APDL: A Practical Privacy-Preserving Deep Learning Model for Smart Devices

Xindi Ma[1](✉), Jianfeng Ma[1], Sheng Gao[2], and Qingsong Yao[1]

[1] School of Cyber Engineering, Xidian University, Xi'an, China
xdma1989@gmail.com

[2] School of Information, Central University of Finance and Economics,
Beijing, China

Abstract. With the development of sensors on smart devices, many applications usually learn an accurate model based on the collected sensors' data to provide new services for users. However, the collection of data from users presents obvious privacy issues. Once the companies gather the data, they will keep it forever and the users from whom the data is collected can neither delete it nor control how it will be used.

In this paper, we design, implement, and evaluate a practical privacy-preserving deep learning model that enables multiple participants to jointly learn an accurate model for a given objective. We introduce a light-weight data sanitized mechanism based on differential privacy to perturb participant's local training data. After that, the service provider will collect all participants' sanitized data to learn a global accurate model. This offers an attractive point: participants preserve the privacy of their respective data while still benefitting from other participants' data. Finally, we theoretically prove that our APDL can achieves the ε-differential privacy and the evaluation results over a real-word dataset demonstrate that our APDL can perturb participant data effectively.

1 Introduction

Over the past years, by virtue of the rapid advances in the development of sensors on smart devices, the applications based on sensors have become an essential and inseparable part of our daily lives. Majority of applications are free, relying on information collected from user's device sensors for targeted service. As the bases, the collected data will be trained to learn an accurate model, which is usually called deep learning. After that, they will use the trained model as a foundation of their new services and applications, including accurate image and speech recognition [1] which surpassing humans [2].

However, the collection of data from users always has a number of privacy concerns for the data contributors. Nowadays, many companies collect photos, video, and speech information from individuals with privacy risks. Once the companies gather the data, they will keep it forever and the users from whom the data is collected can neither delete it nor control how it will be used [3]. What is worse, the collected voice recordings and images always contain many

L. Zhu and S. Zhong (Eds.): MSN 2017, CCIS 747, pp. 377–390, 2018.
https://doi.org/10.1007/978-981-10-8890-2_28

accidentally captured sensitive information, for example, the sound of others speaking, ambient noises, computer screens, people faces, and other sensitive items [4]. After processing the above information, the companies can analyze and obtain users' live environments and social relationships which are also considered as privacy information for users.

In many domains, especially those related to medicine, the privacy and confidentiality worries may prevent hospitals and research centers from share their medical datasets by the law or regulation. As a result, the medical researchers can only perform the deep learning on the datasets which belong to their own institutions. However, it is well known that the deep learning model will be trained more accurately as the training datasets grow bigger and more diverse. Since the training data is simplex, the researchers may obtain worse models which can not be used for other datasets. For example, the training dataset which is owned by a single organization may be homogeneous, the trained model will be overfitted which produce inaccurate results when used on other inputs. In this case, the utility of datasets will be reduced significantly resulted by privacy restriction.

The goal of this paper is to design a privacy-preserving collaborative deep learning model that offers an attractive tradeoff between utility and privacy. To achieve the goal, we propose a practical privacy-preserving deep learning model based on differential privacy, named APDL. In APDL, we introduce differential privacy mechanism to perturb participant local training data and then upload perturbed data to service provider to train a global deep learning model. The main contributions of this paper are summarized as follows:

- We propose a novel privacy-preserving collaborative deep learning model (APDL) which perturb participant data based on differential privacy. The advantages of APDL are that it not only achieves participant data privacy preservation but also enables multiply participants to learn deep learning models on their own inputs collaboratively. As a result, the participant can benefit from other participants who are concurrently learning similar models.
- To protect participant local training data, we introduce the state-of-the-art differential privacy notions. We quantify the participant privacy level by optimizing the utility based on the local training model and then develop a lightweight data sanitized mechanism to preserve the privacy of local training data. In this manner, using the perturbed training data, the service provider can efficiently train a global deep learning model to provide service for all participants without leaking private information of participants.
- We conduct the analysis of APDL in both theory and practice. The results indicate that our APDL achieves ε-differential privacy and can perturb participant data effectively.

