Journal of Scientific & Industrial Research
Vol. 80, July 2021, pp. 570-581

Knitting Machinery Spare Classification using Deep Learning with Differential Privacy

Canan Tastimur[1]*, Songul Kasap[2] and Erhan Akin[3]

[1]Computer Engineering, Erzincan Binali Yildirim University, Erzincan, 24000, Turkey
[2]Department of manufacturing, Nit Orme Textile Co. Ltd., Istanbul, Turkey
[3]Computer Engineering, Firat University, Firat University, Elazig, 23100, Turkey

Received 22 February 2021; revised 25 June 2021; accepted 30 June 2021

Given their widespread use, knitting machines must be maintained regularly. When the spare parts that make up these machines break down or become unusable, they must be replaced with new ones. However, the code/name information of the spare parts is not available to the end user, and can only be accessed with high-cost catalog procurement. Manufacturing companies keep the code/name information of such machine parts confidential. When the literature is examined, there are no studies in which spare parts are classified with machine learning–based algorithms. In line with this, this study focuses on the classification of spare parts using machine learning–based algorithms. The deep learning–based Convolutional Neural Network (CNN) architecture developed in this study can classify highly similar spare parts. In addition, since the code/name information received from the manufacturer and the spare part sample images require confidentiality, the CNN architecture has been developed in combination with the Differential Privacy (DP) method to present the DP-CNN method. As a result of the application of the Differential Privacy method, there has been no great loss of accuracy. This is an important development for our study. In the article, many optimizer algorithms are tested on the proposed method and comparative results are given. A 99.41% accuracy ratio has been obtained with the DP-RMSProp optimization method, which produces the best results. Experimental results of our study are presented in detail.

Keywords: Classification, CNN, Deep learning, Replacement, Spare

Introduction

Knitting machinery building materials can break, crack, wear and so on. In order to obtain these products, people need a product catalog containing knitting machinery parts. However, the cost of the knitting machinery product catalog is quite high. In addition, the actual information on knitting machinery parts is only available from the manufacturer. A person who needs knitting machine part must know the actual code of the product in order to obtain it. In the solution of this problem, lost, broken, worn and so on it is important to determine which product code corresponds to the product that needs to be replaced. Since the price of the manufacturer's knitting machinery product catalog is quite high, it is better to determine which product corresponds to which code by using computer software. Deep learning (DL) has become very common recently in object classification applications. There are many studies on this subject in the literature. However, none involve machine learning–based algorithms for spare part classification. As the correct information about spare parts is available only from the manufacturer, there is a need to classify spare parts using computer vision methods. However, the confidentiality of the spare part information requires the use of the Differential Privacy (DP) algorithm. Differential privacy is a technique used to measure the privacy parameters provided by an algorithm. This study also discusses work in the literature that uses this technique. In this study, spare parts that are substantially similar to each other are classified with a CNN-based approach. Before the spare part classification stage, some image preprocessing steps have been applied to the training data. In this way, this study aims to contribute to classification performance by clarifying the important characteristic points of the data. The training images that passed through the pre-processing process have been trained on the developed CNN network. However, since the confidentiality of the spare part code/name information is important, the training of the CNN network is combined with the DP method. The developed DP-CNN method has been

*Author for Correspondence
E-mail: ctastimur@erzincan.edu.tr

experimentally tested and the experimental results obtained are shown in detail. The contributions of the proposed method are summarized below:

- The literature lacks a machine learning–based approach to spare part classification. Instead, the literature generally adopts the linguistic classification process. In this respect, our study will contribute to the literature.
- A CNN model has been developed using deep neural networks. This model can also be applied adaptively for other datasets.
- To increase the security of the developed model, the accuracy rates have been compared by applying DP-based methods. We apply eight DP-CNN approaches in our study, and that which gives the best result has been determined experimentally.
- Adesuyi & Kim performed on two datasets, and their success rates were 98.1% and 81.5%.(1) Local privacy was tested on the MNIST and CIFAR datasets, achieving success rates of 96% and 91%, respectively.[2] In our study, on the other hand, the accuracy rate is 99.41% on the dataset we created.

Literature Review

Deep Learning-based Object Classification

Human-animal images were taken with a camera, and it was determined whether the object was a human or animal using common background modeling and DL classification techniques. The operations performed in this study were as follows: modeling, cross-frame image path validation, and Deep Convolutional Neural Network (DCNN) Complexity Exactness Analysis (DCEA) for classification.[3] Dolph *et al.* conducted a multiple DL classification of Alzheimer's patients using similar features derived from structural MRI.[4] Two distinct methodologies were introduced. Both models learned subtle differences.[4] Liu *et al.* presented a successful classification of spectral data for DL.[5] They proposed an active learning algorithm based on predominantly cumulative DL for these applications.[5] Anavi *et al.* investigated various approaches to the acquisition of X-ray images and especially for pathology recovery of the breast. Once all data objects have received a total of 443 images, the goal of this research is to sort the images according to similarity.[6] The classification scheme produced by the DL architecture can be seen in Fig. 1.[6]

Xu *et al.* made the classical classification system with a weak recognition rate and a lack of tolerance for noise.[7] Sevakula *et al*. presented a transfer learning classification method for cancer forms. Feature selection and standardization techniques were used.[8] Wood *et al.* suggested a new way for industry classification. They examined the six-digit NAICS codes of their model and the capacity of their model architecture to predict compliance with other industry segmentation schemes.[9] Seth & Biswas used CNN to analyze mailings as images or text, and as a result, this mail content was classified as Spam or not Spam.[10] Jiang *et al*. employed CNN to classify videos.[11] Karahan & Akgül performed eye detection with the DL.[12] Karabulut examined whether the DL approach could achieve successful results for biomedical data. Accordingly, the deep belief network (DBN) and the convolutional neural network (CNN) were used.[13]

Anwer used the convolutional neural network structure from the DL architecture to diagnose breast cancer. Data sets for breast cancer in the Wisconsin UCI Machine Learning Store have also been used to assess the ability of various DL approaches.[14] Elitez used a DL technique to recognize handwriting. In the recognition of a series of figures, the aim is primarily to recognize these figures separately. In the method he developed, a fixed-size filter is shifted over an entire image containing a series of digits and all parts of the filter are taught to the DL network.[15] Kaya applied DL to figure learning.[16] Hatipoglu proposed a method focused on DL for the classification of time series. In

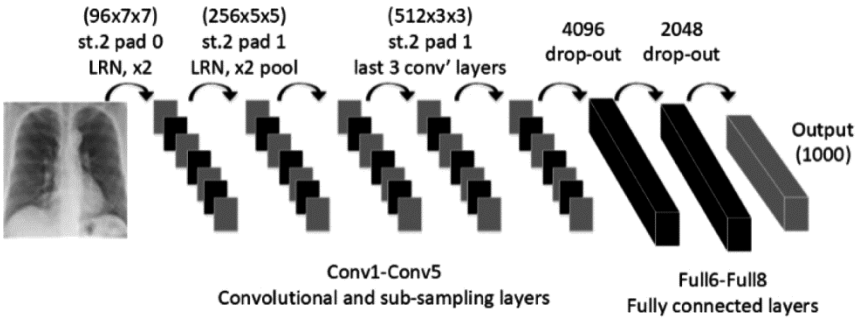


Fig. 1 — Proposed CNN structure[6]

context of this study, both DBN and stacked auto-encoder–based architectures have been established and trained for data obtained from many research areas.[17]

