Privacy Preserving Back Propagation Neural Network Learning using Signature Scheme

C. Nikitha[1] B. Nithila[2] B. Praveen Kumar[3]
[1,2] U. G. Scholar [3]Assistant Professor
[1,2,3]Department of Computer Science Engineering
[1,2,3] Velammal Institute of Technology

Abstract—To improve the accuracy of learning result, in practice multiple parties may collaborate through conducting joint Back propagation neural network learning on the union of their respective data sets. During this process no party wants to disclose her/his private data to others. Existing schemes supporting this kind of collaborative learning are either limited in the way of data partition or just consider two parties. There lacks a solution that allows two or more parties, each with an arbitrarily partitioned data set, to collaboratively conduct the learning. This paper solves this open problem by utilizing the power of cloud computing. In our proposed scheme, each party encrypts his/her private data locally and uploads the cipher texts into the cloud. The cloud then executes most of the operations pertaining to the learning algorithms over cipher texts without knowing the original private data. By securely offloading the expensive operations to the cloud, we keep the computation and communication costs on each party minimal and independent to the number of participants. To support flexible operations over cipher texts, we adopt and tailor the BGN doubly holomorphic encryption algorithm for the multi-party setting. Numerical analysis and experiments on commodity cloud show that our scheme is secure, efficient and accurate.

Key words: Privacy reserving, Learning, Neural Network, Back-Propagation, Cloud computing, Computation Outsource.

I. INTRODUCTION

We address to the problem of Privacy Preserving Back Propagation Algorithm for a Vertically Partitioned Dataset. To enhance cooperation's in learning, it is important to address the privacy concern of each data holder by extending the privacy preservation notion to original learning algorithms. In this paper, we focus on preserving the privacy in an important learning model, multilayer neural networks. We present a privacy preserving multiparty distributed algorithm of back propagation which allows a neural network to be trained without requiring either party to reveal her data to the others. We provide complete correctness and security analysis of our algorithms. The effectiveness of our algorithms is verified by experiments on various real world data sets.

II. RELATED WORK

Several privacy preserving BPN network learning schemes have been proposed recently. Schlitter [19] introduces a privacy preserving BPN network learning scheme that enables two or more parties to jointly perform BPN network learning without disclosing their respective private data sets. But the solution is proposed only for horizontal partitioned data. Moreover, this scheme cannot protect the intermediate results, which may also contain sensitive data, during the learning process. Chen et. al. [6] proposes a privacy preserving BPN network learning algorithm for two-party scenarios. This scheme provides strong protection for data sets including intermediate results. However, it just supports vertically partitioned data. To overcome this limitation, Bansal et. al. [4] enhanced this scheme and proposed a solution for arbitrarily partitioned data. Nevertheless, this enhanced scheme, just like [6], was proposed for the two-party scenario. Directly extending them to the multi-party setting will introduce a computation/ communication complexity quadratic in the number of participants. In practical implementation, such a complexity represents a tremendous cost on each party considering the already expensive operations on the underlying groups such as Elliptic Curves. However, [4] just considers the two-party scenario though it supports arbitrarily partitioned dataset. To our best knowledge, none of existing schemes have solved all these challenges at the same time. There still lacks an efficient and scalable solution that supports collaborative BPN network learning with privacy preservation in the multi-party setting and allows arbitrarily partitioned datasets.

Let be a 2-DNF formula on Boolean variables x1, xn 2 {0, 1}. We present a Homomorphism public key encryption scheme that allows the public evaluation of given an encryption of the variables x1, xn. In other words, given the encryption of the bits x1, xn, anyone can create the encryption of (x1, xn). More generally, we can evaluate quadratic multi-variety polynomials on ciphertext provided the resulting value falls within a small set. We present a number of applications of the system: In a database of size n, the total communication in the basic step of the Kushilevitz-Ostrovsky PIR protocol is reduced from An efficient election system based on homomorphism encryption where voters do not need to include non-interactive zero knowledge proofs that their ballots are valid. The election system is proved secure without random oracles but still efficient. A protocol for universally verifiable computation.

With the development of distributed computing environment, many learning problems now have to deal with distributed input data. To enhance cooperation's in learning, it is important to address the privacy concern of each data holder by extending the privacy preservation notion to original learning algorithms. In this paper, we focus on preserving the privacy in an important learning model,

multi-layer neural networks. We present a privacy preserving two-party distributed algorithm of back-propagation which allows a neural network to be trained without requiring either party to reveal her data to the other. We provide complete correctness and security analysis of our algorithms. The effectiveness of our algorithms is verified by experiments on various real world datasets.