The rest of this paper is organized as follows. Section 2 presents some related works. In this Sect. 3, we present some preliminaries and the system overview, followed by the details of APDL in Sect. 4. Section 5 presents the theoretical analysis of privacy. In Sect. 6, we empirically evaluate the performance of our APDL. Finally, we conclude this paper in Sect. 7.

2 Related Work

In the past few years, deep learning has been considered to be a significant application in big data era. However, most of existing studies has faced an enormous challenge, that is how to protect user privacy while training a accurate deep learning model. In this section, we review the current research status of deep learning and privacy preservation in machine learning.

2.1 Deep Learning

Deep learning is researched to train the nonlinear features and functions from big data. The authors in [5,6] has given some surveys for deep-learning architectures, algorithms, and applications. And in some aspects, the deep learning has been shown to outperform traditional techniques, such as image recognition [7], speech recognition [1,8], and face detection [9]. In the domain of medical research, deep learning has been demonstrated its effective for analyzing biomedical data related to genetics [10] and cancer [11,12].

2.2 Privacy in Machine Learning

Privacy has attracted an increasing concern. A number of approaches have been proposed to address identity privacy [13–15], location privacy [16–20] and search privacy [21,22]. Simultaneously, there are many existing works to research the privacy preservation in machine learning. All of them are try to address the following three objectives: privacy of data used for learning a model or as input to an existing model, privacy of the model, and privacy of the model's output.

Addressing the privacy preservation of training data, the authors in [23–27] proposed some models based on encryption scheme. They encrypted the training data with homomorphic encryption and designed some protocols to train the deep learning model. However, these mechanisms usually had the lower efficiency and can not be used as a practical solution. In [28], Abadi et al. developed new algorithmic techniques for deep learning and a refined analysis of privacy costs within the framework of differential privacy. As a directly related work, Shokri and Shmatikov [3] presented a system for privacy-preserving deep learning, allowing local datasets of several participants staying home while the learned model for the neural network over the joint dataset can be obtained by the participants. Phan et al. [29] also proposed a novel mechanism to preserve differential privacy in deep neural networks. They intentionally added more noise into features which are less relevant to the model output, and vice-versa. Yet, most of these works still suffer from the low learning accuracy and efficiency. In comparison, out APDL perfectly protects participants' privacy by utilizing differential privacy while providing a high-quality learning accuracy.

3 System Overview

As discussed above, massive data collection may invoke unexpected privacy issues, which is a key bottleneck for the development and widespread of deep

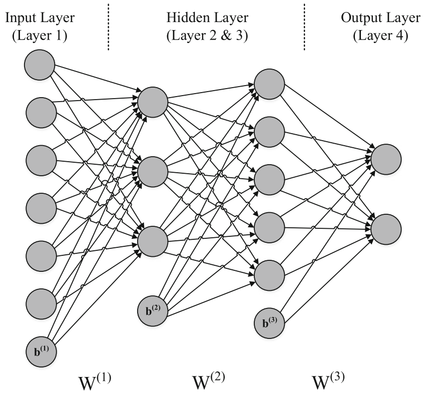

Fig. 1. Neural network mode

learning. To this end, we design APDL based on differential privacy. In this section, we first present some preliminaries that serve as the basis of our APDL, and then present the system model and threat model.

3.1 Preliminaries

Differential Privacy. In our privacy-preserving model, we use the state-of-the-art privacy notion, *Differential Privacy* [30], which can not only provides strong privacy protection but also resist any background knowledge attack from adversaries. Informally, an algorithm A is differentially private if the output is indistinguishable to any particular record in the dataset.

Definition 1 (ε-Differential Privacy [30]). *Let $\varepsilon > 0$ be the privacy budget. A randomized algorithm $\mathcal{A}$ is ε-differentially private if for all data sets D_1 and D_2 differing on at most one element, i.e., $d(D_1, D_2) = 1$, and all $\mathcal{S} \in Range(\mathcal{A})$,*

$$Pr[\mathcal{A}(D_1) \in \mathcal{S}] \leq exp(\varepsilon)Pr[\mathcal{A}(D_2) \in \mathcal{S}] \tag{1}$$

Privacy budget $\varepsilon > 0$ is a small constant, which specifies the desired privacy level. The smaller of ε, the stronger of privacy preservation, leading to more limit on the influence of items. Typically, ε is small (e.g., $\varepsilon \leq 1$).

