Towards secure deep learning architecture for smart farming-based applications

R. Udendhran1 · M. Balamurugan1

Received: 1 September 2020 / Accepted: 26 October 2020 / Published online: 10 November 2020 © The Author(s) 2020

Abstract
The immense growth of the cloud infrastructure leads to the deployment of several machine learning as a service (MLaaS) in which the training and the development of machine learning models are ultimately performed in the cloud providers’ environment. However, this could also cause potential security threats and privacy risk as the deep learning algorithms need to access generated data collection, which lacks security in nature. This paper predominately focuses on developing a secure deep learning system design with the threat analysis involved within the smart farming technologies as they are acquiring more attention towards the global food supply needs with their intensifying demands. Smart farming is known to be a combination of data-driven technology and agricultural applications that helps in yielding quality food products with the enhancing crop yield. Nowadays, many use cases had been developed by executing smart farming paradigm and promote high impacts on the agricultural lands.

Keywords Deep learning · Smart farming · Differential privacy · Image processing · Feature extraction · Convolutional neural networks · Gradient descent

Introduction
Several agriculture farmers and researchers revamp to smart farming technology for determining soil condition and crops status at real time and also it could be used in sprinkling pesticides with the help of assisted drones, thereby protruding its multi-purposes [1]. On the other hand, the introduction of several communication modules and deep learning algorithms makes the system vulnerable to cyber-security [2] and threats in the smart farming infrastructure. This could lower the economy of a particular country, which predominately relies on the agricultural firm. Domain-specific problems, such as generated data, require privacy frameworks relevant to smart farming. Therefore, the implementation of smart farm technology needs more study prior to widespread community acceptance. For example, if any such information is used by rivals or aggressive actors, leakage of information on the procurement of soil, crops, and agriculture will cause significant economic losses for farmers. Aggregating valuable agricultural information about a single nation on a wider scale is indeed a possible danger. While certain, data protection and privacy seem to be a very critical prerequisite for maintaining effective activity in a smart farming environment and is one of the key objectives.

The differential privacy was introduced in the year 2006, and accepted as a de facto standard in preserving private data. Generally, differential privacy can be inferred with two settings: (1) global—real data are collected by a trusted central authority and then secure them with the privacy-preserving mode; (2) local—no such authority is supported, but the user secures their own data with the private version. Here, the local setting is highly preferred. Several research works had been done with the privacy-preserving under the global setting and how the credible data-holding authority provides sanitary models in preserving such users’ data. Parallel work had also been done with the entrusted data scientists performing analysis on the private datasets with the help of models enabling differentially private query series from the trusted authorities.

This paper focuses on certain users’ who are not ready to share their data with any trusted authorities or the users
send the data to the trusted authorities and are not willing to get them shared by the third parties. This provides a pathway to adapt to local settings. Private multiparty machine learning adapts a model in which multiple data-holding parties would be involved without sharing their data directly. A local classifier will be possessed by each party and this setting had been considered by several authors in preserving private data. With the help of the local differential privacy, local classifiers could be assembled and then combined in a powerful manner from the group of data-holding parties and new features could be enabled based on the users’ case. However, it does not promote significant information in training a classifier locally. The application-specific scenario provides user privacy that could be employed in decision-making level for smart farming infrastructure. As a result, a communication-based privacy mechanism is employed and used in combination with a Trusted Third Party (TTP) architecture to certify users’ privacy.

**Technical background**

Multi-modality is a well-recognized method employed in imaging; it provides multi-information about any plant disease such as bacterial wilt, blight, fire blight, rice bacterial blight, canker, crown gall rot, basal rot, and segmentation based on multi-modality contains a fusion of multi-information which can aggrandize the process of segmentation. After the advent of deep learning, advance state-of-the-art techniques were introduced and enhanced image classification, segmentation, object detection, as well as tracking tasks. Due to the reason that deep learning can self-learning and possess the ability to generalize over large amounts of data, deep learning has been deployed widely in multi-modal data image segmentation [3].

