An Efficient Method to Provide Security using Back Propagation Neural Network Learning in Cloud Computing

Chhaya Varade[1] Pratik D.Gaikwad[2] Tejas M. Khedekar[3] Pratik S. Raka[4]
[1,2,3,4]G.H. Raisoni Institute of Engineering and Technology, Pune

Abstract— By using the efficiency of cloud computing we are providing an ideal way for multiparty communication using Back Propagation neural network learning using their respective data set. Security is provided, as no party wants to reveal their data to others. Earlier models lacked multiple participant communication, every one having their own arbitrary data set for learning. Here we havw represented a system in which the participants encodes his data to thecloud server. The cloud server then computes and performs learning process with back propagation learning algorithm over the encoded data. As the cloud performs the maximum process, the communication and computation cost is kept minimum on the participant's side.

Key words: Cloud Computing, Back Propagation, Neural Network Learning

I. Introduction

The accuracy achieved in learning by using 'Back-Propagation' is very effective in the field of neural networks. Thus this method is widely used in various application, be it small scale or large scale[7]. The result of applying this method is a high quality data. Collaborative learning as compared to the local data set learning includes large number of sample data set for learning. This has been achieved by the power of cloud computing[4], the participants carry out learning not only with their own data but also with data of other participants. With the cloud computing infrastructure participants who do not know each other carry collaborative learning.

There are situations where the participants in collaborative learning [2] want to secure their essential data i.e. other participants should not know what the data is, while carrying out learning as well. This is an major issue in cloud computing and internet based learning.

Thus a efficient and effective solution is needed to overcome such security issues, in which participants could carry learning over their arbitrary data sets, without disclosing their data to other non trusted parties/participants. This can be done by secure multiparty communication.

To achieve this scenario three main challenges are to be met. 1) To provide security to the data set and the output result generated of the participants during the back propagation learning process. 2)To guarantee of efficient and effective process, i.e. an cost efficient learning method while delivering the results. 3)And lastly there should be collaborative learning, different data set from different participants should be capable for learning using back-propagation neural network. The training data sets should be arbitrarily partitioned instead of single way of partition.

II. Existing work

There were many different concepts proposed by different people. A multiparty concept was introduced for learning horizontal partitioned data set by Schlitter. This lacked arbitrary partitioning and security for the intermediate results. Another multiparty concept was introduced by Chen and Zhong which worked on vertical partitioned data only[1],[3]. Bansal did introduce an solution for arbitrary partitioned data but it was not for multiparty scenario, it only supported two participants. Therefore there still lacks a solution for multiparty collaborative learning using arbitrary partitioned data sets.

III. Present theory

Therefore the current solution is solved by using cloud computing. In this system every participant encodes his data and then uploads it to the cloud, the required operations then will be carried out by the cloud server over that data. The cloud server then delivers the data to the participant ,which will be decoded to get the actual results, thus the respective weights will be updated in the BPN network. Hence this system is scalable as simple encoding makes computation and communication easy, security is henceforth maintained as no participant disclose their data. To protect the data during the learning process we have implemented a simple encoding technique, in which two important stages are carried out, the first stage is transformation of the data. and the second stage is scaling of the transformed data. This method is easy to implement and delivers a secure output. The data is changed as it passes through the encoding stage, then the BPN network processing is performed on the encoded data, which is send to the participant, this result is then decoded with the same parameters used for encoding, thus giving us the result on the appropriate participants data.

IV. System functionality

The system consist of three modules or we can say three main entities: the trusted authority, the participants and the cloud servers. Trusted Authority is the one who encodes and decodes the participant data, he is said to be trusted by the participant. It does not have any other rights except for key generation for encoding and decoding data sets. Whereas the participating party are the one who owns a private data set and wants to conduct BPN network learning with other participating parties. Hence a collaborative learning over the arbitrary data set is performed, which is not disclosed to the other parties. Therefore every participating party should have one or many computers connected in parallel, it is also necessary that every participant is connected to the cloud throughout the process

V. Security functionality

In the cloud, the participating parties do not trust each other; no party wants to disclose their private data to the other party. Therefore the inclusion of trusted authority is essential. The trusted authority knows only about the key generation. Secret key and issuing, it is unaware of the computation. Therefore unauthorized interruption and parties are avoided.

