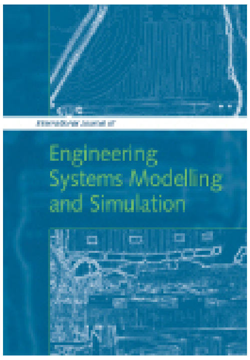


International Journal of Engineering Systems Modelling and Simulation

ISSN online: 1755-9766 - ISSN print: 1755-9758
https://www.inderscience.com/ijesms

An incremental learning on cloud computed decentralised IoT devices

Satish S. Salunkhe, Aditya Tandon, M. Arun, Nazeer Shaik, Supriya Nandikolla, D. Ramkumar, S. Lakshmi Narayanan

DOI: 10.1504/IJESMS.2022.10044923

Article History:

Received:	29 June 2021
Accepted:	16 August 2021
Published online:	03 December 2022

An incremental learning on cloud computed decentralised IoT devices

Satish S. Salunkhe*

Computer Engineering Department,
Terna Engineering College, India
Email: ssalunkhesatish@gmail.com
*Corresponding author

Aditya Tandon

Department of CS&E,
Krishna Engineering College,
Ghaziabad, India
Email: askaditya@ieee.org

M. Arun

School of Computing,
Kalasalingam Academy of Research and Education,
Krishnankoil, Tamil Nadu, India
Email: m.arunvincent@gmail.com

Nazeer Shaik

Department of CSE,
Bapatla Engineering College,
Bapatla, 522102 Guntur District, AP, India
Email: nazeer.shaik@becbapatla.ac.in

Supriya Nandikolla

Department of CSE,
Malla Reddy Engineering College (A),
Hyderabad, Telangana, India
Email: nsupriyacse@gmail.com

D. Ramkumar

Department of ECE,
As-Salam College of Engineering and Technology,
Thirumangalakudi, Aduthurai, Tamil Nadu, India
Email: rkramvictor@gmail.com

S. Lakshmi Narayanan

Gojan School of Business and Technology,
Chennai, Tamil Nadu 600052, India
Email: s.lakshminarayanan@gojaneducation.com

Abstract: It is essential that IoT devices can constantly gather new ideas from streams of data independent of catastrophic forgetfulness. Although merely repeating all prior training samples can solve catastrophic forgetting issues, this method faces privacy problems, memory resources, as well as requires a lot of computational, making it unsuitable for limited-resources IoT devices. In this study, the proposed incremental learning for cloud computed decentralised IoT devices are developed and comprises of constant upgraded information and task resolution model. A neural network is trained and utilised to overcome this problem despite frequent disconnectivity or resource outages without losing a lot of progress using cloud computing. Several research experts have frequent disconnectivity issues regarding cloud computing frameworks because of the platform's free membership. Identical difficulties can be seen when working on a localised computer, where the machine will run out of resources or power at times, forcing the researchers to retrain the systems.

Keywords: incremental learning technique; cloud computing; decentralised IoT devices; internet of things; IoT.

Reference to this paper should be made as follows: Salunkhe, S.S., Tandon, A., Arun, M., Shaik, N., Nandikolla, S., Ramkumar, D. and Narayanan, S.L. (2023) 'An incremental learning on cloud computed decentralised IoT devices', *Int. J. Engineering Systems Modelling and Simulation*, Vol. 14, No. 1, pp.1–7.

Biographical notes: Satish S. Salunkhe holds a PhD in Computer Science and Engineering. Currently, he is working as a Professor in the Department of Computer Engineering, Terna Engineering College, Navi Mumbai. He has overall 16 years of experience in teaching, research and administration. His research areas include AI in fuzzy logic, data mining, machine learning and data warehousing. He has written six journal articles and four conference proceedings published in reputed Scopus and Web of Science listed journals.

