RESEARCH ARTICLE OPEN ACCESS

Differential Privacy Preserving of Training Model in Wireless Big Data with Edge Computing

S. Sambasivam MCA., M.Phil., *, D. Guhan**
*Professor, Department of MCA, Nandha Engineering College (Autonomous), Erode, Tamilnadu, India.
Email: sammy2173@gmail.com
** Final MCA, Department of MCA, Nandha Engineering College (Autonomous), Erode, Tamilnadu, India.
Email : guhansingam8@gmail.com

****

ABSTRACT:

Implement a machine learning strategy for smart edges using differential privacy. In existing system focus attention on privacy protection in training datasets in wireless big data scenario and it also adding Laplace mechanisms, and design two different algorithms are Output Perturbation (OPP) and Objective Perturbation (OJP). Privacy Preserving issues presented in the existing literatures for differential privacy in the correlated datasets, and further provided differential privacy preserving methods for correlated datasets, guaranteeing privacy by theoretical deduction. Its processes have been developed to cleanse private information from the samples while keeping their utility and it can approach that can be applied to decision tree learning, without connected loss of accuracy.

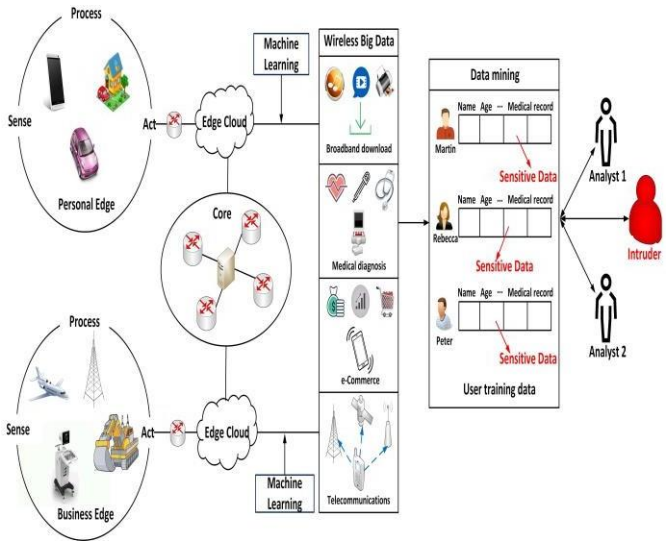
Keywords — **Privacy Preserving, Machine Learning, Perturbation, Wireless Big data, Laplacian Mechanism.**

****

I. INTRODUCTION

CURRENTLY, smart edges have received extensive attention in the data processing, data analysis and data storage in wireless big data scenario [1]. Smart edges bring enormous benefits in the aspect of analysing and mining data, perceiving location information, such as localization and low latency [2]. Some wireless big data, e.g., broadband download, online business, health sensing, etc [3], contains a lot of analysis and mining of effective information. However, it is inevitable to involve some privacy records when the smart edges use machine learning methods for data processing and prediction, such as license plates, tax information, personal assets information [4]. In recent years, it is often seen that hackers have exploited privacy holes in machine learning, and restored private sensitive training data from the model. Fredrik son et al. [5] used the privacy leak of the computer vision classifier to expose the personal picture information from the training data. Thus, the problem of data privacy in machine learning is becoming more and more serious, especially for the privacy protection of training datasets [6]. Once the data with sensitive information is maliciously attacked, it is most likely to be exploited by criminals. To guarantee the privacy protection of the training datasets, this

paper first analyses the possible privacy issues faced by the training datasets for smart edges in