Semantic segmentation is a difficult task when compared with detection and classification tasks. Image classification is used to classify or recognize object class present in an image, while object detection can employ to classify as well as to detect objects present in an image with prior knowledge of position [4, 5], class, as well as the size of each object. Semantic segmentation classifies an object class for every pixel present in an image, which produces a label for every pixel. During classification, the input image is reduced and sent through convolution layers, and then, it enters fully connected known as fully connected layers (FC), and FC produces one predicted label as output for the input image [6, 7]. Fully connected layers can effectively predict dense output based on the arbitrary-sized inputs, inference, as well as learning, which takes place at the dense feed-forward computation and backpropagation. To implement pixel-wise prediction as well as learning in nets [8], in-network upsampling layers with sub-sampled pooling are employed for this purpose [9]. The main advantage is that it provides asymptotically and absolute operations as well as preclude complications in developing network [10]. A simple illustration of a neural network and convolutional neural network is presented in Figs. 1 and 2.

A feasible technique adapted in providing data insights is the data fusion (DF) that combines sensor information and data in the databases based on the knowledge, user mission, and contextual information and the system use the algorithm known as the data fusion [11–13]. Better insights are promoted through the fusion system under progressively varying circumstances [14]. Based on the partial observation, the fusion systems promote real-time simulation globally [15]. On the other hand, the input data obtained from the device or in the form of unstructured data text will be based on the possibilities of combining data into a useful resource. The concept of unstructured data obstructs the fusion of effective data fusion in data processing by smart farming monitoring devices [16]. However, in certain cases, it may also lead to the incomplete data with the explicit and false information may lead the process very difficult to formulate, for example, information of an image from the sensor. Moreover, the lack of communication protocols in the smart farming monitoring equipment between various sensors may also lead to false data due to the lack of mechanism. Additionally, data in the form of text-based and device-based take place at a very high rate, which becomes very difficult for processing that increases cost and also leads to incongruous verdicts [17].

Due to the lack of prior knowledge, data fusion process advancement had been a burden, but this problem could also be resolved through online machine learning with obtained knowledge that is practical with the observed data [18]. Usually, the algorithms of the data fusion are technical and implemented for determining a connection between interesting concepts that are obtained by the sensor data [19, 20]. However, during the algorithm development for the data fusion process, there occurs a trade-off in the complicated data problems [21].

**Proposed methodology**

This paper includes the links that make deep learning imaging and secure user privacy coherent with the applications for smart farming sector. All the important and confidential data are stored in the data centers organized by large companies. Due to the development of advanced algorithms, predictive analytics could be done largely on the user’s private data in the near future with the fastest computation rate. Moreover, privacy of the data could be enhanced by adapting encryption tools, but it also constraints the functionality of data operation [22, 23].
Deep learning neural networks adapts both the encryption and decryption techniques, which had considered in many research fields. Techniques like Hopfield neural networks and chaotic time delayed neural networks. Autoencoder networks have been under profound examination. Especially, generative networks and autoencoder frameworks are stimulating as the proposed networks are derived from the above techniques [24].

On the other hand, conventional algorithms hold their supremacy as the above techniques are not implemented in many applications outside the research study [25]. Due to the high computation time for running several matrix operations involved with the deep learning autoencoders, the technology becomes difficult to be adapted with the real-time applications.

In recent years, many studies adapt and examine cryptanalysis and cryptographic applications in deep learning models. However, the combination of conventional encryption techniques with neural networks is highly adapted followed by enhancing the technique instead of providing full concern on the deep learning methods. Autoencoders deal with the encoding and decoding of data with the help of neural networks. Exemplary results had been obtained by the autoencoder applications that includes speech spectrogram coding, for generation images and for deionising have given useful results.

The encryption process could also be carried out by stacked autoencoders, but they are computationally expensive, and also, they lack in security as the encoder and decoder are trained in tandem [26]. However, the quality of the images will be very high as it promotes lossless compression with the decrypted images. Many autoencoder applications are still unexplored with the cryptographic techniques. Chaotic Hopfield neural network is another deep learning approach,
which utilizes binary sequence during encryption in masking plain text. ‘Analysis of neural cryptography’ is a significant and older approach that is based on mutual learning approach, but it is highly exposed to attacks.

Apart from cryptographic techniques, plug and play, generative adversarial networks (GANs) are the highly adapted generative networks that takes random noise as input, thereby generating photo-realistic images. These network functions on the absence of the discriminator for building generative networks negotiating the image quality.