Spare Classification

Teixeira & Figueiredo produced a group classification for a spare part inventory management system, aimed at classifying spare parts as a continuation of the computerized maintenance system for a production business.[18] The goal of the study was to divide spare parts into three criticality categories and assign a criticality level to each spare part: very risky, risky, and risk-free.[18] Molenaers *et al.* offered a part-based proposal for spare part classification. Their proposed model built a single score by combining relevant criteria that affected the importance of a piece.[19] They used logical diagrams in problem-solving.[19] Hu *et al.* noticed that there were many criteria to consider when classifying spare parts: demand, price, criticality, wear, and lead time. They designed if-then rules using the dominance-based rough set procedure (DRSA).[20]

The author evaluated the grades of replacements according to the criteria determined.[21] A key benefit of the established model was the detection of spare parts, which greatly reduced the overall time requirements by eliminating system failures. Another advantage was that the model was simple and could be applied instantly in production conditions without the need for any additional input. Roda *et al.* explained the application of the multi-criteria classification, which has been widely used in spare part classification in the literature.[22] Normalization and discretization were applied to a real dataset with ABC analysis using data mining.[23] Another study classified a Support Vector Machine (SVM)–based spare part risk level.[24] Fuzzy evaluation was done in terms of features such as the likelihood of failure, the importance of a spare part, and availability status. Like[24], Chen & Chen employed a SVM classifier in spare part classification.[25] However, unlike[24], there was a multi-criteria classification model for the ABC classifier. Thus, the key contribution was the interaction between the SVM-ABC multi-criteria classification system. Li & Wei determined the value of the properties of each spare part with a decision tree.[26] In the next step, as in the previous two studies[24,25], SVM was adopted to determine the spare part category.[26] The focus was on standardizing tasks for the evaluation and control of spare parts.[27] In another study, the Analytical Hierarchy Process (AHP) was utilized for classification.[28] This was presented as an application for the inventory tracking of replacement parts. Spare parts used the ABC (A: very important, B: important, and C: weakly important) classifier, as in the studies[23,25,28] When all of these studies were examined by us, a computer vision–based application that could develop a linguistic expression–based classifier was not encountered. These classifier applications were mostly concerned with the inventory management of spare parts. Although expert knowledge was needed to apply the classifiers in these studies, there was no general framework to cover all spare parts.

Differential Privacy

An ϵ-tuple DP approach based on neuron impact factor estimation was proposed without significantly affecting accuracy.[1] Adesuyi & Kimused the Laplace technique based on the privacy parameter ϵ to generate noise to ensure privacy, but this situation reduced accuracy, because the neurons in the model affected the output of the network equally.[1] To prevent this, they used the information factors of different neurons before applying DP.[1] They proposed an ϵ-tuple DP that predicts the impact factor of each neuron and generates a set of privacy budget parameters ϵ for different neurons. A new study redesigned the educational process DL, realizing an algorithm using Local Differential Privacy.[2]

Materials and Methods

This section details the quantitative criteria, as well as the spare parts classification using the CNN architecture developed with DP. In addition, we give optimizer algorithms and fundamentals of the DP technique. Then, we explain the details of the developed model and parameter adjustments.

The Optimizer Algorithm

Machine learning algorithms employ optimization algorithms to minimize error. Those most often used in DL models are Gradient Descent, Adam, Adagrad, and RMSProp. These algorithms enable a neural network to be trained faster and produce predictions more quickly. The loss function must be defined to train the developed neural network. This function is the difference between the value the network receives as a result of the forecast and the actual label values the network is expected to produce. We employ the difference between the actual value and the predicted value to determine functionalities in the error computation. To reach the values with the least error among all values calculated using the lost function,

we use optimizer algorithms. The purpose of this algorithm is to reach the region where the error is the lowest. Mathematically, reaching this lowest region is the basis of the optimizer. The most basic element of this is gradient descent to reach the region where the error draws toward its minimum.

The Differential Privacy Method

This is a method of measuring the privacy parameter provided by an algorithm. With the DP parameter, algorithms can be created that train models on special data. Thanks to DP, the risk of disclosure of sensitive data is reduced. A DP-trained model should not be affected by any training sample or by small training samples in the data set. In this way, the risk of disclosing sensitive training data is reduced.[29] DP expresses the extent to which data in a certain dataset can be disclosed to third parties.[30] The criterion that gives an indication of the privacy coefficient is the cost of privacy. The higher the privacy cost, the smaller the loss of privacy. The use of DP is beneficial in problems involving sensitive training sets.

DL is widely used in many areas, but it has some privacy problems.[30] Applying DP to DL training and classification processes is an effective means of protecting confidentiality.[30] The ability to achieve and record training data presents a beneficial solution to determining whether appropriate data exists in the training dataset through an attacker deducing invasion. The privacy of the inputs of the test or prediction operation, the model's own privacy, the privacy of the training images, and model output's privacy protection are all components of the confidentiality agreement of model learning. The prediction outcome does not vary depending on whether a query is added to or removed from a DP dataset, so it becomes impossible for third parties to examine the dataset. By adding noise during the calculation, the possibility of identifying any sample is reduced. However, this added noise can also result in a decrease in calculation accuracy. Therefore, there is a need to balance accuracy and privacy protection. The level of privacy is measured by the cost of privacy, and this value is inversely proportional to the privacy protection value. That is, the higher the cost, the lower the degree of data protection, and the higher the probability of revealing sample information. The basis of DP is adding random noise to data. This is done by adding noise when calculating model weights to ensure the safety of the model, which is an added benefit of its regularization.

When an attacker observes the outputs in the dataset, query results indicating whether any two data samples are in the dataset should be indistinguishable from each other. For a random function f and any neighbor dataset D_1, the privacy budget (ϵ) of the DP is calculated as follows. D^* is a sterilized dataset.

$$(D_1 \Delta D_2) = 1 \quad \text{... (1)}$$

$$D_2 \; for \; any \; output \; D^* \in range(f) \quad \text{... (2)}$$

$$\Pr\left[f(D_1) \in D^* \leq e^{\varepsilon} \times \Pr\left[f(D_2) \in D^*\right]\right] \quad \text{... (3)}$$

D_1 and D_2 are neighbor datasets differing in only one record. ϵ is the budget for privacy regulation. The lower its value, the more privacy protection it offers. DP ensures that any query result is insensitive. The probability of an attacker guessing whether a single record exists is at most limited to e^{ε}.

The cost of privacy ε can be considered accumulative as more inquiries are made. This cost value continues to accumulate until it reaches a predetermined privacy budget. The response generated by the three data samples means that the DP is obtained if the response produced by two data samples cannot be distinguished by an attacker. Model querying with and without DP is given in Fig. 2.