VI. Artificial Neural Network

Therefore The human brain provides defines a massive neural networks that can succeed at those cognitive, perceptual, and control tasks in which humans can succeed in. The brain is capable of driving interpretation on perceptual acts like recognition of faces, speech etc and control activities like body movements and functions. The advantage of the brain is its effective use of massive parallelism, the highly parallel computing structure, and the imprecise information processing capability. There are more than 10 billion interconnected neurons in the human brain. Every neuron is a cell which uses biochemical reactions to receive, process, and transmit information.

Artificial Neural Networks (ANN)[6] have been developed as generalization of mathematically models of biological nervous systems. A first wave of interest in neural networks emerged after the introduction of simplified neurons by McCulloch and Pitts (1943). Artificial neurons are the basic processing elements or components of the neural network, they can also be called neurons or nodes. A simplified mathematical model of the neuron specifies that the effect of the associated input signals which are modulated by the connecting weights are represented by the effect of the synapses, and the transfer function represents the nonlinear characteristics exhibited by neurons. The neuron impulse is then computed as the weighted sum of the input signals, transformed by the transfer function. In accordance to the chosen learning algorithm the learning capability of an artificial neuron is achieved by adjusting the weights [4],[5].

The basic architecture defines three types of neuron layers: input layer, hidden layer, and output layers. The signal flows from input to output units, strictly in a feed forward direction, in a feed forward neural network [6]. The data processing can extend over multiple units or layers of units. A neural network is configured in such a way that the set of inputs produces the desired set of outputs of an particular application. Various methods to set the strengths of the connections exist. One way is to set the weights explicitly, using a priori knowledge. Another way is to train the neural network by feeding it with teaching patterns and letting it change its weights according to some learning rule. The learning situations in neural networks may be classified into three distinct sorts. 1) supervised learning, 2) unsupervised learning, and 3) reinforcement learning[6],[10],[11].

In Supervised learning [12], an input vector is presented at the inputs together with a set of desired responses, one for each node, at the output layer. A forward pass is done, and the errors or discrepancies between the desired and actual response for each node in the output layer are found. These are then used to determine weight changes in the net according to the prevailing learning rule. the term supervised originates from the fact that the desired signals on individual output nodes are provided by an external teacher. The examples of this technique are the back propagation algorithm, the delta rule, and the perceptron rule [7],[8].

Whereas in unsupervised learning process, a output unit is trained so that it responds to the clusters of pattern within the input. The system discovers statistically salient features of the input dataset, in such type of learning pattern [12], [13]. Hence the system must build its own representation of input stimuli, as there is no prior classification of input patterns.

In Reinforcement learning [14] the learning process consist of what actions to perform, how to map situations to those actions, so as to maximize a numerical reward signal. The learner must discover which actions deliver the most reward in current as well as in the next situation, he is not told about the selection of actions to perform. Reinforcement learning feature comprises of delayed reward and trial & error search.

VII. Arbitrary Partitioned Data

Here, we have consider arbitrary partitioning of data between two parties. No specific order is defined to divide the data between two parties, it can be either horizontally partitioned or vertically partitioned. A dataset can be defined as a combined result of arbitrary partitions. Specifically if we have a database D, consisting of n rows {DB1,DB2,...DBn}, and each row DBi (i goes from 1 to n) contains m attributes, then in each row, DBA i is the subset of attributes held by A (say j is the number of attributes in the subset DBA DBB, number of attributes in the subset DBB=DBA number of attributes in two subsets can be equal (j=k) but does not have to be equal that is, (j ? = k). It might happen that j=m which means that A completely holds that row or j=0 which means B completely holds that row. Thus arbitrary portioning is a combination of horizontal as well as vertical partitions.