Differential privacy allows to determine a database’s level of privacy. To determine which is better for protecting the privacy of the consumer, this will allow one to experiment with various methods. By recognizing our degree of data protection, we will then calculate the risk that someone will be able to leak confidential data from the dataset and how much data may be leaked at most. One approach used by differential privacy to protect the privacy of individuals is to apply noise to the results. As the primary process, local differential privacy is chosen and it functions as follows:

- Local differential privacy: each independent data point in the dataset is applied to the noise. For cases where different customers wish to cooperate without exchanging raw data with each other, local differential privacy may increase model efficiency.

Once the dataset had been derived from the processed and cleared raw data, it is necessary to interpret a problem. In this situation, it is necessary to interpret a problem in the name of an entity and a label. The label can be defined as an event or a property, which the consequential model will strive to forecast and the person, thing, or event described by a label or a property, which the consequential model will strive to

Optimization of parameters could be done by the variants of gradient descent (GD), which is followed by many machine learning models. In summary, the algorithm begins with the initialization of assignment \( w \) to the model parameters, and in some cases, initial auxiliary variable a will also be given with this parameter. In the \( i \)th iteration, gradient \( g_i \) is calculated by selecting the subset of training data \( B \) as follows:

\[
g_i = \sum_{s \in B_i} F_i'(w_{i-1,s}),
\]

where \( F \) is the optimized function. Meanwhile, the problem parameters should also be updated with an update function \( \varphi \) as:

\[
(w_i, a_i) = \varphi(w_{i-1}, a_{i-1}, g_i),
\]

where \( w_j \) is the new assignment made to the problem parameters and \( a_i \) new assignment made to the auxiliary variables. The entire process is iterated several times that leads to the final weight vector and it is used as the output for the learning process.

The main aim of differential privacy is to evaluate the function on the dataset in a private manner and to build procedures in a way of promoting privacy to the entire mechanism mathematically. In this technique, the privacy is guaranteed with the parameters in the form of two categories: \( \epsilon, \delta \geq 0 \). A function \( f : X^* \rightarrow Y \) is known to be \( (\epsilon, \delta) \)-DP if every \( S, Y \) and every pair of dataset \( D \) it holds differ from the single record \( D' \in X^* \), such that:

\[
\Pr[f(D) \in S] \leq e^\epsilon \Pr[f(D') \in S] + \delta,
\]

where the probabilities are with respect to the internal randomness of \( f \). According to this definition, only minimum amount of information will be revealed from the individual record dataset during computation. Another way of turning the non-private function \( f \) into private function is by adding noise randomly to the dataset. This requires Lipschitz-smooth function \( f \) provided with significant criteria, that is:

\[
\| f(D) - f(D') \| \leq \Delta \| D - D' \|.
\]

It is necessary to determine a distribution in such a way that the function \( D \rightarrow f(d) + \text{Rand} \) has the necessary privacy when Rand is sampled for distribution.

The primary focus of the proposed work is to streamline the process of converting data from raw structure into a feature matrix. Therefore, it can upload to the deep learning application.

The architecture of application is initiated by the following components: extractor: data cleansing and processing of raw
data is performed and it transforms parsimoniously into a series of timestamped dataset. Learner: it represents a binder for features and each learner provides a feature vector and a label. Assembler: it defines the classification problem for the application and it interacts with learner to provide solutions.

### Experimentation and evaluation

**Evaluating the model in terms of convergence as well as accuracy**

In this study, we used the dataset from the Plant Pathology 2020 [27]. It is important to deploy deep learning-based applications to provide solutions to reduce starvation among increasing human population. Due to this reason, we paid a particular attention to data from prediction by a secure deep learning architecture.

The exploration data analysis in this study was carried out using Jupyter Notebook to iterate our code on high-level scripting language Python and implement the crucial features effectively without the compatibility issue for data visualization. We focused specifically on the following procedure: (1) cleansing and transforming data, (2) preparing model input and target, and (3) dataset ready for analyzing.

The raw data were processed to identify the different types of errors like incompleteness, inconsistency, duplication, and value conflicting and prepare for modeling, as
shown in Fig. 3. Following steps were taken into consideration to cleansing and transforming the data (Fig. 4):

- Inserting default value to blank cells.
- Deleting special char that can lead to malfunction computation.
- Transforming date-time today count from the starting date of data.
- Scaling data to lower values.

To visualize the labels as well as target data, parallel category is used. As shown in Fig. 5, color continuous scale uses “Inferno” and it is observed that a single healthy crop does not suffer from any types of disease and a single unhealthy leaf suffers from multiple disease as evident in the frequency of each combination.