To achieve the differential privacy, there are two well-established techniques: the Laplace mechanism [31] and the exponential mechanism [32], which are both based on the concept of global sensitivity [31] to compute over a dataset.

Deep Learning. Deep learning can be seen as a set of techniques applied to neural networks. Figure 1 is a neural network with 6 inputs, 2 hide layers, and 2 outputs. The neuron nodes are connected via weight variables. In a typical multi-layer network, each neuron receives the output of the neuron in the previous layer plus a bias signal from a special neuron, such as $b^{(1)}, b^{(2)}$, and $b^{(3)}$. In a deep learning structure of neural network, there can be multiply layers each with thousands of neurons.

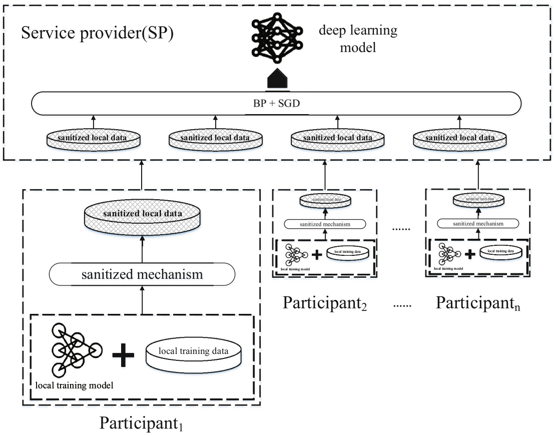

Fig. 2. System model of APDL

Each neuron node (except the bias node) is associated with an activation function f. Examples of f in deep learning are $f(x) = max\{0, x\}$ (rectified linear), $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ (hyperbolic tangent), and $f(x) = (1 + e^{-x})^{-1}$ (sigmoid). The output at layer $l+1$, denoted as $a^{(l+1)}$, is computed as $a^{(l+1)} = f(W^{(l)}a^{(l)} + b^{(l)})$, in which $(W^{(l)}, b^{(l)})$ is the weights connecting layers l and $l+1$, $b^{(l)}$ is the bias term at layer l, and $a^{(l)}$ is the output at layer l. In APDL, we assume that the deep learning model has k layers and the i-th layer owns $n^{(i)}$ nodes.

The learning task is, given a training dataset, to determine these weight variables to minimise a pre-defined cost function such as the cross-entropy or the squared-error cost function [33]. In our model, we consider each participant has trained his own deep learning model using his dataset, expressed as $\overline{Y} = M(X)$, in which X is the input dataset, M is the deep learning model, and $\overline{Y}$ is the computed output of the network.

3.2 System Model

Our APDL is designed to protect participants' data privacy without changing the existed deep learning model. Before describe the details, we derive the basic components in our APDL in Fig. 2.

- **Participants**. In APDL, we consider all participants has the same training objective and each participant has his local training data and local training model. However, his data maybe very homogeneous and training an overfitted model which will be inaccurate when used on other inputs. So we design a service provider (SP) to collect all participants' local training data and train a global deep learning model. Before sharing their local data, participants will sanitize the data using sanitized mechanism (e.g., differential privacy in APDL) to protect the privacy of data owners.

- **Service Provider (SP)**. To train the accurate deep learning model, SP collects the sanitized training data from participants. After that, the global accurate model is trained based on back propagation (BP) and stochastic gradient descent (SGD).

3.3 Threat Model

In APDL, malicious attackers may exist around the participants and steal information during uploading training data. Then, we consider the SP to be curious-but-honest. During the training process, SP may be curious about participants' local data. So SP may be strictly follow the training protocol but also violate and disclose participants' privacy information.

3.4 Design Goals

As a privacy-preserving deep learning model, APDL should fulfill the following requirements.

- **Learning accuracy:** The proposed mechanism should train an accurate deep learning model to suit for all participants' data.
- **Security goals:** The proposed mechanism should keep the privacy of participants' training data. In more details, no sensitive information about the data will be leaked to SP and other participants.

