu College of Engineering, Pune, India [5]

Abstract: In today's world, many people interested to share the data, but they afraid about disclose of data so they want secure their data with communication so, the concept of privacy preservation of data is come in picture. Back-propagation is the most effective algorithm for training deep learning models. To protect the private data, the proposed model having the BGV encryption scheme to encrypt the private data and perform the high-order back propagation algorithm on the encrypted data. Sigmoid function as a polynomial function with the BGV encryption. The proposed algorithm helps to improve the efficiency of back-propagation. BGV is Homomorphic encryption technique that allows computations to be carried out on cipher text. Homomorphic encryption is the conversion of data into cipher text that can be help to analyse and work with as if it were still in its original form. Furthermore, the BGV encryption scheme is used to protect the private data during the learning process. Experiments show that our proposed scheme is secure and efficient.

Keywords: Privacy, Back-propagation, the BGV encryption scheme.

I. INTRODUCTION

With the rapid development of the internet of things, social networks and e-commerce in recent years, we have entered the era of big data. Deep learning models have been proved to have a great ability of learning features and hierarchical representations of database by supervised strategies. However, the huge amount of data poses an important challenge on data. Specially, as the most effective training algorithm, back-propagation finds it difficult to satisfy the real-time requirement of database feature learning since it is of high time complexity.

Therefore, how to improve the efficiency of back propagation learning has become a problem needed to be settled urgently in data processing and mining these days. However, privacy concerns bring forward in the database computing because some data in datasets, such as identity information and the association rules hidden in the database are considered as private data, which may contain sensitive data of governments or proprietary information of the enterprises. Sensitive data is easily disclosed during the process of the computation.

Disclosure of sensitive data is not only a privacy issue but of legal concerns according to privacy protection laws such as the Health Insurance Portability and Accountability Act (HIPAA). A large number of works have focused on the secure back propagation learning, which can be grouped by two categories: data perturbation methods and cryptographic methods. Methods of the first type protect the private data by adding noise to the source data Perhaps; the noise will reduce the accuracy of the back-propagation learning.

Methods of the second type use the encryption algorithms to preserve the privacy. However, they can only work in the scenario of multiple parties’ collaboration. Aiming at this problem, we propose a privacy preserving back-propagation algorithm based on the full homomorphic encryption scheme. The proposed algorithm improved the efficiency of back-propagation learning by offloading the expensive operations on the databases.

Furthermore, to prevent the disclosure of private data, we use the full homomorphic encryption scheme to encrypt the source data. Specially, the BGV encryption algorithm that is the currently most efficient full homomorphic encryption scheme is utilized in the proposed algorithm.

II. RELATED WORK

We try to implement this project on client server model. So, our future/related work is to implement this project on cloud.