wireless big data scenario. There are a large number of personal edge nodes and business edge nodes in the smart edges, as shown in Fig. 1, and they own a certain ability to compute and process data. These nodes can make use of the edge cloud to make the network cooperate with the terminals, which can achieve the business localization processing, the service delay reduction, and the network efficiency promotion [7]. As the most popular methods of data analysis, machine learning can efficiently analyse and mine valuable information hidden behind the wireless big data. From the figure, we can see that smart edge nodes can analyse a variety of different wireless big data by using machine learning methods [8]. For example, the analysts establish a machine learning model to analyse and predict the results of medical diagnosis, which is more conductive to the prediction and resolution of problems. Analysts can find out the common features of cancer patients by sorting out massive training datasets, thereby providing better help in diagnosing cancer. On the other hand, however, intruders may invade certain sensitive data individuals in datasets to achieve their ulterior motives. That is exactly the great challenge in privacy preserving at present: how to ensure that analysts are not likely to cause sensitive data leakage when analysing data in machine learning model Fortunately, differential privacy algorithm is a promising technology solution that can alleviate this tension [9]. This approach allows analysts to perform benign aggregation analysis while ensuring that personal privacy is effectively protected. Differential privacy makes a great contribution in the aspects of input disturbances, data publishing, output perturbations, etc [10], because this algorithm can add Laplacian noise to make input disturbances and output perturbations, respectively. This method ensures that the training data acquired by attackers is not much difference between what they can achieve from individual data that no one has ever recorded [11]. Since sensitive individual information is almost completely irrelevant to the output of the system, users can be sure that the organization that processes their data cannot violate their privacy. Most existing literatures [11, 12, 13,14] mainly address privacy issues in network data publishing, which are mainly aimed at dealing with different privacy models for different privacy requirements. Some literatures [15, 16, 17] focus on preventing nodes from re-identifying the relevant attribute information revealed by the adversaries. Others [18, 19] consider privacy threats due to edge disclosures, allowing adversaries to learn sensitive relationships among individuals. Nevertheless, little attention has been paid to the study of privacy issues for machine learning training datasets. To address the privacy issues of training data, this paper shows how to enhance the training data privacy assurance by adding Laplace noise [20]. Since each node in the edge network has a certain computational power, we block the data into each node for data processing and training, and then summarize the training datasets for further data

prediction. In order to prevent privacy leaks during the use of training data, we add appropriate noise into the summarized training datasets, and use differential privacy methods to ensure its privacy and security. Moreover, considering that any node

may have privacy problems when dealing with the data, we add noise to the blocked data in advance, and then compute and process it by each edge node, which further strengthening the performance of privacy preserving. In addition, existing researches [21, 22, 23, 24] on differential privacy assume that the sampling of data is independent and identically distributed (IID). In practice, however, most of the records in the datasets are related, and these datasets are defined as correlated datasets [22]. Differential privacy technique is imperfectly effective for privacy preserving on a correlated dataset [23]. Correlated differential privacy has become a crucial problem that needs to be solved. At present, limited efforts have been made in correlated differential privacy. Knifer *et al.* [24] was the fifirst to propose that differential privacy reduces privacy guarantees on the correlated datasets if the correlation between records is not taken into account. Some genetic diseases, for instance, are likely to spread among family members. If a hacker knows that one person suffers from an illness, he is likely to infer the health of the rest of the family. An attacker with relevant information knowledge will have higher access to privacy information. Therefore, how to satisfy the strict differential privacy in the correlated datasets is another challenge to be solved.

II.IMPLEMENTATION AND EXPERIMENT

In this section, we introduce the implementation of the proposed methodology, including hardware and software platforms, dataset description, and experimental setup.

i) Hardware and Software platforms

We make the experiments on an Intel Core i7-6700 with a 3.4GHz CPU and equipped with a NVIDIA Tesla K80 GPU accelerator. We use TensorFlow to program the machine learning code, and TensorFlow provides a good experimental platform for machine learning. It has the following advantages [35]:

1) TensorFlow is a lightweight software that supports the current popular programming language, Python.

2) TensorFlow can use multiple GPU on a single machine, and its workflow is relatively easy.

3) TensorFlow uses symbolic programming models to make programming flexible and efficient.

ii) Dataset Description

The experiments involve four datasets:

- **MNIST** [36]: The MNIST dataset consists of handwritten digital images. We divide the 50,000 sample into the training dataset, and the 10,000 sample as the test set. All digital images are regularized in length and centred into 28 $*$ 28 pixels.

- **SVHN** [37]: The SVHN dataset is a real-world image dataset. We divide the 72,048 digits for training, and the 25,964 digits for testing. The SVHN is achieved by house numbers in Google Street View images.

- **CIFAR-10** [38]: The CIFAR-10 dataset consists of color images classified into 10 classes such as airplane, bird, and deer. We partition 50,000 samples into training examples and the 10,000 samples as the test examples. Each example is a 32 * 32 image.