A. EXISTING WORK

Existing schemes supporting this kind of collaborative learning are either limited in the way of data partition or just consider two parties. There lacks a solution that allows two or more parties, each with an arbitrarily partitioned data set, to collaboratively conduct the learning. The participants from different trust domains may not want to disclose their private data sets, which may contain privacy or proprietary information, to anybody else. Provides privacy preservation for multi- party collaborative BPN network learning over arbitrarily partitioned data. To support multi-party secure scalar product and introduce designs that allows decryption of arbitrary large messages.

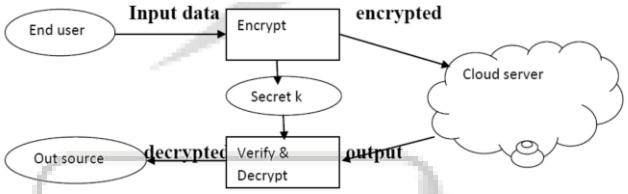

Fig 1.1 Existing System Diagram

Fig. 1: AES Encryption Decryption

The below tabular column shows the necessary details regarding the attribute based encryption variations.

Dataset	Sample	Architecture	Class	Epochs	Learning Rate
Iris	150	4-5-3	3	80	0.1
Diabetes	768	8-12-1	2	40	0.1
kr-vs-kp	3196	36-15-1	2	20	0.1

Table. 1: AES Variations

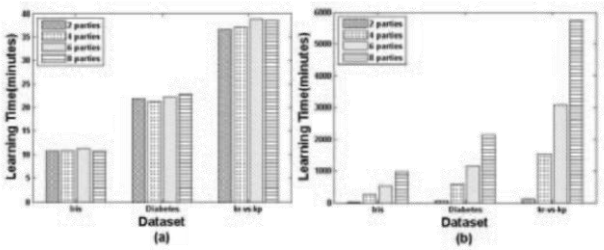

Fig. 2: Graph AES Graphical Representation

III. PROPOSED WORK

In this work, we address this open problem by incorporating the computing power of the cloud. The main idea of our scheme can be summarized as follows: each participant first encrypts her/his private data with the system public key and then uploads the ciphertext to the cloud; cloud servers then execute most of the operations pertaining to the learning process over the ciphertext and return the encrypted results to the participants; the participants jointly decrypt the results with which they update their respective weights for the BPN network. During this process, cloud servers learn no privacy data of a participant even if they collude with all the rest participants. Through off-loading the computation tasks to the resource-abundant cloud, our scheme makes the computation and communication complexity on each participant independent to the number of participants and is thus highly scalable. For privacy preservation, we decompose most of the sub-algorithms of BPN network into simple operations such as addition, multiplication, and scalar product. To support these operations over ciphertext, we adopt the BGN (Boneh, Goh and Nissim) 'doubly homomorphism' encryption algorithm [5] and tailor it to split the decryption capability among multiple participants for collusion-resistance decryption. As decryption of [5] is limited to small numbers, we introduce a novel design in our scheme such that arbitrarily large numbers can be efficiently decrypted. To protect the intermediate data during the learning process, we introduce a novel random sharing algorithm to randomly split the data without decrypting the actual value. Thorough security analysis shows that our proposed scheme is secure. Experiments conducted on Amazon Elastic Compute Cloud (Amazon EC2) [15], over real datasets from UCI Machine Learning Repository [12], show that our scheme significantly outperform existing ones in computation/communication cost and accuracy loss. Our contribution can be summarized as follows:

1) To our best knowledge, this paper is the first that provides privacy preservation for multi-party (more than two parties) collaborative BPN network learning over arbitrarily partitioned data;
2) Thorough analysis investigating privacy and efficiency guarantees of proposed scheme is presented; real experiments on Amazon Cloud further show our scheme's several magnitudes lower computation/ communicational costs than the existing ones.
3) We tailor [5] to support multi-party secure scalar product and introduce designs that allow decryption of arbitrary large messages. These improvements can be used as independent general solutions for other related applications. The rest of this paper is organized as follows. Section 2 presents the models and assumptions. In section 3 we introduce technique preliminaries which are followed by detailed description of our proposed scheme in section 4.
4) Section 5 evaluates our proposed scheme. We conclude our work in section 6.