4 Design of Sanitized Mechanism

In this section, we design the sanitized mechanism based on the differential privacy. As described above, one of the most widely used mechanism to achieve ε-differential privacy is Laplace mechanism [31] (Theorem 1), which adds random noises to the numeric output of a query, in which the magnitude of noises follows Laplace distribution with variance $\frac{\Delta f}{\varepsilon}$ where Δf represents the global sensitivity of function f.

Theorem 1 (Laplace Mechanism [31]). *For function $f : \mathcal{D} \rightarrow \mathbb{R}^n$, a randomized algorithm $\mathcal{A}_f = f(D) + Lap(\frac{\Delta f}{\varepsilon})$ is ε-differential private, where $Lap(\frac{\Delta f}{\varepsilon})$ is generated from the Laplace distribution with parameter $\frac{\Delta f}{\varepsilon}$. That is:*

$$Pr[Lap(\frac{\Delta f}{\varepsilon}) = z] \propto exp(-z \cdot \frac{\varepsilon}{\Delta f}) \tag{2}$$

Given two neighboring datasets $\mathcal{D}_1$ and $\mathcal{D}_2$, we present the global sensitivity of function f as follow:

$$\Delta f = \max_{d(D_1,D_2)=1} \|f(D_1) - f(D_2)\|_1 \tag{3}$$

Unfortunately, the naive application of Laplace mechanism results in the significantly large noise magnitude and uselessness of perturbed data because of large global sensitivity. So we adopt a practical differentially private method [34] to sanitize the local training data. In the following, we divide the sanitized mechanism into two phases: *noise calibration*, focuses on selecting the magnitude (denoted as $z_i^{(k)}$) for each output neuron node using local training model; *data sanitization*, aims to generate the useful sanitized training data based on the local training model.

4.1 Noise Calibration for Local Training Model

Based on the pre-defined cost function and local training model, the noise magnitude can be determined by optimizing the cost. In our model, we assume that each local training model has $n^{(1)}$ input neuron nodes and select the average sum-of-squares error between computed output $\overline{Y}$ and true value Y as the cost function. The sanitized noise injected to input neuron nodes is denoted as $Z = (z_1^{(k)}, z_2^{(k)}, \ldots, z_{n^{(k)}}^{(k)})$ in which $z_i^{(k)}$ is the magnitude of Laplace noise for output node i in output layer k. For simplicity, we denote the reciprocal of Z as $Z_r = (1/z_1^{(k)}, 1/z_2^{(k)}, \ldots, 1/z_{n^{(k)}}^{(k)})$. After that, we can determine the magnitude of Laplace noise on each input neuron node via the following programming:

$$\begin{aligned} &\textit{minimize} && \|Z\|_1 \\ &\textit{subjective to} && [M(X) - M(X_p)] \cdot Z_r \leq \varepsilon \\ &&& Z, Z_r \geq 0 \\ &&& d(X, X_p) = 1 \end{aligned} \tag{4}$$

Through the above equations, we can obtain the minimized expected error of all injected noises onto input neuron nodes since the Laplace noises are independent and each of them satisfies $E[|Lap(z_i^{(k)})|] = z_i^{(k)}$. Since the first constrain, we can guarantee the ε-differential privacy for the sanitized local training data. Then, another purpose of the first constrain is to capture the correlation between local training model and neighbouring local data. The noise magnitude Z and Z_r also be ensured non-negative by the second constrain.

Since the above formulation (4) is non-convex, it must be transformed into a convex one to obtain a global optimal solution. For simplicity, we introduce another two variables Z_1 and Z_2 and set $Z_1 = Z, Z_2 = Z_r$. As described above, we can get $z_i^{(k)} \cdot z_{ri}^{(k)} = 1$. Thus, another additional constraint can be added to ensure the reciprocal relationship for each $i \in [1, n^{(k)}]$. Moreover, the constraint can be relaxed to $Z_1 \cdot Z_2^T \geq I$. So we can transform the formulation (4) into the following programming:

$$\begin{aligned} &\textit{minimize} && \|Z_1\|_1 \\ &\textit{subjective to} && [M(X) - M(X_p)] \cdot Z_2 \leq \varepsilon \\ &&& Z_1, Z_2 \geq 0 \\ &&& Z_1 \cdot Z_2^T \geq I \\ &&& d(X, X_p) = 1 \end{aligned} \tag{5}$$

After that, we first solve the convex formulation in programming (5). Then, we set $Z_r = Z_2$ such that our sanitized mechanism also satisfy ε-differential privacy and set Z by letting each item $z_i^{(k)}$ be the reciprocal of the i-th item in Z_r. Since the formulation (5) is convex, we can ensure that our noise calibration algorithm is outperform the traditional Laplace algorithm.