The Proposed Method

This study presents a method for the multi-classification of substantially similar spare parts for knitting machines. When the images of these parts, which are quite like one another, are examined, it can be noticed that the similarity is quite high as shown in Fig. 3. To prevent this similarity from decreasing the

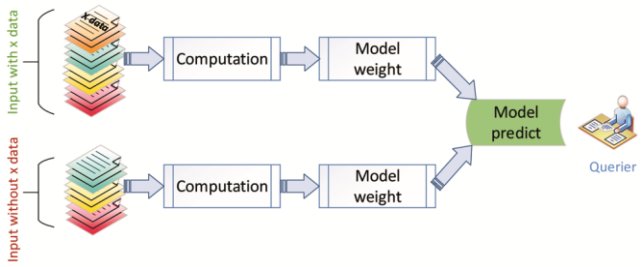


Fig. 2 — Model querying with and without DP[2]

Fig. 3 — The examples of spares used in this study

performance of the classification process, we aim to make the distinguishing points of the spare parts more distinct by applying preprocessing steps to the images in the dataset. Preprocessing steps have been implemented to the images in the dataset and we use the dataset to train the CNN model.

While performing the training process, the DP technique has been applied so that the images and information in the model cannot be obtained by third parties. This stage is important because the manufacturer of the spare parts keeps this information confidential. While applying the DP technique, we also apply many algorithms to reduce error one by one, such as Adam, Adagrad, SDG, and RMSProp, to determine the most successful. The CNN model developed in this study consists of many convolution and pooling layers. Attributes are extracted using the Convolution layer. Detailed information about the parameter settings of the CNN model are given in Table 1. Various optimizer algorithms are utilized throughout the training of the CNN model, the algorithm that gives the best result is determined, and privacy is fitted using DP. Using the DP technique changes the gradients of the optimization algorithms frequently used in normal DL algorithms. Differential privacy offers demonstrably different privacy guarantees for educational model input data. Two changes are made to normal optimization algorithms. First, the sensitivity of each gradient must be limited. It is important to restrict how many gradient calculations will impact each training point collected in the mini batch and the resulting updates to model parameters. DP does this by clipping the gradients calculated at each training point.

This adjustment allows one to decide how much each training point will influence the parameters of the model. Second, we need to randomize the algorithm's behavior to ensure that a certain point is included in the training set and to see how the algorithm will behave by comparing the optimization algorithm parameter updates when it runs with or without this point. This is achieved by sampling random noise and adding it to clipped gradients.

There are three privacy-specific hyper-parameters in DP. The 'l2 norm clip' hyper-parameter is the maximal Euclidean norm of each gradient used to change the model parameters. This hyper-parameter is used to link the optimizer's precision to individual training points. The 'noise multiplier' hyper-parameter is the amount of noise sampled and added to the gradients during training. Although it is not necessary to apply this noise addition process, the more noise is added, the better privacy will be. The 'micro-batch' value of each data group is divided into units. Each micro-batch should contain a single training sample. This allows us to trim gradients by sample, not after they have been averaged on the mini batch. The higher the 'learning rate' hyper-parameter, the more important each update will be. If updates are noisy, a low learning rate helps the educational process converge. This hyper-parameter is already included in normal optimizer algorithms.

In the DP pseudocode algorithms, C is the gradient norm bound, μ is the noise scale, and I is the identity matrix. The hyper-parameter adjustments in the DP algorithms performed in this study are given in Table 2.

Table 1 — Architecture parameters of the suggested CNN

Layer (type)	Output shape
Input image	150, 150
Convolution-1 (32 filters of 3×3 size)	148, 148, 32
Max Pooling-1 (32 filters of 2×2 size)	74, 74, 32
Convolution-2 (64 filters of 3×3 size)	72, 72, 64
Max Pooling-2 (64 filters of 2×2 size)	36, 36, 64
Convolution-3 (64 filters of 3×3 size)	34, 34, 64
Max Pooling-3 (64 filters of 2×2 size)	17, 17, 64
Dropout	0.25
Convolution-4 (128 filters of 3×3 size)	15, 15, 128
Max Pooling-4 (128 filters of 2×2 size)	7, 7, 128
Dropout	0.25
Convolution-5 (256 filters of 3×3 size)	5, 5, 256
Max Pooling-5 (256 filters of 2×2 size)	2, 2, 256
Dropout	0.25
Fully Connected 1	1×262400
Fully Connected 2	1×65792
Fully Connected 3	1028
Softmax	num = 6
Epochs	50
Batch size	128

Table 2 — Hyper-parameter assignment of DP-optimizer algorithms in CNN

Hyper-parameter	Value
Epoch	50
Batch size	128
L2 norm clip (γ)	1.0
Noise multiplier (ϑ)	0.001
Learning rate	0.001
Micro-batch size	1
DP Sum Query	Gaussian ($\gamma, \gamma * \vartheta$)

ALGORITHM 1: DP-SGD ALGORITHM PSEUDOCODE
Input: $x, y, f, \delta, L, \alpha, C, \mu$, I
Output: θ
while δ not met:
$N \leftarrow shuffle\ (x)$ # create random minibacth $Nx^1, x^2, \ldots x^N$ with y^i
$\dot{G} \leftarrow \frac{1}{N}\nabla_\theta \sum_i^N L(f(x^i, \theta), y^i)$ # evaluate the gradient
for $j \in N$ **do**
$\dot{G}_j \leftarrow \dot{G}_j / \max\left(1, \frac{\|\dot{G}_j\|_2}{C}\right)$ # clip gradient in l2_norm
$\dot{G}_j \leftarrow \frac{1}{N}\left(\sum_j^N \dot{G}_j + \mathcal{N}(0, \mu^2, N^2, I)\right)$ # add noise
$$\theta \leftarrow \theta - \alpha.\dot{G}$$
End for
End while

ALGORITHM 2: DP-ADAGRAD ALGORITHM PSEUDOCODE
Input: $x, y, f, \delta, L, \alpha, \varepsilon, I, g_t, C, \mu$
Output: θ
while δ not met:
$N \leftarrow shuffle\ (x)$ # create random minibacth
$Nx^1, x^2, \ldots x^N$ with y^i
$G_t \leftarrow \sum_{j=1}^t g_j \odot g_j$ # sum of the outer product of gradients
$$g_t \leftarrow \frac{1}{N}\sum_{i=1}^N \nabla_\theta L(x^i, y^i, \theta_t)$$
for $l \in N$ **do**
$g_l \leftarrow g_l \Big/ \max\left(1, \frac{\|g_l\|_2}{C}\right)$ # clip gradient in l2_norm
$g_l \leftarrow \frac{1}{N}(\sum_l^N g_l + \mathcal{N}(0, \mu^2, N^2, I))$ # add noise
$$\theta_{t+1} \leftarrow \theta_t - \frac{\alpha}{\sqrt{\varepsilon I + diag(G_t)}} g_t$$
End for
End while