DOI10.17148/IJARCCE.2017.6457

III. SYSTEM ARCHITECTURE

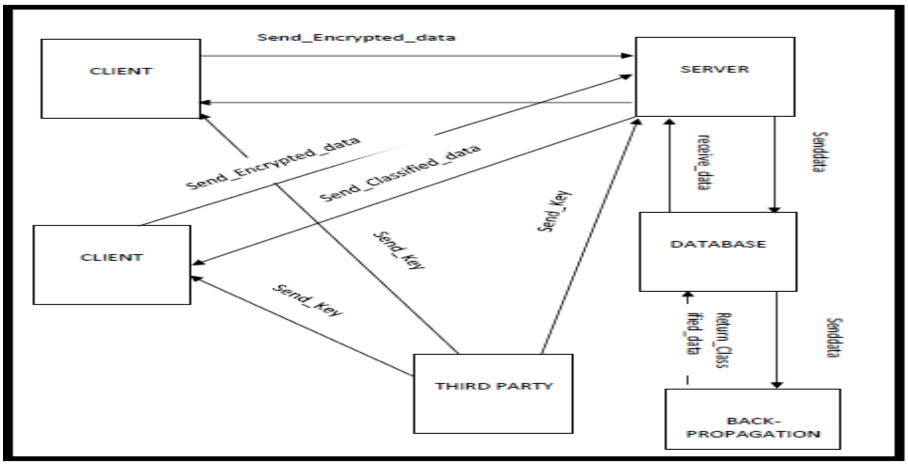

Fig. 1 System Architecture

IV. PROPOSED SCHEME

A. BACK PROPAGATION

```
Input: x={x1,x2,…,xn}, $\text{iteration}_{\text{max}}$ , η , threshold
Output: W, b, W’, b’
begin:
Initialize the parameters randomly.
for iteration = 1,2, … , $\text{iteration}_{\text{max}}$ do
for sample=1,2,…,N do
//feed forward
        for j=1,2, …, m do
      $z_j^{(2)} = W.X + b_j$
      $a_j^{(2)} = f\left(z_j^{(2)}\right)$
for i=1,2, …, n do
          $z_i^{(3)} = W'.a^{(2)} + b_i'$
          $h_{W,b}(X) = a_i^{(3)} = f(z_i^{(3)})$
        if $J_{TAE}$ (θ)> threshold then
//back-propagation
for i=1,2, …, n do
        $\sigma_i^{(3)} = \left(a_i^{(3)}.\left(1 - a_i^{(3)}\right)\right).\left(a_i^{(3)} - y_i\right)$
        for  j=1,2,…, m do
        $\sigma_j^{(2)} = \left(\sum_{i=1}^{n} w_{ij}^{(2)}.\sigma_i^{(3)}\right) f'\left(z_j^{(2)}\right)$
        for i=1,2, …, n do
        $\Delta b_i^{(2)} = \Delta b_i^{(2)} + \sigma_j^{(3)}$
         for j=1,2, …, m do
       $\Delta W_{ij}^{(2)} = \Delta W_{ij}^{(2)} + a_j^{(2)}.\sigma_i^{(3)}$
         for j= 1,2,…,m do
              $\Delta b_{ij}^{(2)} = \Delta b_j^{(1)} + \sigma_j^{(2)}$
              for i=1,2,…,n do
              $\Delta W_{ji}^{(1)} = \Delta\Delta W_{ji}^{(1)} + x_i.\sigma_j^{(2)}$
End
```

B. FEED FORWARD

A **feed forward neural network** is an artificial neural network wherein connections between the units do not form a cycle. As such, it is different from recurrent neural networks. The feed forward neural network was the first and

DOI10.17148/IJARCCE.2017.6457

simplest type of artificial neural network devised. In this network, the information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network.

```
Input: $x = \{x1, x2, \dots, x_n\}, \{W, b, W', b'\}$
Output: $\{z^{(2)}, z^{(3)}; a^{(2)}, a^{(3)}\}$
Begin:
for $iteration = 1,2, \dots, iteration_{max}$ do
for sample= 1,2,…,N do
        for j=1,2,,,m do
//Using secure addition and multiplication to calculate:
        $z_j^{(2)} = W.X + b_j$
//Using Algorithm 2 to Calculate:
        $a_j^{(2)} = f(z_j^{(2)})$
        for i= 1,2,…,n do
//Using secure addition and multiplication to calculate:
$z_i^{(3)} = W'.a^{(2)} + b_i'$
//Using Algorithm 2 to calculate:
            $h_{w,b}(X) = a_i^{(3)} = f(z_i^{(3)})$
  End
```

C. BGV ALGORITHM

```
Input: ciphertext of x: C(x), a: C(a), b: C(b) and c: C(c).
Output: ciphertext of y: C(y)
begin
Using secure addition: $C_1 = C(b) \times C(x)$
Using secure multiplication:
                                   $C_2 = C(c) \times C(x) \times C(x) \times C(x)$
Using secure addition:
 $C(y) = C(a) + C_1 + C_2$
end
```

V. MATHEMATICAL MODEL

Let S is the Whole System Consists:
S = {I, P, O}
Where, S = System
I = Input
P = Process
O = Output