Aditya Tandon received his BTech degree in Computer Science Engineering from A.I.E.T. Jaipur (Rajasthan), India and MTech degree in Computer Science and Engineering from Amity University, Noida UP, India. He has teaching experience of 6+ years. Currently, he is working as an Assistant Professor in Department of CS&E, Krishna Engineering College, Ghaziabad (affiliated by AKTU). He is also a PhD research scholar at Quantum University, Roorkee. His research interests are in the field of internet of things, artificial intelligence, wireless sensor networks and network security. So far, he has published 18 papers in reputed conferences and journals and two books – '*Introduction to Artificial Intelligence using Python*' and '*Introduction to Machine Learning*' so far.

M. Arun is an Assistant Professor in the Department of Computer Applications, Kalasalingam Academy of Research and Education and has more than 14 years of teaching experience. He completed his undergraduate degree in Computer Applications at Madurai Kamaraj University, postgraduate degree in Computer Applications at Anna University and Doctoral degree in Kalasalingam Academy of Research and Education. His major research area is network security.

Nazeer Shaik, obtained his PhD in Computer Science and Engineering, awarded in 2017, Acharya Nagarjuna University, Nambur, Guntur. His area of research is network security. He has a total of 19 years of teaching experience and two years of industry experience at Linyi Top Network Co. Ltd., Linyi City, China. Currently, he is working as a Professor and Head of the Department of CSE, Bapatla Engineering College (Autonomous), Bapatla, A.P., India. He is working in current Institution since 2007. In this tenure, he filed three patents at Indian Patent office, Chennai and as a co-inventor granted two patents under 'utility patent' category from State intellectual Property Office (SIPO), China. He published 23 papers in international peer reviewed journals and 18 conference papers presented. As a resource person he delivered guest lectures in various colleges on NAAC/NBA and Autonomous accreditation process and Outcome Based Education workshops.

Supriya Nandikolla is working as an Associate Professor in the Department of Computer Science and Engineering, Malla Reddy Engineering College (Autonomous), Secunderabad. She obtained her MTech (CSE) and PhD (CSE) from ANU, Guntur. She published several papers in national and international journals. She is active member of various professional bodies. Her current research is focused on software reliability, predictive analytics and cloud computing.

D. Ramkumar received his Bachelors degree in Electronics and Communication Engineering from The Indian Engineering College (Affiliated by Manonmaniam Sundaranar University) in the year 1997 and the Masters degree in Applied Electronics with first class with distinction from Anna University, Chennai in year 2007. He received his Doctor of Philosophy (PhD) in the faculty of Information and Communication Engineering in the year 2015 from Anna University, Chennai. Currently he is working as Principal in As Salam College of Engineering and Technology, Thanjavur District from 27 June 2018 to till date. He has 17 more years of teaching experience and published more than ten research papers in reputed international and national journals. He is a life member of ISTE and IEEE. His fields of interest are image processing, biomedical, network and communication.

S. Lakshmi Narayanan is currently working as an Associate Professor at the Gojan School of Buisness and Technology, Chennai, Tamil Nadu, India. He has more than 17 years of teaching experience in reputed institutes. His interest areas are data warehousing, signal and image processing and low power VLSI. He has 20 international/national research papers to his credit.

1 Introduction

The adoption of internet of things (IoT) is an unsustainable rise as information networks grow and many devices are utilised to realise the IoT vision. When the connectivity among devices is not effective then the IoT devices ought to be able to study, think, and understand the social and physical words (Wu et al., 2014). The cognitive needs for IoT have increased significantly in subsequent years and equipping IoT includes learning capabilities is the future ahead. Because IoT devices have been frequently employed in uncontrolled and extremely dynamic situations, accessing training data of every class at a similar time is hard. Thus, while training data for decentralised IoT devices is continually accessible, the devices must be capable of learning about classes progressively.