4.2 Adding Noise to Local Training Data

In this section, a noise vector is generated to sanitize the local training data. We take the above noise magnitude output Z as input and generate the Laplace noise to form a useful sanitized local training data. The usefulness of sanitized data is qualified by minimizing the error between local model output based on the sanitized training data and the noisy local model output. Specifically, two error vectors $\{R, L\}$ are also introduced and the utility is qualified by their root mean square error (RMSE): $\frac{1}{2}\|R + L\|_2^2$. Then, the optimization formulation is given as follows:

$$\begin{aligned} &\mathit{minimize} && \frac{1}{2}\|R + L\|_2^2 \\ &\mathit{subjective\ to} && O_z - L \le M(X_p) \le O_z + R \\ & && X_p \in \{0, 1\}^{n^{(1)}} \end{aligned} \tag{6}$$

where O_z is the noise local model output vector and $O_z(i) = v_i^{(k)} + Lap(z_i^{(k)}), i \in [1, n^{(k)}]$, $v_i^{(k)}$ is the deep learning model output of node i in layer k.

However, we can easily find that solving formulation (6) is **NP**-hard by reducing it from Exact Cover problem (The proof is omitted because of the limited space and it is similar to that in [35]). So we replace the X_p with X_p^r, $X_p^r \in [0, 1]^{n^{(1)}}$, to solve the relaxed formulation (6) in our data sanitized algorithm. After that, we can obtain X_p by rounding each item x_{pi}^r to 1 with probability x_{pi}^r.

After the process described above, the participants can generate the sanitized local training data. Then, the participants will upload the sanitized data to SP and SP trains the global deep learning model based the uploaded data. The details of the process are listed in Algorithm 1.

5 Theoretical Analysis

In this section, we theoretically analyze the privacy preservation which APDL satisfies, which is described above that our APDL is ε-differential privacy.

Theorem 2. *Based on the local training data perturbation, APDL satisfies ε-differential privacy.*

Algorithm 1. Process of APDL for Deep Learning Model

```
Input: Local training data $X$, local training model $M$, privacy budget $\varepsilon$.
Output: Sanitized local training data $X_p$.
Solve mathematical formulation (5);
Set $Z_r = Z_2$;
Set $Z$ be the reciprocal of each item in $Z_r$;
Generate noise according to $Lap(z_i^{(k)})$ for each output node $i, i \in [1, n^{(k)}]$;
for each node $i$ in output layer $k$ do
    Set $O_z(i) = v_i^{(k)} + Lap(z_i^{(k)})$;
end for
Relax the constrains in formulation (6);
Replace $X_p$ with $X_p^r \in [0,1]^{n^{(1)}}$ to solve the relaxed (6);
for each node $i$ in input layer do
    Randomly generate a number $\rho$ in $[0,1]$;
    if $\rho \leq x_{pi}^r$ then
        Set $x_{pi} = 1$;
    else
        Set $x_{pi} = 0$;
    end if
end for
Send perturbed local training data $X_p$ to SP;
```

Proof. Since the data sanitization in Sect. 4.2 is considered as post-processing on differentially privacy without the access of local training data, we consider that there is no privacy loss in this phase. Hay et al. [36] had shown that any post-processing of the answers cannot diminish the rigorous privacy guarantee, so we only need to focus on analyzing the privacy guarantee in Sect. 4.1.