• **STL-10** [39]: The STL-10 dataset is similar to the

CIFAR-10 dataset, but STL-10 has fewer labelled information than CIFAR-10 in the training dataset. All digital images are 96*96 pixels.

iii) Experimental setup

We have implemented the differentially private OPP and OJP algorithms in TensorFlow. In order to protect data privacy, we need to use OPP Pre-processing function to complete the gradient of the update parameter. Moreover, we use OPP Improvement function to minimize a loss function using differentially private OPP, and OPP Train function, which calls OPP Improvement function for minimizing a loss function differentially privately. In all experiments, the regularization coefficient is fixed to 10*04*, and the setting for mini-batch size is 10. In addition, we change the number of participants in each OPP and OJP scenario between N=10 and 100. Approximately 1% of the entire dataset is randomly selected as the initial training dataset for each participant, i.e., 500 data samples for the CIFAR-10 scenario.

iv) Experiment Results

In this section, the proposed methods are validated on realistic datasets. We compare our OPP and OJP algorithms with two existing algorithms, i.e., SGD and PATE-G, in regard to accuracy, data utility and privacy.

a) Accuracy

To verify the effectiveness of our methods compared with the existing mechanisms SGD and PATE-G, we evaluate the accuracy of OPP and OJP when training a machine learning on the MNIST, SVHN, CIFAR-10, and STL-10 datasets. In machine learning mechanisms, there are many parameters that can affect accuracy. We mainly consider the influence of these two parameters on the accuracy, which are α and ε, respectively. Taking the special status of the participants into account, in addition, we change the number of participants in each OPP and OJP scenario between N=10 and 100. In general, participants can optimize parameter values through calibrated training datasets.

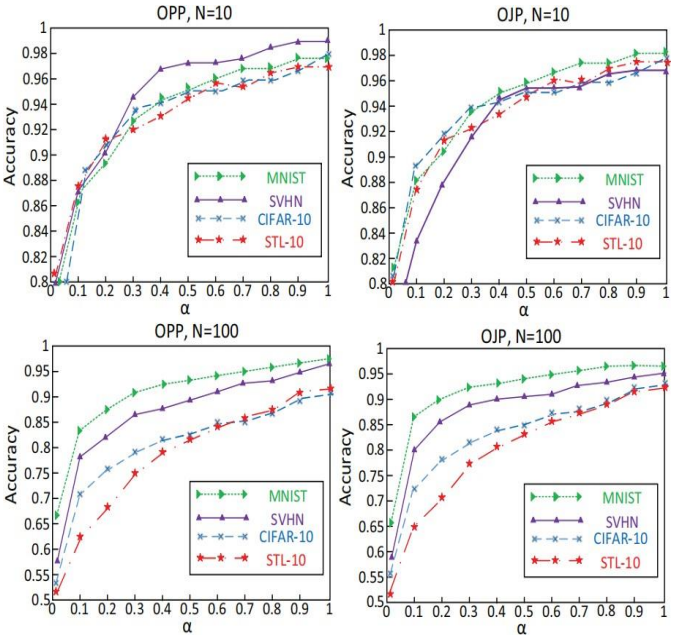


Fig. 2: α-accuracy of OPP and OJP on the four datasets when N=10 and 100.

First, Fig. 2 shows that with the increase of value α, the accuracy of algorithm OJP on four datasets is continuously improved. Moreover, we find that it has a great impact on the accuracy when the value of α is small, with the number of participants N increases. However, when the value of α increases and reaches more than 0.5, the number of participants N has less impact on accuracy. From these four figures, we can obtain that our machine learning-based algorithms OPP and OJP, which can maintain good accuracy on these four datasets, include near 95% accuracy in MNIST and SVHN datasets. Secondly, we further explore the effect of parameter ε on the accuracy. In this experiment, we

compare our algorithms OPP and OJP with the existing SGD and PATE-G on four different datasets. As shown in Fig. 3, our algorithms OPP and OJP embody superior accuracy. When the number of participants is N=10, the accuracy is improved as the value of parameter ε increases. Among them, algorithm OPP can achieve more than 90% accuracy in the MNIST dataset when the value of ε is between 5 and 10.