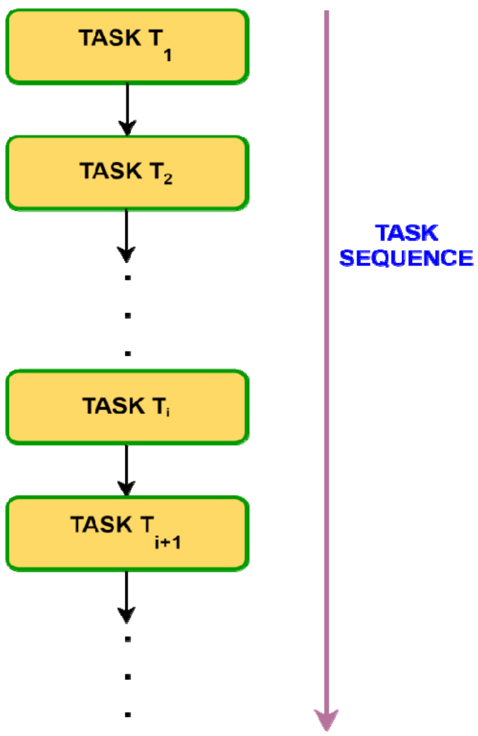
Even though machine learning is an important technique for knowledge empowerment that is being extensively utilised in a variety of different market areas, the present machine learning technique is likely insufficient for progressively active learning. Standardised machine learning is standalone learning that does not preserve the information gained from prior assignments (Liu, 2017). It is also known as catastrophic forgetting (CF), so it refers to the situation wherein learning several challenges causes the prior loss in the task's information. Humans may learn progressively with limited time by gaining new information and retaining earlier acquired knowledge. Developing a smart IoT system with an incremental learning method has proven to be a difficult challenge.

Most of the technique uses a gradient descent optimisation to identify appropriate universal minimum with the lowest cost about the tuning weights and parameter. Gradient descent minimises for every iteration by using the loss function where the loss is defined as the divergence among infers outcomes and expected forecasting outcomes while training data. If we are forecasting sales, for instance, loss equals actual sales minus anticipated sales in every observation. The differential loss in between expected probability and exact class for categorisation issue is being examined. The learning method is involved in providing training to the algorithmic approach with the training data and then adjusts the algorithm's weights to steadily reduce the loss during the iteration process. In an idealistic circumstance, a pleasant route is expected from the beginning of the algorithm up to the optimisation point. It might not be the situation owing to network facility or service interruptions via cloud computation. The intricacy issue is not easy to handle and therefore, to deal with these issues regularly and several data are utilised for training. Many days are continuously involved in training to optimise a deep learning method.

More suited for academics that use Google's free accessibility, AWS's free account and other similar services. Whether the connectivity is lost, then the same instance cannot be accessed again, resulting in a loss of critical training period. Therefore, the training process is restarted again. Cloud computing encloses a COLAB account automatically cancelled after 12 hours for each instance. Furthermore, an identical situation in a typical internal server arrangement or, to put it another way, a personal computer with the power-saver switched on are observed.

In this study, we aim to build a limited resource cloud-computed decentralised IoT device with a proposed incremental learning architecture (Gupta et al., 2017; Kumar et al., 2017). We concentrate on the scenario when a succession of tasks appears regularly, as illustrated in Figure 1. Only a single task is permitted for every training period which is completely distinct from the prior classes of a task. If the relevant training data are ready, the goal is to gradually learn additional tasks and create a competing method for all of them. For instance, an IoT device installed in a user's home can recognise activity through a model that has previously been trained. This trained model is believed to be able to distinguish behaviours including running and walking. While the user conducts new actions, including lying and leaping, the IoT device is intended to learn to categorise those new activities gradually while not forgetting about the previous ones.

Figure 1 Incremental learning for IoT devices (see online version for colours)



The model is developed not just to identify new classes more accurately, and to ensure that existing classes are not forgotten. The recognition method is enhanced by retraining it during offline mode using fresh class data combined with previous training samples, this method has several drawbacks:

1 the refinement is severely slowed since retraining depending on the cumulative training samples could take a long time

2 the refinement poses a privacy risk since it necessitates the previous training data

3 the retraining would then consume a huge amount of computation and storage resources, making it unsalable and unsuitable for constrained resource environments.