ALGORITHM 3: DP-RMSPROP ALGORITHM PSEUDOCODE
Input: $x, y, f, \delta, L, \alpha, \varepsilon, p, g_t, C, \mu, I$
Output: θ
while δ not met:
$N \leftarrow shuffle\ (x)$ # create random minibacth
$Nx^1, x^2, \ldots x^N$ with y^i
for $j \in N$ **do**
$v_j \leftarrow p\sum_j^N v_t + (1-p)g_t^2$ # locally accumulated squared gradinets
$v_j \leftarrow v_j / \max\left(1, \frac{\|v_j\|_2}{C}\right)$ # clip gradient in l2_norm
$v_j \leftarrow \frac{1}{N}\left(\sum_j^N v_j + \mathcal{N}(0, \mu^2, N^2, I)\right)$ # add noise
$$\theta_{t+1} \leftarrow \theta_t - \frac{\alpha}{\sqrt{\varepsilon + v_t}} g_t$$
End for
End while

ALGORITHM 4: DP-ADAM ALGORITHM PSEUDOCODE
Input: $x, y, f, \delta, L, \alpha, \varepsilon, g_t, m_t, v_t, \beta_1, \beta_2, \mu, C, I$
Output: θ
while δ not met:
$N \leftarrow shuffle\ (x)$ # create random minibacth
$Nx^1, x^2, \ldots x^N$ with y^i
$m_t \leftarrow \beta_1 m_{t-1} + (1-\beta_1)g_t$ # sum of the outer product of gradients
$$v_t \leftarrow \beta_2 v_{t-1} + (1-\beta_2)g_t^2$$
$$m_t' = \frac{m_t}{1-\beta_1^t}$$
$$v_t' = \frac{v_t}{1-\beta_2^t}$$
for $z \in N$ **do**
$\theta_z \leftarrow \theta_z / \max\left(1, \frac{\|\theta_z\|_2}{C}\right)$ # clip gradient in l2_norm
$\theta_z \leftarrow \frac{1}{N}\left(\sum_z^N \theta_z + \mathcal{N}(0, \mu^2, N^2, I)\right)$ # add noise
$$\theta_{z+1} \leftarrow \theta_z - \frac{\alpha}{\sqrt{\varepsilon + v_z'}} m_z'$$
End for
End while

Experimental Results

This study has examined many methods to ensure privacy for the classification of spare parts for knitting machinery. Six classes of spare parts are classified in our study. The hardware features of the computer used in this study are an Intel i7 processor 1.8 GHz CPU, 8GB RAM, and NVIDIA GeForce MX150 GPU. The training process used approximately 200 training samples belonging to each class. Approximately 1200 image data samples have been used in total, and it took 298 seconds for the model to finish the training. Performance criteria obtained from these algorithms are illustrated in Table 3. Performance results of these methods are shown in detail in Table 4. The accuracy, loss, verification accuracy, and verification loss rates obtained by each algorithm as a result of the model training process are also shown in Figs 4–11. Confusion matrices produced by each algorithm are also given (Fig. 12).

Eight different optimization algorithms have been tested to ensure DP privacy. These algorithms are DP-GradientDescent, DP-GradientDescentGaussian, DP-RMSProp, DP-RMSPropGaussian, DP-Adagrad, DP-AdagradGaussian, DP-Adam, DP-AdamGaussian. The DP is described by the values of delta and epsilon. Epsilon is the value of the privacy budget that establishes the set. An epsilon value evaluates how much we cause the output of a dataset in a database to vary from the output of the same model in a

Table 3 — Classification report of different algorithms used in this study

Label	Precision	Recall	F1-Score	Support
DP-GradientDescent				
Nakcc1613	0.20	0.72	0.31	36
Nanac0070a	0.14	0.05	0.08	56
Nanac0148	0.20	0.54	0.29	37
Nanbc0024	0.00	0.00	0.00	44
Nanec0049	0.00	0.00	0.00	41
Nanec0051	1.00	0.05	0.10	40
DP-Adagrad				
Nakcc1613	0.46	0.76	0.57	37
Nanac0070a	0.79	0.24	0.37	45
Nanac0148	0.43	0.29	0.35	51
Nanbc0024	0.38	0.26	0.31	42
Nanec0049	0.46	0.68	0.55	40
Nanec0051	0.43	0.62	0.51	39
DP-RMSProp				
Nakcc1613	1.00	1.00	1.00	40
Nanac0070a	1.00	1.00	1.00	34
Nanac0148	1.00	1.00	1.00	41
Nanbc0024	1.00	1.00	1.00	51
Nanec0049	0.98	1.00	0.99	46
Nanec0051	1.00	0.98	0.99	42
DP-Adam				
Nakcc1613	0.96	1.00	0.98	45
Nanac0070a	1.00	1.00	1.00	50
Nanac0148	0.93	0.97	0.95	40
Nanbc0024	1.00	0.86	0.96	36
Nanec0049	0.98	1.00	0.99	40
Nanec0051	0.98	0.98	0.98	43
DP-GradientDescentGaussian				
Nakcc1613	0.43	0.47	0.45	49
Nanac0070a	0.30	0.37	0.33	46
Nanac0148	0.12	0.05	0.07	43
Nanbc0024	0.29	0.28	0.28	43
Nanec0049	0.22	0.35	0.27	40
Nanec0051	0.16	0.09	0.12	33
DP-AdagradGaussian				
Nakcc1613	0.42	0.93	0.58	44
Nanac0070a	0.51	0.59	0.55	44
Nanac0148	0.36	0.10	0.15	42
Nanbc0024	0.50	0.07	0.12	42
Nanec0049	0.65	0.31	0.42	48
Nanec0051	0.38	0.74	0.50	34
DP-RMSPropGaussian				
Nakcc1613	1.00	1.00	1.00	50
Nanac0070a	0.97	1.00	0.99	34
Nanac0148	1.00	1.00	1.00	43
Nanbc0024	1.00	1.00	1.00	46
Nanec0049	1.00	0.98	0.99	46
Nanec0051	1.00	1.00	1.00	35
DP-AdamGaussian				
Nakcc1613	1.00	1.00	1.00	47
Nanac0070a	1.00	1.00	1.00	47
Nanac0148	1.00	1.00	1.00	44
Nanbc0024	1.00	1.00	1.00	41
Nanec0049	0.97	1.00	0.99	33
Nanec0051	1.00	0.98	0.99	42

Table 4 — Performance metrics of the optimizer algorithms in this study

Algorithm	Accuracy	Loss	Validation Accuracy	Validation Loss
DP-Gradient Descent	0.1795	1.9967	0.2008	1.7690
DP-Adagrad	0.2613	1.7178	0.4567	1.6719
DP-RMSProp	0.9941	0.0414	0.9921	0.0311
DP-Adam	0.9872	0.0821	0.9724	0.3082
DP-Gradient Descent Gaussian	0.1874	1.9414	0.2795	1.7474
DP-Adagrad Gaussian	0.2692	1.7232	0.4488	1.5906
DP-RMSProp Gaussian	0.9862	0.0645	0.9961	0.0108
DP-Adam Gaussian	0.9852	0.0209	1.0000	0.0131