I = {t, P_uKU }

Where I = Input,

P_uKU = Public Key of User

P = {SC, T_r, T_t, CD, GR}

Where P = Process,

SC = Scaling,

T_r = Train the Network,

T_t = Testing,
CD = Classification of Data,

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 4, April 2017

GR = Generation OF Result

O = [[DEC]]

Where O = Output,

DEC = Decision

Users side parameters = {P_uKU, P_rKDU, t}

Where P_uKU = Public Key of User,

P_rKDU = Private Key of User

Server side Parameters = { [[t]]}

U = {u_1, u_2,..u_n}

Where U = User,

u_1, u_2,..u_n = No of User

UD = {t_1, t_2,...t_n}

Where UD = User Data,

t_1, t_2,...t_n = Users Data

U = EP_uKU{UD}

{t_1, t_2,...t_n} {[[t_1]],[[t_2]]…[[t_n]]}

Where U = User,

UD = User Data,
t_1, t_2,...t_n= Users Data

P = {SC, T_r, T_t, }

Where P = Process,

SC = Scaling,

T_r = Train the Network,

T_t = Testing

U = send {[[UD]]} to server

Then server,

SS = {T_t, CD, GR}

Where T_t = Testing,

CD = Classification of Data
Output = [[Dec]]

DOI10.17148/IJARCCE.2017.6457

t_1, t_2,...t_n = Users Data

P = {SC, T_r, T_t, }

Where P = Process,

SC = Scaling,

T_r = Train the Network,

T_t = Testing

U = send {[[UD]]} to server

Then server,

SS = {T_t, CD, GR}

Where T_t = Testing,

CD = Classification of Data

Output = [[Dec]]

VI. CONCLUSIONS

We proposed the privacy preservation data classification using machine learning technique. Many people wanted to secure their data with communication, hence we implemented BGV encryption scheme to encrypt the private data and perform the high-order back propagation algorithm on the encrypted data

ACKNOWLEDGMENT

The heading of the Acknowledgment section and the References section must not be numbered. Causal Productions wishes to acknowledge Michael Shell and other contributors for developing and maintaining the IEEE LaTeX style files which have been used in the preparation of this template.

REFERENCES

[1] Yuan Zhang, Sheng Zhong Neural Compute& Applic (2013) 22 (Suppl 1):S269–S282 DOI 10.1007/s00521-012-1000-8A privacy-preserving algorithm for distributed training of neural network ensembles.

[2] Secretan J, Georgiopoulos M, Castro J (2007) A privacy preserving probabilistic neural network for horizontally partitioned databases. In: Proceedings of international joint conference on neural networks, Orlando, FL, USA.

[3] T. Condie, P. Mineiro, N. Polyzotis, and M. Weimer, "Machine learning on big data," in Proc. IEEE Int. Conf. Data Eng., 2013,pp. 1242–1244.

[4] C. Gentry, "Fully homomorphic encryption using ideal lattices," in Proc. ACM Symp. Theory Comput., 2009, pp. 169–178.

[5] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in Proc. Int. Conf. Theory Appl. Cryptographic Techn., 1985, pp. 10–18.

[6] Z Brakerski, "Fully homomorphic encryption without modulus switching from classical GapSVP," in Proc. 32nd Annu. Cryptol.Conf. Adv. Cryptol., 2012, pp. 868–886.

[7] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptoticallyfaster, attribute-based," in Proc. 33rd Annu. Cryptol. Conf. Adv. Cryptol., 2013, pp.75–92.

[8] Chen T, Zhong S (2009) Privacy preserving back-propagation neural network

[9] Yang Z, Zhong S, Wright R (2005) Privacy-preserving classification of customer data without loss of accuracy. In: Proceedings of the 5th SIAM international conference on data mining (SDM)

[10] Ankur Bansal, Tingting Chen, Sheng Zhong (2009) Privacy preserving Back-propagation neural network learning over arbitrarily partitioned data.

[11] Blake CL, Merz CJ (1998) UCI repository of machine learning databases, Department of Information and Computer Science, University of California, Irvine, CA. http://www.ics.uci.edu/mlearn/MLRepository.html.