b) Quality of Privacy Preserving

We also evaluate the quality of privacy preserving. In this subsection, we make experiments to seek out the appropriate value of ε. As shown in Table I, two sets of comparatives experiments are carried out onfour different datasets, respectively. We assume that there are different amounts of queries in different training datasets, i.e., MNIST dataset has 100 and 1000 queries, respectively. First, we set ε = 2.06 and find that the accuracy of OPP and OJP is 97.01% and 96.88% when the number of queries is 100, which is slightly higher than that of SGD with 95.12% and PATE-G with 96.43%, respectively. Then we assume the number of queries is 1000, and find that the value of ε needs increasing to 8.23 to have favourable accuracy, i.e., the accuracy of SGD, PATE-G, OPP and OJP at this time are 94.47%, 93.82%, 95.23% and 94.67%, respectively. We can see that we should increase the privacy budget to ensure the accuracy when the queries are added. In addition, we have done the same experiments on the other three datasets SVHN, CIFAR-10, and STL-10, respectively. From Table I, we further observe that the accuracy of OPP and OJP is superior than that of SGD and PATE-G in most circumstances. With the injection of Laplace noise, machine learning methods can process and train data on different types of datasets. We might be able to think about introducing Laplace mechanisms into different machine learning methods to find suitable methods for processing corresponding datasets.

III) RELATED WORK

In this section, we review the existing literatures on wireless big data, edge computing and differential privacy, respectively.

i) Wireless Big Data

In recent years, wireless big data has drawn many researchers' attention in the aspect of data collection and analytics, network architecture, privacy and security, and its application. Zhang and Qiu [40] proposed a compression aware collection strategy with the purpose of maintaining data quality while minimizing the amount of collected data. In addition, they presented a new idea of dealing with the energy shortage of wireless sensor nodes. AL sheikh *et al.* [41] introduced deep learning into wireless communication, and proposed an extensible learning architecture that supports distributed deep learning. Yang *et al.* [42] proposed a multi-cognitive agent network management architecture, and also presented a Markov game model, which is designed to provide a variety of learning technologies for wireless big data. Another significant area of research for wireless big data is about privacy and security. Hua *et al.* [43] proposed a differential privacy algorithm for generalization of time series trajectory data based on exponential mechanism. Mano *etal.* [44] proposed a steganography algorithm to hide user location, and can preserve the location dataset of user path information. Furtak*et al.* [45] proposed a symmetric encryption technique to protect data in sensor node memory. In addition, much attention has been paid to the application of wireless big data. Pan *et al.* [46] proposed a method for dynamically clustering electric power consumption into wireless big data. Ahmad *et al.* [47] proposed a data collection architecture from the IoT equipment to the social network, which can be used to analyse

big data collection and reflect real-time intelligent city scenario.

ii) Edge Computing

Satyanarayana *et al.* [48, 49] proposed the concept of cloudlet, which can be used as an intermediate layer among the terminal devices, edge cloud platforms and centralized data centers. Bonomi *et al.* [50, 51] proposed that cloud nodes located on the edge of the network can provide new applications and services, especially for wireless big data and Internet of things services. Cisco [52] developed the first commercial fog device, which can be hosted on an operating system running on a virtual machine hypervisor. Hu *et al.* [53] confirmed that edge computing can achieve computationally intensive and highly interactive applications in Wi-Fi networks, and greatly improve latency. Bastug *et al.* [54] proposed an active caching scheme, and the experimental results show that backhaul can be saved as high as 22%. Vallati *et al.* [55] evaluated three different deployment modes for fog nodes connected to LTE networks: macro based, device to device (D2D) based, and traditional deployment. Gazis *et al.* [56] proposed an adaptive operation platform for fog components in industrial networking environments. Bonomi *et al.* [57] proposed that geographical distribution can be used as the fourth dimension of big data characteristics. Ahmed *et al.* [58] proposed

that edge networks can acquire analyse real-time data from ubiquitous installations of sensor devices, thus smart parking and traffic control can actually be done.