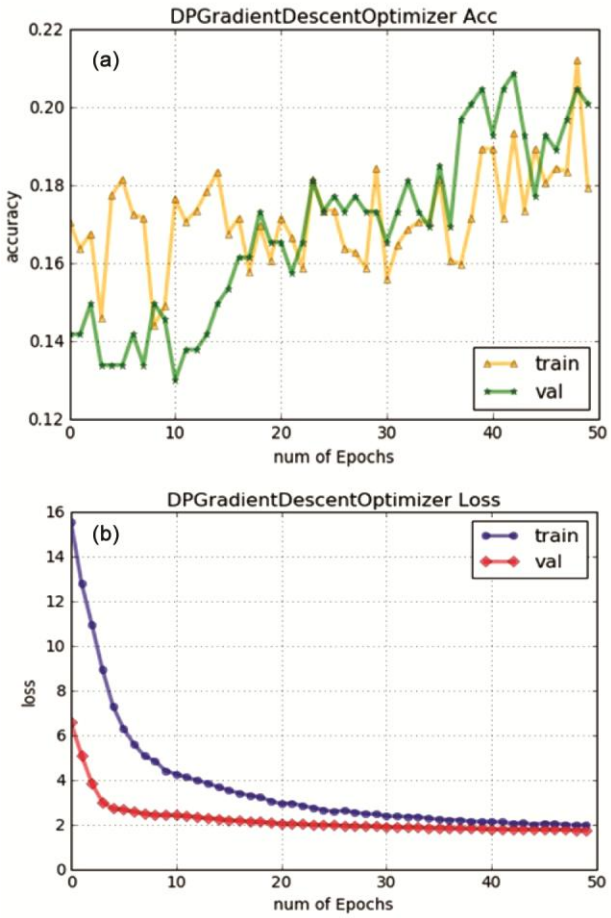


Fig. 4 — DP-GradientDescent's (a) accuracy and (b) loss ratio as a consequence of training and validation

neighboring dataset. This variable shows whether privacy is preserved. The smaller the epsilon, the greater the privacy — delta is the rate at which we ensure privacy. Our cumulative privacy loss delta value increases with each estimation, i.e., query.

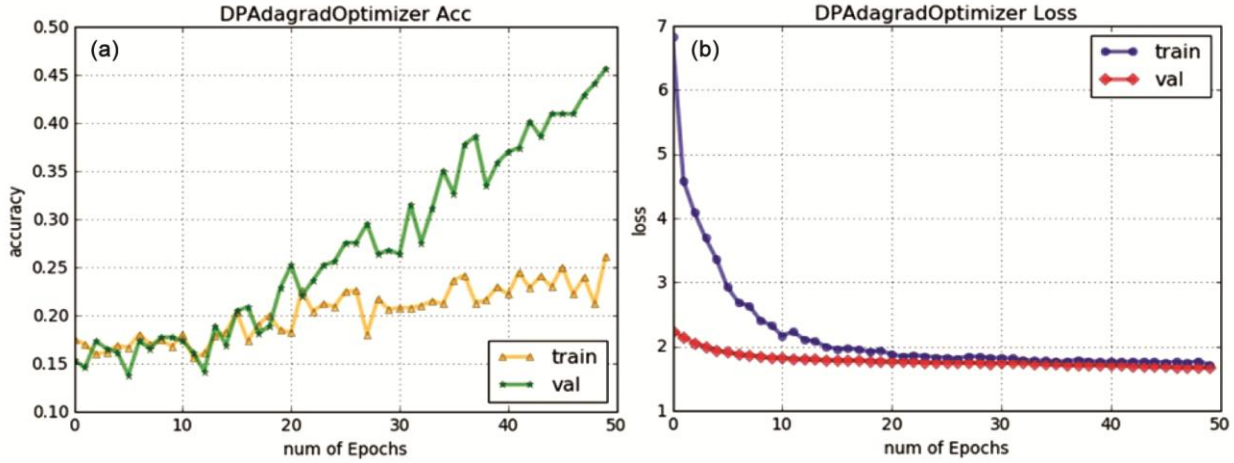


Fig. 5 — DP-Adagrad's (a) accuracy and (b) loss ratio as a consequence of training and validation

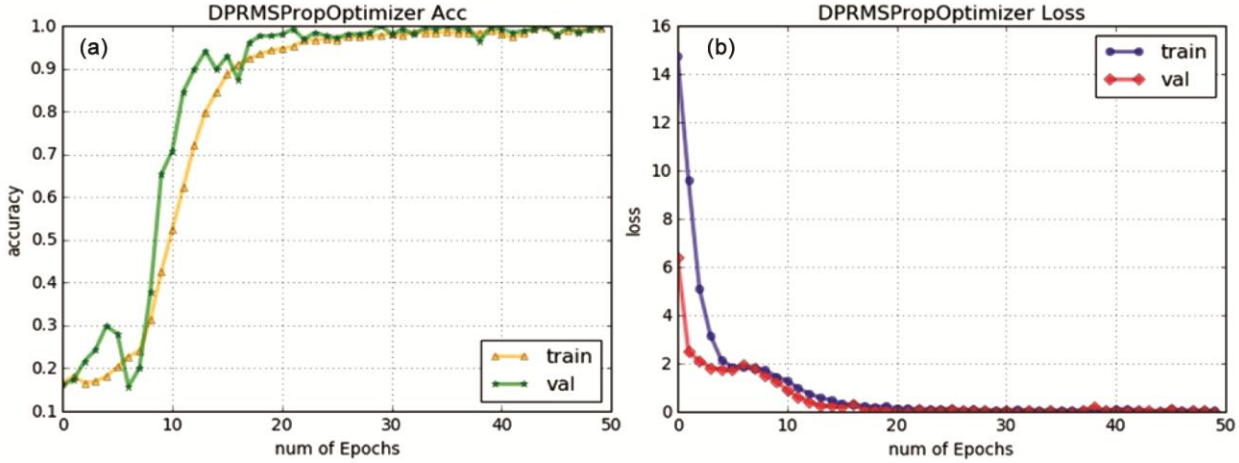


Fig. 6 — DP-RMSProp's (a) accuracy and (b) loss ratio as a consequence of training and validation

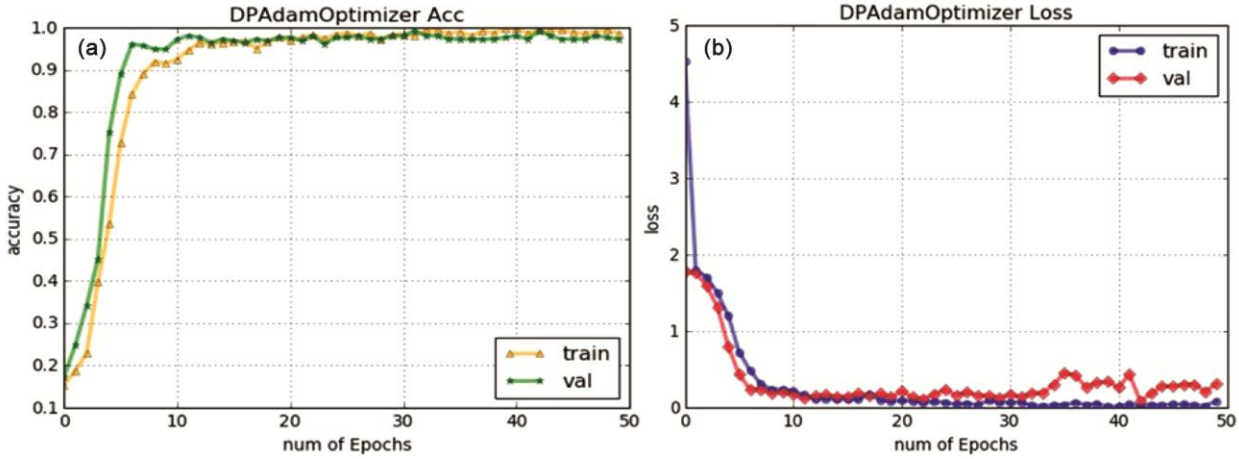


Fig. 7 — DP-Adam's (a) accuracy and (b) loss ratio as a consequence of training and validation