VIII. Back Propagation Neural Network

The back-propagation neural network[7] was developed by Rumelhart as a solution to a problem of training multi-layer perceptrons. The fundamentals advances represented by the Back Propagation Neural Network were the inclusion of a differentiable transfer function at each node of the network weights after each training epoch. Back-propagation algorithm requires target patterns or signals as it is a supervised learning algorithm. Training patterns are obtained from the samples of the types of inputs to be given to the multilayer neural network and their answers are identified by the researcher. Training patterns can be samples of handwritten characters, processed data etc. following the task to be solved. The error is calculated at every iteration and is back propagated through the layers of the ANN to adapt the weights. The weights are adapted such that the error is minimized. Once the error has reached a justified minimum value, the training is stopped, and the neural network is reconfigured in the recall mode to solve the task.

The Network is first initialised by setting up all its weights to be small random numbers, say between -1 to +1. The Back propagation algorithm mainly consist of two steps, initially the feed forwarding step and then the back propagation step. Output is generated by applying the input pattern. And is called as Feed Forwarding step. As all the weights are random, the output we get is totally different from what we expect; this output is called as target. We then calculate the error of each neuron, which is essentially a target. Mathematical algorithms are applied iteratively so that the error gets smaller on every iteration. In other words, the output of each neuron will get closer to its target, this is called back propagation step. The process is repeated again and again until the error is minimal. The figure below describes the back propagation process.

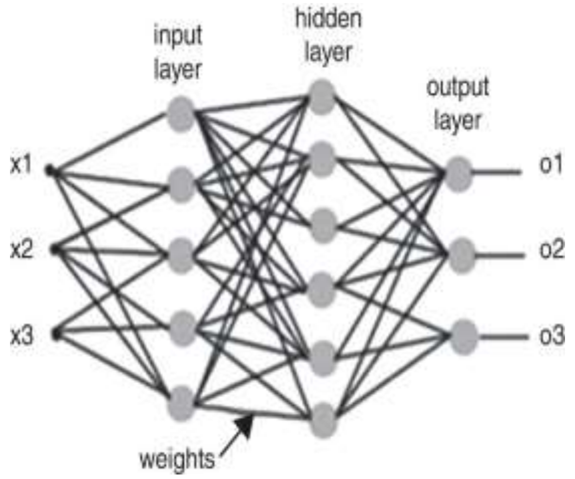

Fig. 1: Back Propagation Neural Network Learning

IX. Back propagation neural network algorithm

```
Algorithm for a 3-layer network with one hidden layer :
initialize network weights (genrally random small values)
do
    forEach training example ex
        prediction = neural-net-output (network,ex) //     forward pass
        actual = teacher-output(ex)
        compute error (predici=tion - actual) at the output units
        for all weights from hidden layer to output layer compute $\Delta\omega h$ // backward pass
        compute $\Delta\omega i$ for all weights from input layer to hidden layer // backward pass ellaborated
        network weights updation // input layer not modified by error estimate
untill all examples are classified correctly or another stopping criterion is satissfied
return the network
```

We can compute the gradient of cost function with the help of the Back Propagation equations. this is how we can derive it in the form of algorithm :

1) input x : Set thecorresponding activation a^1 for the input layer.
2) FeedForward : For each l = 2, 3, . . . , L compute $z^1 = w^1 a^{l-1} + b^l$ and $a^l = \sigma(z^l)$.
3) Output error δ^L: Compute the vector
$$\delta^L = \nabla_a C \odot \sigma'(z^l).$$
4) Backpropagate the error : For eacl l = L - 1, L - 2, . . . , 2 compute
$$\delta^l = ((w^{l+1})^T \delta^{l+1}) \odot \sigma' \ (z^l).$$
5) Output : The gradient of the cost function is given by $\frac{\partial C}{\partial w^l jk} = a_k^{l-1} \delta_j^l$ and $\frac{\partial C}{\partial b_j^l} = \delta_j^l$

Hence the algorithm states that, we compute the error vectors δ^l backward, starting from the final layer. It is unexpected,but we go through the network from backward. The backpropagation proves that the backward movement is a consequence of the fact that the cost is a function of outputs from the network. By applying chain rule iteratively we can understand the variation of cost with the earlier weights and biases [15].