This article focuses on a particular type of neural network model, known as a "feed-forward back-propagation network". This model is easy to understand, and can be easily implemented as a software simulation. First we will discuss the basic concepts behind this type of NN, and then we'll get into some of the more practical application ideas. The back propagation algorithm trains a given feed-forward multilayer neural network for a given set of input patterns with known classifications. When each entry of the sample set is presented to the network, the network examines its output response to the sample input pattern.

In this work, we address this open problem by incorporating the computing power of the cloud. The main idea of our scheme can be summarized as follows: each

participant first encrypts her/his private data with the system public key and then uploads the cipher texts to the cloud; cloud servers then execute most of the operations pertaining to the learning process over the cipher texts and return the encrypted results to the participants; the participants jointly decrypt the results with which they update their respective weights for the BPN network. During this process, cloud servers learn no privacy data of a participant even if they collude with all the rest participants.

IV. IMPLEMENTATION DIAGRAM

To connect with Cloud .user must give their username and password then only they can able to connect the server. If the user already exits directly can login into the server else user must register their details such as username, password, Email id, City and Country into the server. Server will create the account for the entire user to maintain upload and download rate

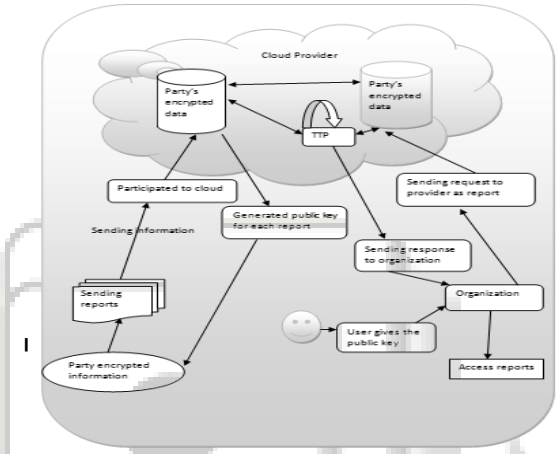

Fig. 3: Proposed Methodology Architecture

We aim at enabling multiple parties to jointly conduct BPN network learning without revealing their private data. The input data sets owned by the parties can be arbitrarily partitioned. The computational and communicational costs on each party shall be practically efficient and the system shall be scalable.

A. Proposed System Model Explanation Back-Propagation Neural Network Learning

Definition:

We design a secret sharing algorithm that allows the parties to decrypt only the random shares of the intermediate values. The random shares allow the parties to collaboratively execute the following steps without knowing the actual intermediate values.

1) Algorithm:

Back-Propagation neural network learning algorithm is mainly composed of two stages: feed forward and error back − propagation configuration for a three layer (a-bc) BP network. We use vector{x1, x2, ・ ・ ・ , xa} to denote the values of input nodes, vector{h1, h2, ・ ・ ・ , hb} and vector{o1, o2, ・ ・ ・ , oc} to denote the values of hidden layer nodes and output layer nodes respectively. Where, jk is used to represent the weight connecting the input layer node k and the hidden layer node j. wo ij denotes the weight connecting j and the output layer node i, where 1 ≤ k ≤ a, 1 ≤ j ≤ b, 1 ≤ i ≤ c.

Algorithm 1: Back-Propagation Neural Network Learning Algorithm

Input: N input sample vectors $\vec{V_i}, 1 \leq i \leq N$, with a dimensions, $iteration_{max}$, learning rate η, target value t_i, sigmoid function $f(x) = \frac{1}{1+e^{-x}}$

Output: Network with final weights: $w^h_{jk}, w^o_{ij}, 1 \leq k \leq a, 1 \leq j \leq b, 1 \leq i \leq c$

begin

Randomly Initialize all w^h_{jk}, w^o_{ij}.

for $iteration = 1, 2 \cdots, iteration_{max}$ **do**

for $sample = 1, 2 \cdots, N$ **do**

//Feed Forward Stage:

for $j = 1, 2 \cdots, b$ **do**

$h_j = f(\sum_{k=1}^{a} x_k * w^h_{jk})$

for $i = 1, 2 \cdots, c$ **do**

$o_i = f(\sum_{j=1}^{a} h_j * w^o_{ij})$

if $Error = \frac{1}{2}\sum_{i=1}^{c}(t_i - o_i)^2 > threshold$ **then**

//Back-Propagation Stage:

$\Delta w^o_{ij} = -(t_i - o_i) * h_j$

$\Delta w^h_{jk} = -h_j(1 - h_j)x_k \sum_{i=1}^{c}[(t_i - o_i) * w^o_{ij})]$

$w_{ij} = w_{ij} - \eta \Delta w_{ij}$

$w^h_{jk} = w^h_{jk} - \eta \Delta w^h_{jk}$

else

//Learning Finish

break

Fig. 4: Algorithm

V. PERFORMANCE ANALYSIS

A. SECURITY ANALYSIS

Propose an algorithm that allows multiple parties to perform secure scalar product and homomorphism addition operations on cipher texts using cloud computing. Party encrypts her/his data with the system public key and uploads the cipher texts to the cloud. The cloud servers compute the sum of original messages based on their cipher texts.

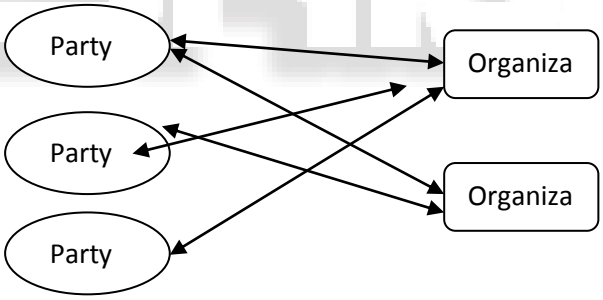

Fig. 5: Security Analysis Architecture

B. NUMERIC ANALYSIS

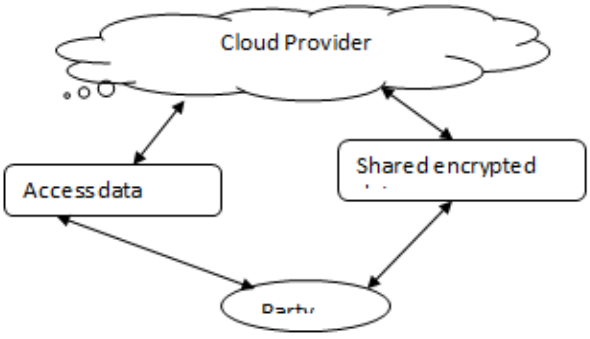

Fig. 6: Numeric Analysis Architecture

In practice, however, it is hard to guarantee that the final results (numbers) are always small enough for the Pollard's

lambda method to efficiently decrypt. This is either because the numbers contained in the vectors are too large, or the vectors are too long (of high dimension). To overcome this limitation, we propose to let the data holders divide the numbers, if they are large, into several numbers, and the cloud then decrypt the smaller ”chunks” with which the final result can be recovered. The decryption process can be parallelized for efficiency.

VI. CONCLUSION

We proposed the first secure and practical multi-party BPN network learning scheme over arbitrarily partitioned data. In our proposed approach, the parties encrypt their arbitrarily partitioned data and upload the ciphertext to the cloud. The cloud can execute most operations pertaining to the BPN network learning algorithm without knowing any private information. The individual keys generated for each and every party in the particular organisation. Whenever we login into the cloud we can simply view how many times ours has been accessed.

REFERENCES

[1] The health insurance portability and accountability act of privacy and security rules. URL: http://www.hhs.gov/ocr/privacy.
[2] National standards to protect the privacy of personal health information. URL: http://www.hhs.gov/ocr/hipaa/finalreg.html.
[3] M. Abramowitz and I. A. Stegun. Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables. Dover books on mathematics. Dover, New York, 1964.
[4] A. Bansal, T. Chen, and S. Zhong. Privacy preserving back propagation neural network learning over arbitrarily partitioned data. Neural Compute. Appl., 20(1):143–150, Feb. 2011.
[5] D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-dnf formulas on ciphertext. In Proceedings of the Second international conference on Theory of Cryptography, TCC’05, pages 325–341, Berlin, Heidelberg, 2005.
[6] T. Chen and S. Zhong. Privacy-preserving back propagation neural network learning. Trans. Neur. Netw., 20(10):1554–1564, Oct. 2009.
[7] L. Cun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Handwritten digit recognition with a back-propagation network. In Advances in Neural Information Processing Systems, pages 396–404. Morgan Kaufmann, 1990.
[8] S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Over-encryption: management of access control evolution on outsourced data. In Proceedings of the 33rd international conference on Very large data bases, VLDB ’07, pages 123–134. VLDB Endowment, 2007.
[9] T. El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In Proceedings of CRYPTO 84 on Advances in cryptology, pages 10–18, New York, NY, USA, 1985.