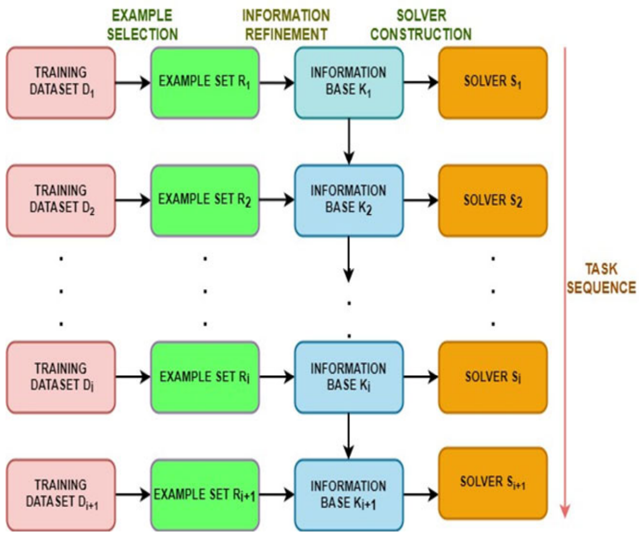
2 Literature review

Because the information gained from prior tasks would be discarded while learning the current task, CF offers a special problem for continuous learning (Rebuffi et al., 2017). Various techniques are being developed to cope with CF, which may be classified into different categories dependent on old data else independent of old data. Parts or the entire old data are required by the procedures in the initial category and utilise rehearsal to avoid forgetting (Rebuffi et al., 2017). While performing the rehearsal process, data are acquired from previous training parts are randomly selected and combined with those from the present training part. The pseudo rehearsal is introduced encloses forgetting is avoided by randomly creating samples and sending them via the network to acquire labels (McCloskey and Cohen, 1989). In practice, such samples with their corresponding labels are utilised. A prioritised exemplary selection is executed with store examples to update data representation, as well as accomplish classification using a nearest-mean-of-exemplars algorithm to maintain the information of past classes (Robins, 1995). Gradient episodic memory is a method proposed to tackle the issue of forgetting through the effective storage of data from prior training periods and conducting gradient upgrades includes the error on earlier acquired tasks does not rise (Lopez-Paz et al., 2017). The primary distinction between the proposed approach with these previous solutions shows that old classes with genuine data need not be preserved. Real data is not permitted to be retained for a longer time in real-world applications because of privacy and regulatory issues, as well as the limited resource restrictions of IoT devices. Because our technique extracts information from illustrations, the only information to be retained is the encoded example and decoder network that have nothing to do with data privacy. As a result, the proposed technique not only protects data security, but that also saves devices computational and memory storing capacity expenses. A data augmentation method based on the closest class means (NCM) is described (Hacene et al., 2018). On picture data categorisation, transfer learning is combined with DNN includes majority voting is used to accomplish incremental training precision. Adaboost's distribution updating mechanism has been tested (Polikar et al., 2001). The weak learners provided a preliminary approximation of the decision boundary, allowing for quicker training. Such weak learners acquire numerous decision boundaries utilising separate subsets of data, resulting in segmented learning for every group of weak learners. To create the categorisation, it is subsequently combined with weighted scoring. Stochastic gradient descent (SGD) is successful in achieving incremental learning via a persistent data collection in which the full data is an essential parameter and memory demand updating is the route to be proceeded (Bottou, 2010).

3 System model

We initially introduce the IL-IoT architecture and its primary components in this part, after which we extensively discuss the three important phases and offer algorithmic specifics. Each level has two cooperating parts namely an information base as well as a task resolution paradigm, which help preserve prior acquired knowledge and aid knowledge accrual known as a solver. In the proposed IL-IoT architecture, the information base comprises encoded examples and associated decoders for every task. Thus, the task pattern with the information base is constantly improved. We may then acquire the solution for every stage depending on the constantly enhanced information base in every stage. It consists of three critical phases, as indicated in Figure 2. To decrease computational complexity, an autoencoder is utilised to construct hidden representations layer of the data where the herding method is utilised to pick examples. To decrease computational complexity as well as preserve data privacy, an autoencoder is utilised to learn representations of all the examples and gather the information that includes the encoded examples with their decoder which is being maintained and transmitted to the next task. We rebuild the encoded examples into examples with actual dimensions using the retrieved information from the previous challenges.