X. Expected outcome

We have implemented the scheme, and the result we got in terms of computational and communicational cost is comparatively low with respect to prior schemes.

In our scheme the party needs to encrypt the data and then send it to the cloud for further computation. The encryption process is thus offline and is one time cost.

There are two factors into consideration with BPNN Learning - the no of parties and secondly the size of data set. Our implementation derived that the increase in number of parties does not affect the learning time, as the learning operations are parallely carried out by the cloud. Whereas the system is slightly affected when the data set size increases. But the growth of data set size will not affect the participants, as they are carried over by the cloud. Thus we can stabilize the processing time by implementing more cloud servers.

References

[1] A.Bansal, T. Chen and S.Zhong, "Privacy Preserving Back-Propagation Neural Network Learning over Arbitrary Partitioned Data", Neural Computing Applications, vol. 20, no.1,pp. 143-150,Feb.2011.

[2] Jiawei Yuan and Shuchenh Yu " Privacy Preserving Back-Propagation Neural Network Learning Made Practical with cloud Computing", IEEE Transactions on parallel and distributed systems, vol. 25, no. 1Januyary 2014.

[3] T.Chen and S.Zhong " Privacy Preserving Back-Propagation Neural Network Learning", IEEE trans. Neural Network, vol. 20, no. 10, pp. 1554-1564,Oct.2009.

[4] R.L. Grossman, "The Case for Cloud Computing", IT Professional, vol. .11, no. 2,pp.23-27, Mar. 2009.

[5] R.L. grossman and Y. Gu, "Data Mining Using High PerformanceData Clouds: Experimental Studies Using Sector and Sphere", Proc. 14th ACM Int'l Conf. Knowledge Discovery and Data Mining (SIGKDD '08), pp. 920-927, 2008.

[6] Anil K. Jain, Jianchang Mao, and K.M. Mohiuddin - IBM Almaden Research Centre, "Artifical Neural Networks: A Tutorial", IEEE 0018-9162/96/55.000, 1996.

[7] Mirza Cilimkovic, "Neural Networks and Back Propagation Algorithm", Institute of Technology Blanchardstown,Dublin,Ireland.

[8] Zhen-Guo Che, Tzu-An Chiang, and Zhen-Hua Che, "Feed-Forward neural network training: A comparison between genetic algorithm and back-propagation learning algorithm", ICIC InternationalISSN 1349-4198, 2011.

[9] Bahram Javidi, "Optical and digital techniques for information security", Springer Science- Buisness Media, 2005.

[10] Luis Guerra,Laura M. McGarry,Victor Robles,Concha Biezla,Pedro Larranaga,Rafael Yuste, "Comparison between supervised and unsupervised classification of

neuronal cell types: A case study", Wiley Online Library,2010

[11] R.Sathya and Annanna Abraham," Comparison of supervised and unsupervised learning algorithms for pattern classification", IJARI,vol. 2, no. 2, 2013

[12] S.B. Kotsiantis, "Supervised Learning : A Review of Classification Techniques" Informatica 31(2007) 249-268 249.

[13] Jennifer G. Dy and Carla E. Brodley, "Feature selection for unsupervised learning", Jornal of Machine Learning 5, 845-889, 2004.

[14] Leslie Pack Kaelbling, Michael L. Littman, and Andrew W. Moore, "Reinforcement Learning : A Survey" Jornal of artificial intelligence Research,237-285, 1996.

[15] Michael A. Nielsen, "Neural Network and Deep Learning" Determination Press, 2015.