As shown in Fig. 6, the model achieves an accuracy of 95% for training data, but after ten epochs, the model reaches greater level of accuracy for validation data, and this is due to the fact that validation metrics fluctuates after 10–13 epochs and then achieves more than 90% of accuracy.

Validation metric estimation for deep learning-assisted software

The important analysis of the application validation metric estimation and their algorithm basically depends on the typical estimation process. First step of the algorithm is to define the m number of performance levels of the application such as availability with the estimated components. Then, based on the performance levels, the mathematical model structure for the application will be estimated. For instance, the application that deals with the decision support in the smart farming applications.

Next step deals with the important analysis estimation that is done with the logical differential calculus with the multiple-valued logic mathematical tool. Based on the variable changes, the function of the mathematical model that involves with the logical differential calculus will be modified. Hence, the changes of every state of the application should be evaluated to make validation metric-level changes with the direct partial logic derivative.

The definition of the validation metric analysis derived with the change in ith variable with the direct partial logic derivative is given as:

$$\frac{\partial \varphi(j \rightarrow \tilde{j})}{\partial x_i(a \rightarrow \tilde{a})} = \begin{cases} 1, & \text{if } \varphi(ai, x) = j \text{ and } \varphi(\tilde{a}i, x) = \tilde{j}, \end{cases}$$
where \( f(i, x) = f(x_1, \ldots, x_i - 1, i, x_i + 1, \ldots, x_n) \) is value of structure function; \( \tilde{a} \neq a, \tilde{j} \neq j \) and \( a, j, \tilde{a}, \tilde{j} = \{0, \ldots, m - 1\} \).

For the monotone structure, the changes in the function appear from \( a \) to \( \tilde{a} \) and from \( j \) to \( \tilde{j} \) and these changes could be defined as \( a \) to \( \tilde{a} = (a + 1) \) or \( \tilde{a} = (a + 1) \) and from \( j \) to \( \tilde{j} = (j - 1) \) or \( \tilde{j} = (j + 1) \) accordingly. These changes mainly occur due to the validation metric changes that causes gradual jumps. The application boundary states are calculated with the help of direct partial logic derivative for which the change from variable \( a \) to \( \tilde{a} \) and \( j \) to \( \tilde{j} \) in the \( i \)th component state makes certain changes in the application’s performance level from \( j \) to \( \tilde{j} \). All these states correspond to the non-zero state of derivative. The states (first component-level reduction for the boundary states of the application performance) computed with the direct partial logic derivative as defined with the structure function of the application.

Most software metrics involves in the evolution of any entity, institution or process by means of measuring the state when they are considered as a measuring tool. However, detecting the metrics is quite a difficult task, since numerous metrics are normally detected. Stakeholders tend to handle this problem and it is essential to detect the minimum number of metrics with the investigated methods to interpret the data and analyze them.

This issue has been addressed by several software industries. Some of frameworks created by the software industries include personal measurement software (PSM) and the goal question metric (GQM) approach, and the framework is said to be a practical and rigorous approach in software metrics.

**Conclusion**

In this paper, a secure deep learning architecture for smart farming applications is proposed and analyzed. Implementing deep learning algorithms and testing it on smart farming dataset yielded effective results. In the present artificial intelligence (AI) research field, deep neural network had gained much attention due to the advancement of AI-based machine learning algorithms and systematically utilized with the big data stream, promoting excellent result in analyzing data which could be significantly promoted with smart farming technology.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