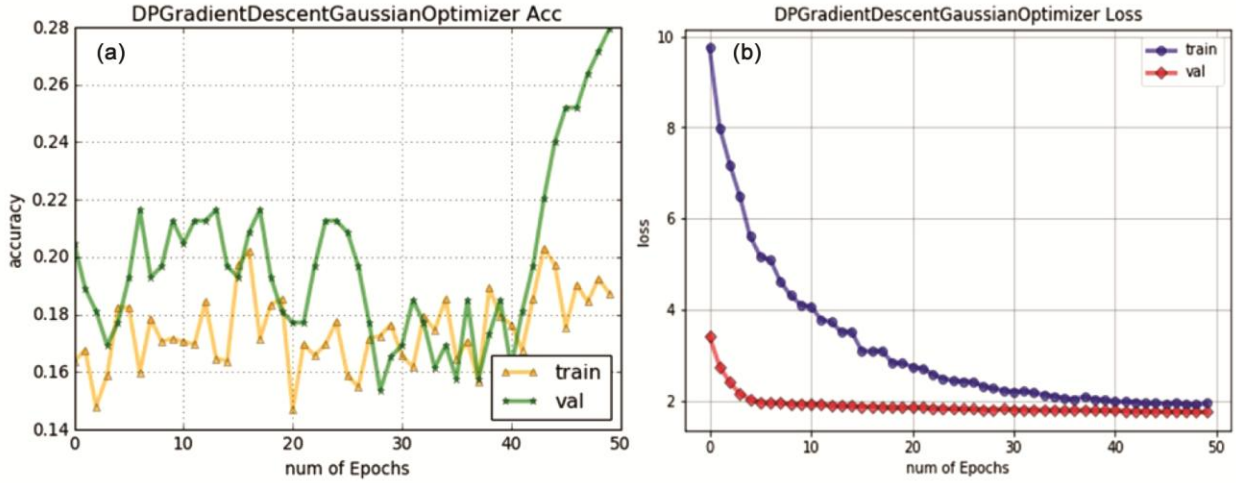


Fig. 8 — DP-GradientDescentGaussian's (a) accuracy and (b) loss ratio as a consequence of training and validation

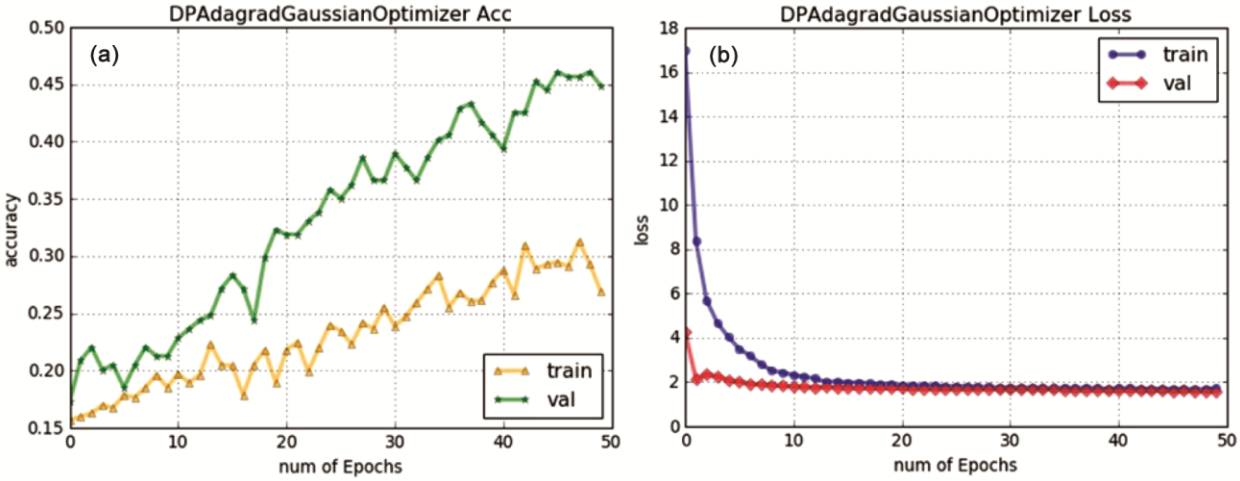


Fig. 9 — DP-AdagradGaussian's (a) accuracy and (b) loss ratio as a consequence of training and validation.

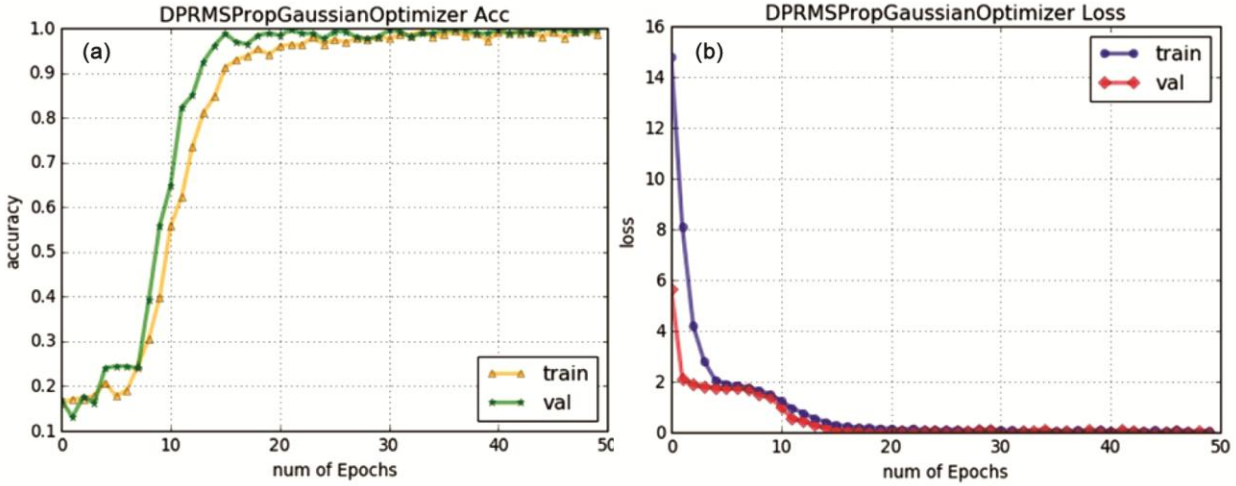


Fig. 10 — DP-RMSPropGaussian's (a) accuracy and (b) loss ratio as a consequence of training and validation