Let $M(D_i)$ be the output of local training model with input dataset D_i, $\mathcal{A}$ be the sanitized mechanism, and D_1, D_2 be the neighboring datasets. For any $S = (s_1, s_2, \ldots, s_n{}^{(k)}) \in Range(A)$, the following formulation can be established:

$$\begin{aligned} \frac{Pr[\mathcal{A}(D_1)] = S}{Pr[\mathcal{A}(D_2)] = S} &= \prod_{i=1}^{n^{(k)}} \frac{\Pr[\mathcal{A}(D_1)_i = r_i]}{\Pr[\mathcal{A}(D_2)_i = r_i]} \\ &\geq \exp(-\sum_{i=1}^{n^{(k)}} \frac{1}{z_i^{(k)}} |M(D_1)_i - M(D_2)_i|) \\ &\geq \exp(-\max_{d(D_1,D_2)=1} \sum_{i=1}^{n^{(k)}} \frac{1}{z_i^{(k)}} |M(D_1)_i - M(D_2)_i|) \\ &\geq \exp(-\varepsilon) \end{aligned} \tag{7}$$

The first step is established because of the noises is injected independently on each model output; the second step is obtained from the introduced Laplace noises and triangle inequality, and the last step is derived from the first constraint in formulation (5).

The proof is complete.

Utility Analysis. The expected Mean Absolute Error (MAE) is used to measure the deviation between participant's raw and perturbed training data, which is formally defined as following [35].

$$MAE(X, X_p) = E[\frac{1}{n^{(k)}} \sum_{j=1}^{n^{(k)}} |x_j - x_{pj}|] \tag{8}$$

Let $z_j^{(k)}, l_j, r_j$ be the j_{th} entry in vector Z, R, L.

$$\begin{aligned}
\text{MAE} &= \frac{1}{n^{(k)}} E[\sum_{j=1}^{n^{(k)}} |x_j - x_{pj}|] \\
&\leq \frac{1}{n^{(k)}} (E[\sum_{j=1}^{n^{(k)}} |Lap(z_j^{(k)})|] + E[\sum_{j=1}^{n^{(k)}} |max\{l_j, r_j\}|]) \\
&\leq \frac{1}{n^{(k)}} (\sum_{j=1}^{n^{(k)}} E[|Lap(z_j^{(k)})|] + E[\sum_{j=1}^{n^{(k)}} |l_j + r_j|]) \\
&= \frac{1}{n^{(k)}} (||Z||_1 + E[||R + L||_1])
\end{aligned} \tag{9}$$

6 Experimental Evaluation

In this section, we present a series of empirical results of APDL conducted over MINIST dataset [37] which is composed of 60,000 training handwritten digits and 10,000 test ones. Then, we use Torch7 [38] and Torch7 *nn* packages to construct and train the deep learning model. During the training, we use LeNet neural network as the training model.

While evaluating the local training data perturbation in APDL, we mainly focus on analyzing the influence of participants for model accuracy and the perturbation quality of APDL. We also compare the accuracy of APDL with the non-privacy-preserving scheme (NPP). We assume that there are three participants contributing their local training data and each participant have 20,000 examples. In the evaluation, we use the probabilistic method to measure the learning accuracy: $P = sum\{x = x_p\}/total$, where x is the true value, x_p is the output of the learning model, and $total$ is the number of test examples.

First of all, we carry out the analysis on the training model accuracy influenced by the number of participants. As shown in Fig. 3, the model accuracy increase with the number of epoch. As the increase of epoch, our APDL can train a more accurate deep learning model. So the process of test will be more accurate. Additionally, with more participants joining the training process, the SP has more training data to learn the model. As described above, the deep learning model will be trained more accurately as the training datasets grow bigger and more diverse. So the trained model which has 3 participants has a better model accuracy than that with 1 or 2 participants.

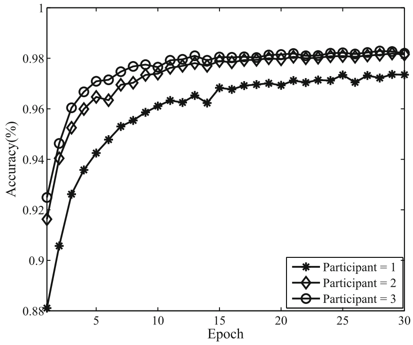

Fig. 3. Model accuracy influenced by participants

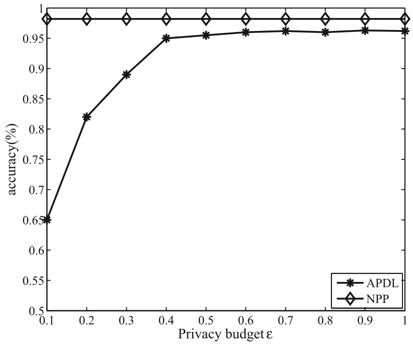

Fig. 4. Model accuracy influenced by privacy budget

Then, we plot the model accuracy by varying the privacy budget ε in differential privacy and compare the accuracy of our APDL with NPP in Fig. 4. As the simulation result shows, the accuracy of our APDL increases with the privacy budget ε. With the increasing of ε, APDL will generate less noise to sanitize the local training data to guarantee the utility. So our APDL will train a more accurate model. However, the accuracy of our APDL has not reached the NPP resulted by the added noise.