Figure 2 IL-IoT architecture (see online version for colours)



The number of examples is reduced for the previous tasks to maintain the storage needs steady. The examples acquired for all classes through integrating the examples for previous tasks with the examples for the recently arriving task. This allows us to enhance our information and gain newly refined information. The solver construction is acquired based on our enhanced refined information.

The actual training data to be considered is being represented as $D_i = \{(X_i, Y_i), i = 1, 2, \ldots\}$, $X_i = \{x_{i,j}\}_{j=1}^{j=n_i}$, $Y_i = \{y_{i,j}\}_{j=1}^{j=n_i}$. The conventional autoencoder approach is implemented to map the actual data $X_i \in [0, 1]^d$ via the function $Z_i = F_i(W_iX_i + b_i)$ to the hidden place $Z_i \in [0, 1]^{d'}$ is being parameterised via $\theta_i = \{W_i, b_i\}$ to minimise the computational complexity. The vector reconstruction $\hat{X}_i \in [0, 1]^d$ is being mapped back with the hidden representation Z_i in which $\hat{X}_i = \zeta_i(W_i, b_i\}$ with element $\beta_i = \{W_i', b_i'\}$. The minimum optimisation of variables F_i and ζ_i, then the mean reconstruction error is given as follows:

$$\theta_i^*, \beta_i^* = \arg_{\theta_i, \beta_i} \frac{1}{n} \sum_{j=1}^{n_i} L\left(X_i, \hat{X}_\imath\right) \tag{1}$$

Here, the loss function L including the mean squared error is given as $L(X_i, \hat{X}_\imath) = \left\|X_i - \hat{X}_\imath\right\|_2$. The information of the prior tasks is saved to avoid catastrophic forgetfulness. The examples are considered of every class as task information and transmit it to succeeding tasks when examples can roughly reflect the entire training data. Moreover, the total number of examples Q is assumed that can be kept due to the limited resources of IoT devices. There are $q = Q / c$ examples for every class, in which c is the set of classes previously examined. A representative group of samples is constructed on account of the entire training data, similarly to the herding method. Samples being chosen are added to the example set for each class k, causing the mean feature vector across all examples to closest approximate the mean feature vector across total training samples. Assume the mean feature vector as $\mu_i^{(k)}$ with all training samples of class k is being estimated via the following expression:

$$\mu_i^{(k)} = \frac{1}{q} \sum_{j=1}^{n_i} z_{i,j} 1_{yi,j=k} \tag{2}$$

Here, an indicator function that is utilised to select class k with encoded samples is represented as $1_{yi,j=k}$. In task T_i, the examples of class k is chosen in the following order:

$$r_{i,h}^{(k)} = \arg\min_{t \in [1,2,\ldots,n_i]} \left\| \mu_i^{(k)} - \frac{1}{h}\left[z_{i,t=k} + \sum_{j=1}^{h-1} z_{i,j} 1_{yi,j=k} \right] \right\| \tag{3}$$

Thus, the examples set R_i for every task, i is acquired by aggregating the examples of all tasks of classes.

Because examples can roughly reflect the entire training sample, thus, the examples are transferred to later tasks and it wants to maintain information from previous tasks. Direct transmission of examples without any additional processing will constitute a data privacy issue. In light of this issue, the information is extracted from these examples before transmission, not only to preserve the information and to safeguard data privacy.

The superficial autoencoder F_i is employed to map the examples to the hidden layer, and then the function G_i to transfer backward to reconstructed examples. Equation (1) may also be used to improve the variables of F_i and ζ_i. The retrieved information K_i for the previously arrived i tasks may then be formed using the encoded examples $F_i(R_i)$ as well as the function G_i, in which $K_i\{F_i(R_i), \zeta_i\}$. To avoid CF, learned the previous i tasks while learning the new task T_{i+1}. Therefore, by utilising the following expression, the reconstructed examples are derived $\hat{R}_i$ in actual dimension depending on the transmitted information K_i,