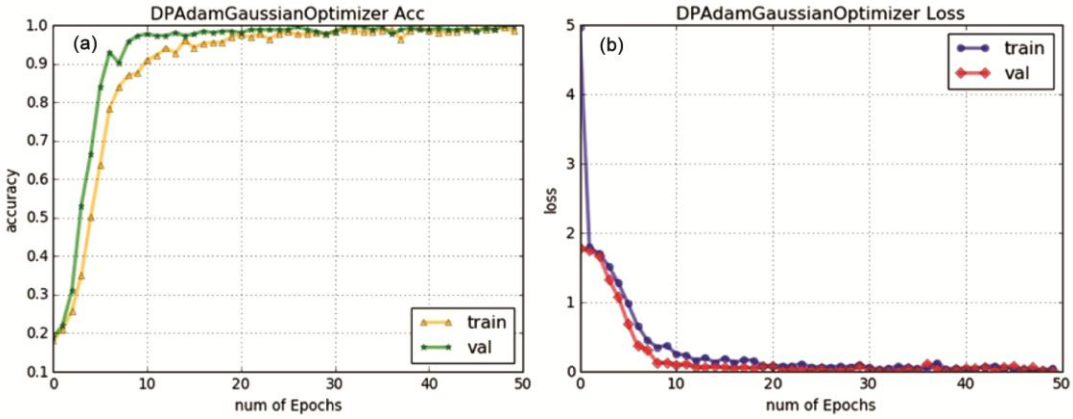
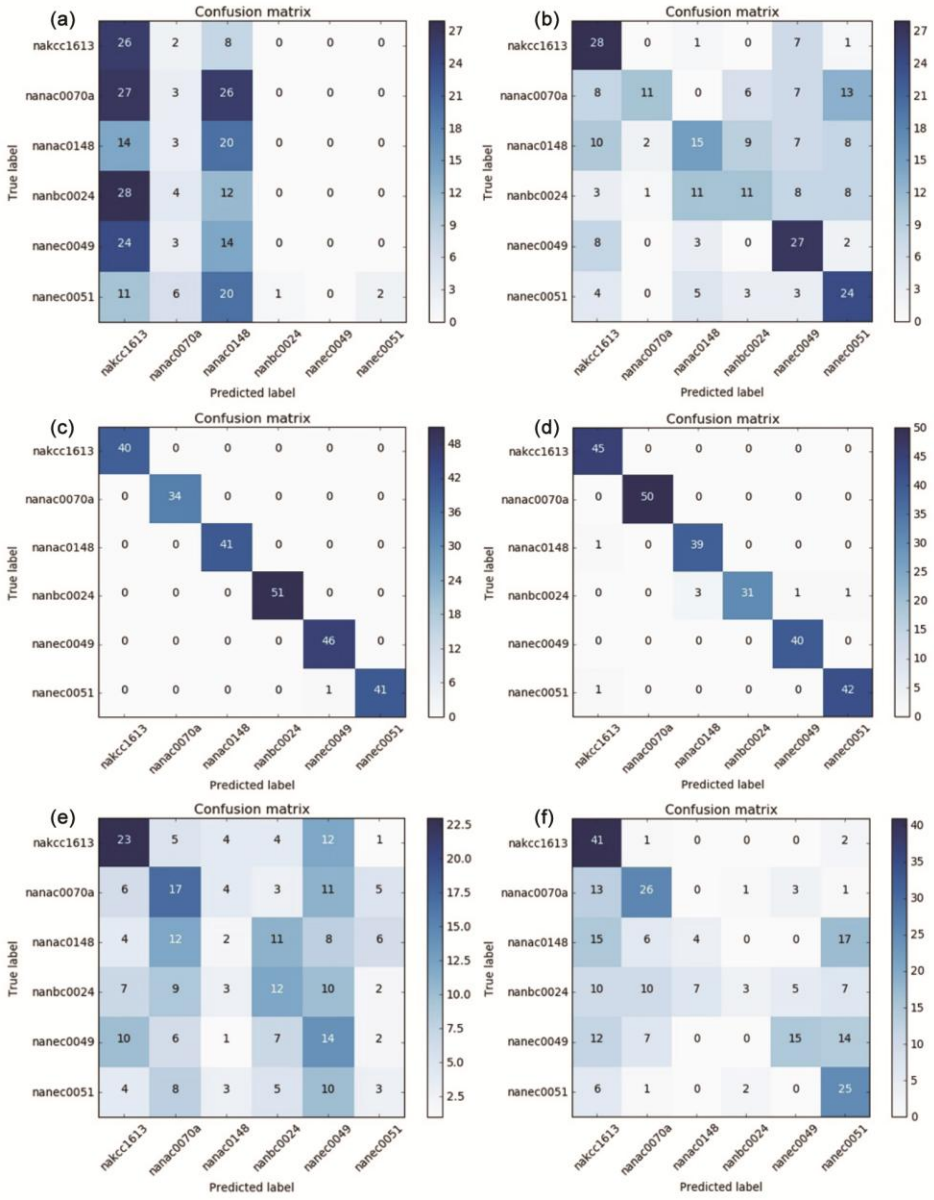


Fig. 11 — DP-AdamGaussian's (a) accuracy and (b) loss ratio as a consequence of training and validation



(*Contd*.)

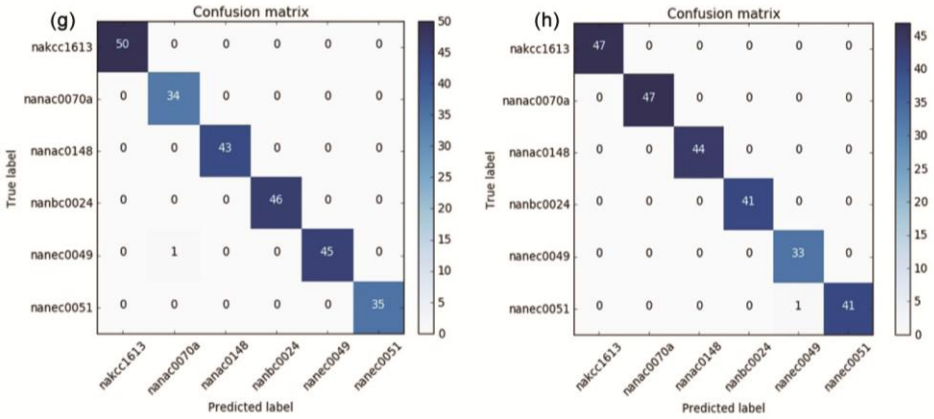


Fig. 12 —Confusion matrixs obtained upon the execution of (a) DP-Gradient Descent, (b) DP-Adagrad, (c) DP-RMSProp, (d) DP-Adam, (e) DP-Gradient Descent Gaussian, (f) DP-Adagrad Gaussian, (g) DP-RMSProp Gausian and (h) DP-Adam Gaussian in the proposed model

The higher the number of predictions, the higher our privacy costs.