7 Conclusion

The disclosure of training data in a union deep learning system seriously threatens participants privacy, especially when participants send their raw data to SP. In this paper, we propose a novel solution, called APDL, to address the privacy issues in deep learning model. In APDL, we introduce differential privacy as the sanitized mechanism to perturb participant's local training data. Our methodology works for any type of neural network. Therefore, it can help bring the benefits of deep learning to domains where data owners are precluded from sharing their data by confidentiality concerns.

Acknowledgment. This work was supported by the National Natural Science Foundation of China (Grant Nos. U1405255, 61672413, 61602537, 61602357, 61303221), National High Technology Research and Development Program (863 Program) (Grant Nos. 2015AA016007), Shaanxi Science & Technology Coordination & Innovation Project (Grant No. 2016TZC-G-6-3), Shaanxi Provincial Natural Science Foundation (Grant Nos. 2015JQ6227, 2016JM6005), China 111 Project (Grant No. B16037), Beijing Municipal Social Science Foundation (Grant No. 16XCC023).

References

1. Hannun, A.Y., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., Ng, A.Y.: Deep speech: scaling up end-to-end speech recognition, CoRR, vol. abs/1412.5567 (2014)
2. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, 7–13 December 2015, pp. 1026–1034 (2015)
3. Shokri, R., Shmatikov, V.: Privacy-preserving deep learning. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, 6–12 October 2015, pp. 1310–1321 (2015)
4. Shultz, D.: When your voice betrays you (2015)
5. Bengio, Y.: Learning deep architectures for AI. Found. Trends Mach. Learn. **2**(1), 1–127 (2009)
6. Deng, L.: A tutorial survey of architectures, algorithms, and applications for deep learning. APSIPA Trans. Sig. Inf. Process. **3**, 1–29 (2014)
7. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Commun. ACM **60**(6), 84–90 (2017)
8. Graves, A., Mohamed, A., Hinton, G.E.: Speech recognition with deep recurrent neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, 26–31 May 2013, pp. 6645–6649 (2013)
9. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: DeepFace: closing the gap to human-level performance in face verification. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, 23–28 June 2014, pp. 1701–1708 (2014)
10. Xiong, H.Y., Alipanahi, B., Lee, L.J., Bretschneider, H., Merico, D., Yuen, R.K., Hua, Y., Gueroussov, S., Najafabadi, H.S., Hughes, T.R., et al.: The human splicing code reveals new insights into the genetic determinants of disease. Science **347**(6218), 1254806 (2015)
11. Fakoor, R., Ladhak, F., Nazi, A., Huber, M.: Using deep learning to enhance cancer diagnosis and classification. In: Proceedings of the International Conference on Machine Learning (2013)
12. Liang, M., Li, Z., Chen, T., Zeng, J.: Integrative data analysis of multi-platform cancer data with a multimodal deep learning approach. IEEE/ACM Trans. Comput. Biol. Bioinform. **12**(4), 928–937 (2015)
13. Jiang, Q., Zeadally, S., Ma, J., He, D.: Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks. IEEE Access **5**, 3376–3392 (2017)