iii) Differential Privacy

Many efforts have been made about differential privacy. D-work *et al.* [59] took the lead in proposing differential privacy and implementing the privacy preserving. Alhadidi *et al.* [60] proposed a differential privacy-based bidirectional protocol for publishing partitioned data. Goryczka *et al.* [61] presented a differential privacy concept that constrains the number of common aspects in a distributed anonymous manner. Xiao *et al.* [62] presented a data publishing method with differential privacy, and provided accurate answers for count queries. McSherry and Mironov [63] addressed the differential privacy preserving in collective user behaviour. Recently, differential privacy [64] has often been used to protect privacy of machine learning. Many works have been done on adding differential privacy for shallow machine learning models [25, 26, 36, 37, 59]. Shokri and Shmatikov [25] proposed a differential privacy-based distributed SGD algorithm. Abadi *et al.* [36] further improved the privacy loss of SGD in terms of injecting noise. Jagannathan *et al.* [65] used random forest method to protect privacy. Moreover, limited researches have focused on correlated differential privacy. Kifer *et al.* [23] proposed that the privacy2332-7790 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information. This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TBDATA.2018.2829886, IEEE Transactions on Big Data 10 of published data would be more easily to be compromised if the correlated records were ignored. Cao [21] proposed the establishment of function model to analyze and recognize correlated records. Zhu *et al.* [22] made a complete definition of correlated datasets, and investigated and expounded the privacy challenges faced by correlated datasets. Chen *et al.* [66] used the characteristics of correlated datasets to deal with privacy issues in social networks. Nevertheless, our work differs from these state-of-art in the following

aspects. We focus on the privacy protection of the training datasets in wireless big data scenario. Considering the privacy issues of the correlated datasets presented in the existing work, we prove that differential privacy can achieve excellent and reliable privacy protection for correlated datasets through rigorous mathematical derivation.

IV. CONCLUSIONS

Machine Learning approach with differential privacy for preserving training datasets privacy, and apply this machine learning approach to smart edges in wireless big data scenario. We first design two different algorithms OPP and OJP to satisfy differential privacy by adding Laplacian mechanism. In addition, we consider the privacy issues of correlated datasets, and prove the differential privacy preserving of correlated datasets via theoretical analysis. Last but not least, we establish the experiments on the TensorFlow, and evaluate our methods on four different datasets. We compare our OPP and OJP algorithms with two benchmark protocols, i.e., SGD, PATE-G. The experiment results show that the proposed methods can achieve high quality privacy preserving and accuracy assurance.

REFERENCES

[1] X. Ding, Y. Tian, and Y. Yu, "A real-time big data gathering algorithm based on indoor wireless sensor networks for risk analysis of Indus-trial operations," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 1232-1242, 2016.

[2] L. Kong, D. Zhang, and Z. He, "Embracing big data with compressive sensing: a green approach in industrial wireless networks," *IEEE Communications Magazine*, vol. 54, no. 10, pp. 53-59, 2016.

[3] F. Xu, Y. Lin, and J. Huang, "Big data driven mobile traffic under-standing and forecasting: a time series approach," *IEEE Transactions on Services Computing*, vol. 9, no. 5, pp. 796-805, 2016.

[4] S. H. Zhang, D. D. Yin, and Y. Q. Zhang, "Computing on base station behaviour using erlang measurement and call detail record," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 3, pp. 444- 453, 2015.

[5] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," *In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1322-1333.

[6] O. Denas and J. Taylor, "Deep modeling of gene expression regulation in an erythropoiesis model," *In Representation Learning*, ICML Workshop, 2013.

[7] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "NFV: State of the art, challenges, and implementation in next generation mobile networks(vEPC)," *IEEE Network*, vol. 28, no. 6, pp. 18-26, 2014.

[8] Y. Wang, Q. Chen, and C. Kang,"Clustering of electricity consumption behavior dynamics toward big data applications," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2437-2447, 2016.

[9] R. Sarkar and J. Gao, "Differential forms for target tracking and aggregate queries in distributed networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 377-388, 2010.

[10] Q. Liu, C. C. Tan, J. Wu, and G. Wang, "Towards differential query services in cost-effificient clouds," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 6, pp. 1648-1658, 2014.

[11] N. Li, W. Qardaji, and D. Su, "On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy," *in Proc. of ACM Symposium on Information, Computer and Communications Security*, 2012, pp. 32-33.