$$\hat{R}_i = \zeta_i\left(F_i\left(R_i\right)\right) \tag{4}$$

Since the number of examples that may be saved is set at Q, the existing examples of all previous tasks reduce before combining them with the examples of the newly arriving task T_{i+1}, that can be retrieved through the example selection phase. The initially available q examples have to be picked first and acquire the reduced examples set $\hat{\tilde{R}}_i = \hat{R}_i[1, 2, \ldots, q]$ because the examples set R_i is a priority list, where q is the goal number of examples for each class in the present training phase. Therefore, by combining the decreased examples $\hat{\tilde{R}}_i$ for all previous tasks with the present task's examples $\overline{R_{i+1}}$, the examples for all $i + 1$ task is derived as follows:

$$R_{i+1} = \left(\bigcup_{j=1}^{i} \hat{\tilde{R}}_j \right) U R_{i+1} \tag{5}$$

Utilising equation (1), the information is refined depending on the refined examples R_{i+1} and acquire the refined information $K_{i+1} = \{F_{i+1}(R_{i+1}), \zeta_{i+1}\}$.

The nearest-mean-of examples classification approach is utilised in ICaRL, to complete the classification due to the limited resources of IoT devices. The refined examples including mean value $\alpha_i^{(k)}$ for every class, k is then computed utilising function ζ_i to forecast the labelling for a fresh sample.

$$\alpha_i^{(k)} = \frac{1}{q} \sum_{j=1}^{n_i} \zeta_i\left(z_{i,j}\right) 1_{yi,j=k} \tag{6}$$

The class label is provided with the maximum resemblance by evaluating the resemblance of the new input x to the average value of every class.

$$y^* = \arg\min_{k \in [1,2,\ldots,c]} \left\| x - \alpha_i^{(k)} \right\| \tag{7}$$

Thus, the proposed incremental learning technique on cloud computation decentralised IoT devices.

4 Result and discussion

The average value and fluctuations of a three-axis-based gyroscope, accelerometer, and magnetometer are used to create the SDA dataset which further encloses 285,000 samples. Every sample represents different every day and sporting activities. Nineteen different activities are to be chosen such as sitting, lying on your back, standing, lying on your right side, etc. The SDA dataset is divided into ten tasks, where the first two classes contain nine tasks then the next class contains only one task. Later, five tasks are considered in which the first four classes contain four tasks, and the next three classes contain one task in our studies. With the incremental task count, similar findings for the corresponding task are found in the categorisation levels of accuracy mostly on the SDA dataset (in which four classes for initially available four tasks whereas three class for final one task). The proposed IL-IoT technique performance is compared to three other ways. The upper bound is abbreviated as UB is determined by considering the method that will be trained using the training data. Nothing is a poorly trained method with the baseline, i.e., when CF occurs but no attempt is made to relieve it.

The accuracies may drop dramatically from 100% to 6% while CF is not addressed is being shown in Figure 3, demonstrating that the approach is useless and not suitable for incremental tasks while CF is not addressed. The accuracy of ICaRL trials is roughly 21% higher than that of none tests, with a mean accuracy of 63.49%.

Figure 3 Accuracy vs. five incremental task (see online version for colours)

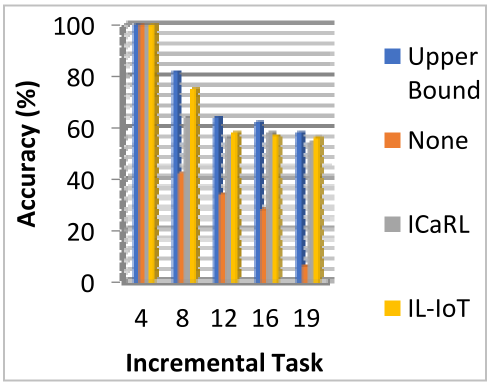
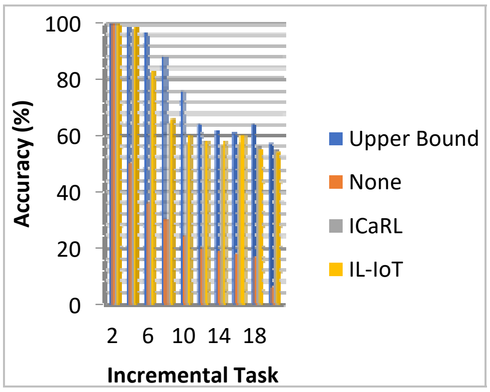