14. Jiang, Q., Ma, J., Yang, C., Ma, X., Shen, J., Chaudhry, S.A.: Efficient end-to-end authentication protocol for wearable health monitoring systems. Comput. Electr. Eng. **63**, 182–195 (2017)
15. Ma, X., Ma, J., Li, H., Jiang, Q., Gao, S.: ARMOR: a trust-based privacy-preserving framework for decentralized friend recommendation in online social networks. Fut. Gener. Comput. Syst. **79**, 82–94 (2018)
16. Gao, S., Ma, J., Sun, C., Li, X.: Balancing trajectory privacy and data utility using a personalized anonymization model. J. Netw. Comput. Appl. **38**, 125–134 (2014)
17. Gao, S., Ma, J., Shi, W., Zhan, G., Sun, C.: TrPF: a trajectory privacy-preserving framework for participatory sensing. IEEE Trans. Inf. Forensics Secur. **8**(6), 874–887 (2013)
18. Ma, X., Li, H., Ma, J., Jiang, Q., Gao, S., Xi, N., Lu, D.: APPLET: a privacy-preserving framework for location-aware recommender system. Sci. Chin. Inf. Sci. **60**(9), 092101 (2017)
19. Ma, X., Ma, J., Li, H., Jiang, Q., Gao, S.: AGENT: an adaptive geo-indistinguishable mechanism for continuous location-based service. Peer-to-Peer Netw. Appl. **11**(3), 473–485 (2017)
20. Gao, S., Ma, X., Zhu, J., Ma, J.: APRS: a privacy-preserving location-aware recommender system based on differentially private histogram. Sci. Chin. Inf. Sci. **60**(11), 119103 (2017)
21. Fu, Z., Huang, F., Sun, X., Vasilakos, A., Yang, C.-N.: Enabling semantic search based on conceptual graphs over encrypted outsourced data. IEEE Trans. Serv. Comput. **12**(8), 1874–1884 (2016)
22. Fu, Z., Wu, X., Guan, C., Sun, X., Ren, K.: Toward efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement. IEEE Trans. Inf. Forensics Secur. **11**(12), 2706–2716 (2016)
23. Zhang, Q., Yang, L.T., Chen, Z.: Privacy preserving deep computation model on cloud for big data feature learning. IEEE Trans. Comput. **65**(5), 1351–1362 (2016)
24. Mohassel, P., Zhang, Y.: SecureML: a system for scalable privacy-preserving machine learning. In: 2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, 22–26 May 2017, pp. 19–38 (2017)
25. Li, P., Li, J., Huang, Z., Li, T., Gao, C., Yiu, S., Chen, K.: Multi-key privacy-preserving deep learning in cloud computing. Future Gener. Compt. Syst. **74**, 76–85 (2017)
26. Gilad-Bachrach, R., Dowlin, N., Laine, K., Lauter, K.E., Naehrig, M., Wernsing, J.: CryptoNets: applying neural networks to encrypted data with high throughput and accuracy. In: Proceedings of the 33rd International Conference on Machine Learning, ICML, New York City, NY, USA, pp. 201–210 (2016)
27. Bost, R., Popa, R.A., Tu, S., Goldwasser, S.: Machine learning classification over encrypted data. In: 22nd Annual Network and Distributed System Security Symposium, NDSS, San Diego, California, USA (2015)
28. Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 308–318. ACM (2016)
29. Phan, N., Wu, X., Hu, H., Dou, D.: Adaptive laplace mechanism: differential privacy preservation in deep learning, CoRR, vol. abs/1709.05750 (2017)
30. Dwork, C.: Differential privacy. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 1–12. Springer, Heidelberg (2006). https://doi.org/10.1007/11787006_1

31. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 265–284. Springer, Heidelberg (2006). https://doi.org/10.1007/11681878_14
32. McSherry, F., Talwar, K.: Mechanism design via differential privacy. In: Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science FOCS, Providence, RI, USA, pp. 94–103 (2007)
33. Murphy, K.P.: Machine Learning - A Probabilistic Perspective. Adaptive Computation and Machine Learning Series. MIT Press, Cambridge (2012)
34. Shen, Y., Jin, H.: EpicRec: towards practical differentially private framework for personalized recommendation. In: Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, pp. 180–191 (2016)
35. Shen, Y., Jin, H.: Privacy-preserving personalized recommendation: an instance-based approach via differential privacy. In: IEEE International Conference on Data Mining, ICDM, Shenzhen, China, pp. 540–549 (2014)
36. Hay, M., Rastogi, V., Miklau, G., Suciu, D.: Boosting the accuracy of differentially private histograms through consistency. PVLDB **3**(1), 1021–1032 (2010)
37. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
38. Collobert, R., Kavukcuoglu, K., Farabet, C.: Torch7: a Matlab-like environment for machine learning. In: BigLearn, NIPS Workshop, no. EPFL-CONF-192376 (2011)