Conclusions

When parts of knitting machinery are damaged or need to be replaced, the end user finds it difficult to know the name/code of the spares due to confidentiality and high cost of product catalogue. It has been suggested to use a classifier to find the codes for spare parts; however, ensuring confidentiality since the manufacturer pays attention on the same. This study has developed privacy protection using a CNN-based classifier with DP. The points that distinguish this study from those in the literature are as follows: there is no machine learning–based classifier in the literature for the classification of spare parts. Also, this study achieved the highest accuracy rate when the DL classifier results were compared with those of other methods in the literature. This study showed which DP-based classifier gives the best results in image classification (DP-RMSProp optimization, with 99.41%) by applying many optimization techniques with DP and examining the results in detail. Future studies will examine the effects of hyper-parameters such as epoch and batch size on the DP method.

Acknowledgements

We would like to thank Nit Örme Textile Ind. Trade. Co. Ltd. company for providing us the images in the dataset used in this study.

References

1 Adesuyi T A & Kim B M, Preserving privacy in convolutional neural network: An∈-tuple differential privacy approach, in *IEEE 2nd Int Conf on Knowl Innov and Invent (ICKII)* (Seoul, South Korea) 2019, 570–573.

2 Arachchige P C M, Bertok P, Khalil I, Liu D, Camtepe S & Atiquzzaman M, Local differential privacy for deep learning, *IEEE Internet Things J*, **7(7)** (2020) 5827–5842.

3 Yousif H, Yuan J, Kays R & He Z, Fast human-animal detection from highly cluttered camera-trap images using joint background modeling and deep learning classification, in *IEEE Int Symp Circuits Syst (ISCAS)* (Baltimore, MD, USA) 2017, 1–4.

4 Dolph C V, Alam M, Shboul Z, Samad M D & Iftekharuddin K M, Deep learning of texture and structural features for multiclass Alzheimer's disease classification, in *Int Jt Conf Neural Netw (IJCNN)* (Anchorage, AK, USA) 2017, 2259–2266.

5 Liu P, Zhang H & Eom K B, Active deep learning for classification of hyperspectral images, *IEEE J Sel Top Appl Earth Obs Remote Sens*, **10(2)** (2017) 712–724.

6 Anavi Y, Kogan I, Gelbart E, Geva O & Greenspan H, A comparative study for chest radiograph image retrieval using binary texture and deep learning classification, in *IEEE Eng Med Biol Soc (EMBC)* (Milan, Italy) 2015.

7 Xu Q, Li W, Xu Z & Zheng J, Noisy SAR image classification based on fusion filtering and deep learning, in *3rd IEEE Int Conf Comput Commun Netw (ICCC)* (Chengdu, China) 2017, 1928–1932.

8 Sevakula R K, Singh V, Verma N K & Kumar C, Cui Y, Transfer learning for molecular cancer classification using deep neural networks, *IEEE/ACM Trans Comput Biol Bioinform*, **16(6)** (2019) 2089–2100.

9 Wood S, Muthyala R, Jin Y, Qin Y, Rukadikar N, Rai A & Gao H, Automated industry classification with deep learning, in *IEEE Int Conf on Big Data* (Boston, MA, USA) 2017, 122–129.

10 Seth S & Biswas S, Multimodal spam classification using deep learning techniques, in *13th Int Conf Signal Image Technol Internet-Based Syst (SITIS)* (Jaipur, India) 2017, 346–349.

11 JiangY G, Wu Z, Tang J, Li Z, Xue X & Chang S F, Modeling multimodal clues in a hybrid deep learning framework for video classification, *IEEE Trans Multimedia*, **20(11)** (2018) 3137–3147.

12 Karahan S & Akgül Y S, Eye detection by using deep learning, in *24th Signal Process Commun Appl Conf (SIU)* (Zonguldak, Turkey) 2016, 2145–2148.

13 Karabulut E M, *Investigation of Deep Learning Approaches for Biomedical Data Classification*, Ph D Thesis, Dept Elect Eng, Cukurova University, Adana, Turkey, 2016.

14 Anwer A M O, Breast *Canser Diagnosis using Deep Learning Methods*, Master Thesis, Dept Elect Compt Eng, Turkish Aeronautical Association University, Ankara, Turkey, 2017.

15 Elitez O, *Handwritten Digit String Segmentation and Recognition Using Deep Learning*, Master thesis, Dept Elect Eng, Middle East Technical University, Ankara, Turkey, 2015.

16 Kaya O, *Number Teach with Deep Learning*, Master thesis, Dept Comp Eng, Beykent University, Istanbul, Turkey, 2017.

17 Hatipoglu P U, *Time Series Classification Using Deep Learning*, Master thesis, Dept Indst Eng, Middle East Technical University, Ankara, Turkey, 2016.

18 Teixeira C, Lopes I & Figueiredo M, Multi-criteria classification for spare parts management: a case study, *Procedia Manuf*, **11** (2017) 1560–1567.

19 Molenaers A, Baets H, Pintelon L & Waeyenbergh G, Criticality classification of spare parts: A case study, *Int J Prod Econ*, **140(2)** (2012) 570–578.

20 Hu Q, Chakhar S, Siraj S & Labib A, Spare parts classification in industrial manufacturing using the dominance-based rough set approach, *Eur J Oper Res*, **262(3)** (2017) 1136–1163.

21 Antosz K & Ratnayake R C, Classification of spare parts as the element of a proper realization of the machine maintenance process and logistics-case study, *IFAC-PapersOnLine*, **49(12)** (2016) 1389–1393.

22 Roda I, Macchi M, Fumagalli L & Viveros P, On the classification of spare parts with a multi-criteria perspective, *IFAC Proc Vol*, **45(13)** (2012) 19–24.

23 Prachuabsupakij W, ABC Classification in spare parts for inventory management using ensemble techniques, in *IEEE Asia Pac Conf Circuits Syst (APCCAS)* (Bangkok, Thailand) 2019, 333–336.

24 Su X Y, Zhou X L & Mo Y, Forecast of spare parts inventory risk level based on support vector machine, in *IEEE 17Th Int Conf Ind Eng Eng Manag* (Xiamen, China) 2010, 1344–1346.

25 Chen J & Chen T, Research on classification method of spare parts inventory based on warranty data, In *IEEE Int Conf Serv Oper Logist Inform (SOLI)* (Beijing, China) 2016, 195–199.

26 Li W L & Wei X C, Research on the classification of spare parts for supplier management, In *Int Conf Manag Sci Eng* (Helsinki, Finland) 2014, 379–386.

27 Ratnayake R C, Consequence classification based spare parts evaluation and control in the petroleum industry, in *IEEE Int Conf Ind Eng Eng Manag (IEEM)* (Macao, China) 2019, 1204–1210.

28 Jingjiang G & Zhendong H, A classification model for inventory management of spare parts and its application, in *Int Conf on Ind Control and Electron Eng* (Xi'an, China) 2012, 592–595.

29 https://colab.research.google.com/github/anirudh161/privacy/blob/add-dpsgd-keras-tutorial/tutorials/Classification_Privacy.ipynb#scrollTo=00fQV7e0Unz3 (27 June 2021)

30 Papernot N, Mc Daniel P, Sinha A & Wellman M, Towards the science of security and privacy in machine learning, in *Proc 3rd IEEE Eur Symp Secur Priv (Euro S & P) (London, United Kingdom)* 2016, 1–19.