Figure 4 Accuracy vs. ten incremental tasks (see online version for colours)



The proposed IL-IoT achieves around a 3% gain in accuracy in comparison with the ICaRL technique, demonstrating that our IL-IoT is better efficient rather than the ICaRL technique. The accuracy rates of incremental tasks range from 100% to 57% in UB trials is shown in Figure 4; however, when there are four incremental tasks then the accuracy rates can be greater than 88%. Moreover, while CF is not addressed, accuracy is plummeting when the task count is increased to four so the accuracy would drop to 22%, demonstrating the massive impact of CF. The accuracy rates of ICaRL trials are on average 37% better than none test. If four tasks are assigned, then the accuracy is 67%. In the proposed IL-IoT tests, the accuracies improved by 39% and 3.6%, respectively, as compared to the ICaRL and no methods. If four tasks are used, then the accuracy rises to 74%, a gain of 8.4% in comparison with the ICaRL technique.

5 Conclusions

The IL-IoT is a simple incremental-learning architecture for IoT devices that comprises of two cooperating parts namely a constantly upgraded information base as well as a task resolution model, as described in this paper. The proposed incremental learning is accomplished with this approach when avoiding CF, safeguarding the confidentiality and lowering computational needs. Utilising two standard datasets, our experimental study shows encouraging results when in comparison with available methods, acquiring the overall accuracy gain of 3.53% and 45% reduction in computational cost. Finally, the proposed IL-IoT technique not only achieves greater accuracy over ICaRL at a lower cost but also highlights a strategy that eliminates loss of time during training owing to frequent disconnection from the cloud infrastructure. Experiments describe the effectiveness of this technique. It will allow academics and data experts who do not have a lot of processing capacity on their local computers to utilise a cloud computing infrastructure to train their methods without having to worry about network disconnection.

References

Bottou, L. (2010) *Large-scale Machine Learning with Stochastic Gradient Descent*, Paper NEC Labs America, Princeton NJ 08542, USA.

Gupta, C. et al. (2017) 'ProtoNN: compressed and accurate kNN for resource-scarce devices', in *ICML*.

Hacene, G.B., Gripon, V., Farrugia, N., Arzel, M. and Jezequel, M. (2018) 'Transfer incremental learning using data augmentation', *ICLR*.

Kumar, A., Goyal, S. and Varma, M. (2017) 'Resource-efficient machine learning in 2 KB RAM for the internet of things', in *ICML*.

Liu, B. (2017) 'Lifelong machine learning: a paradigm for continuous learning', *Front. Comput. Sci.*, Vol. 11, No. 3, pp.359–361.

Lopez-Paz, D. et al. (2017) 'Gradient episodic memory for continual learning', in *NIPS*, pp.6470–6479.

McCloskey, M. and Cohen, N.J. (1989) 'Catastrophic interference in connectionist networks: the sequential learning problem', *Psychol. Learn. Motiv.*, Vol. 24, pp.109–165.

Polikar, R., Udpa, L., Udpa, S.S. and Honavar, V. (2001) *Learn++: An Incremental Learning Algorithm for Supervised Neural Networks*, IEEE.

Rebuffi, S.A., Kolesnikov, A. and Lampert, C.H. (2017) 'ICaRL: incremental classifier and representation learning', in *CVPR*.

Robins, A. (1995) 'Catastrophic forgetting, rehearsal and pseudo rehearsal', *Connect. Sci.*, Vol. 7, No. 2, pp.123–146.

Wu, Q. et al. (2014) 'Cognitive internet of things: a new paradigm beyond connection', *IEEE Internet Things J.*, Vol. 1, No. 2, pp.129–143.