**References**

1. Sanjeevi P, Siva Kumar B, Prasanna S et al (2020) An ontology enabled internet of things framework in intelligent agriculture for preventing post-harvest losses. Complex Intell Syst. https://doi.org/10.1007/s40747-020-00183-y
2. Feng Y, Wang D, Yin Y et al (2020) An XGBoost-based casualty prediction method for terrorist attacks. Complex Intell Syst. https://doi.org/10.1007/s40747-020-00173-0
3. Liu W, Li F, Jing C et al (2020) Recognition and location of typical automotive parts based on the RGB-D camera. Complex Intell Syst. https://doi.org/10.1007/s40747-020-00182-z
4. Suresh A, Reyana A, Varatharajan R (2018) CEMulti-core architecture for optimization of energy over heterogeneous environment with high performance smart sensor devices. Wirel Pers Commun 103:1239–1252. https://doi.org/10.1007/s11279-018-5504-0
5. Ji Z, Nie LH (2016) Texture image classification with noise-tolerant local binary pattern. J Comput Res Dev 53(5):11281135
6. Lioutas ED, Charatsari C (2020) Smart farming and short food supply chains: Are they compatible?, Land Use Policy 94:104541. https://doi.org/10.1016/j.landusepol.2020.104541
7. Lioutasa ED, Charatsari C (2020) Smart farming and short food supply chains; are they compatible?, Microprocess Microsyst. https://doi.org/10.1016/j.micropro.2020.104541
8. Li S, Yao Y, Hu J, Liu G, Yao X, Hu J (2018) An ensemble stacked convolutional neural network model for environmental event sound recognition. Appl Sci. https://doi.org/10.3390/app8071152
9. Kermany DS et al (2018) Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell 172(5):1122–1131
10. Krauss C, Do XA, Huck N (2017) Deep neural networks, gradient boosted trees, random forests: statistical arbitrage on the S&P 500. Eur J Oper Res 259:689–702
11. Würt T, Hoffmann M, Christlein V, Breininger K, Huang Y, Unberath M et al (2018) Deep learning computed tomography: learning projection-domain weights from image domain in limited angle problems. IEEE Trans Med Imaging 37:1454–1463
12. Yinan Y, Jiujin L, Wenchao Z, Chao L (2016) Target classification and pattern recognition using micro-Doppler radar signatures. In: Seventh ACIS international conference on software engineering, artificial intelligence, networking, and parallel/distributed computing, pp 213–217
13. Diamant I, Bar Y, Geva O, Wolf L, Zimmerman G, Lieberman S et al (2017) Chest radiograph pathology categorization via transfer learning. In: Deep learning for medical image analysis. Elsevier, pp 299–320
14. Hammernik K, Klatzer T, Kohler E, Recht MP, Sodickson DK, Pock T et al (2018) Learning a variational network for reconstruction of accelerated MRI data. Magn Reson Med 79:3055–3071
15. Domingos P (2012) A few useful things to know about machine learning. Commun ACM 55(10):78–87
16. Tokognon B, Gao G, Tian Y (2017) Structural health monitoring framework based on Internet of Things: a survey. IEEE Internet Things J 4(3):619–635
17. Mehta M, Agrawal R, Rissanen J (1996) SLIQ: a fast scalable classifier for data mining. In: Proceedings of the fifth international conference on extending database technology (EDBT), Avignon, France
18. Panchal G, Ganatra A (2012) Optimization of neural network parameter using genetic algorithm. Lambert Academic Publishing, Germany
19. Gutiérrez PA, Martínez C (2012) Hybrid artificial neural networks: models, algorithms and data, vol 6692, pp 177–184
20. Olaronke I, Oluwaseun O (2016) Big data in healthcare: prospects challenges and resolutions. In: Proceedings of future technologies conference (FTC), December 2016, pp 1152–1157
21. Altun Y, Hofmann T, Johnson M (2003) Discriminative learning for label sequences via boosting. In: Becker STS, Obermayer K (eds) Advances in neural information processing systems 15. MIT Press, Cambridge, pp 977–984
22. Suresh A, Kumar R, Varatharajan R (2020) Health care data analysis using evolutionary algorithm. J Supercomput 76:4262–4271. https://doi.org/10.1007/s11227-018-2302-0
23. Altun Y, Hofmann T, Smola A (2004) Gaussian process classification for segmenting and annotating sequences. In: Proceedings of 21st international conference on machine learning (ICML), Banff, Alberta, Canada
24. Suresh A, Udendhran R, Balamurgan M (2020) Hybridized neural network and decision tree based classifier for prognostic decision making in breast cancers. Soft Comput 24:7947–7953. https://doi.org/10.1007/s00500-019-04066-4
25. Geetha R, Sivasubramanian S, Kaliappan M et al (2019) Cervical cancer identification with synthetic minority oversampling technique and PCA analysis using random forest classifier. J Med Syst 43:286. https://doi.org/10.1007/s10916-019-1402-6
26. Arularasan AN, Suresh A, Seerangan K (2019) Identification and classification of best spreader in the domain of interest over the social networks. Cluster Comput 22:4035–4045. https://doi.org/10.1007/s10586-018-2616-y
27. Thapa R, Snively N, Belongie S, Khan A The Plant Pathology 2020 challenge dataset to classify foliar disease of apples. https://arxiv.org/abs/2004.11958

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.