Properties of federated averaging on highly distributed data

Nirmit Desai[a] and Dinesh Verma[a]

[a]IBM T. J. Watson Research Center, Yorktown Heights, NY, USA

ABSTRACT

Tactical edge environments are highly distributed with a large number of sensing, computational, and communication nodes spread across large geographical regions, governments, and situated in unique operational environments. In such settings, a large number of observations and actions may occur across a large number of nodes but each node may only have a small number of these data locally. Further, there may be technical as well as policy constraints in aggregating all observations to a single node. Learning from all of the data may uncover critical correlations and insights. However, without having access to all the data, this is not possible. Recently proposed federated averaging approaches allow for learning a single model from data spread across multiple nodes and achieve good results on image classification tasks. However, this still assumes a sizable amount of data on each node and a small number of nodes. This paper investigates the properties of federated averaging for neural networks relative to batch sizes and number of nodes. Experimental results on a human activity dataset finds that (1) accuracy indeed drops as the number of nodes increase but only slightly, however (2) accuracy is highly sensitive to the batch size only in the federated averaging case.

Keywords: Federated learning, distributed data, privacy

1. INTRODUCTION

Tactical edge environments are highly distributed with a large number of sensing, computational, and communication nodes spread across large geographical regions, governments, and situated in unique operational environments. In such settings, a large number of observations and actions may occur but each node may only have a small number of these data locally.[1]

Further, there may be technical as well as policy constraints in aggregating all observations to a single node. Especially in coalition military operations where the resources are governed by autonomous partners, flow of information is highly regulated and incurs operational overhead. Communication resources among the nodes may be limited, or no suitable central node may have sufficient computational resources to support downstream analysis on aggregated data. Most importantly, privacy and security concerns may prevent export of data from the nodes. Learning from all of the data may uncover critical correlations and insights. Machine learning and deep neural networks in particular have proven to be powerful techniques in effective learning from a wide variety of data.[2,3] However, a large body of the existing machine learning apparatus assumes access to aggregated data centrally.

Hence, without having access to all the data, most of the well-known machine learning techniques cannot be applied. Recently proposed federated learning approaches[4] allow for learning a single model from data spread across multiple nodes. In such methods, a central parameter server coordinates an iterative learning process. In each iteration, each node trains a local deep neural network model with a small batch of local data and reports model parameters to the parameter server. No raw data is exported. The parameter server applies an aggregation function to the parameters, e.g., averaging, and sends back the aggregated parameters. In the next iteration, each node begins its local training process using the aggregated model parameters on the next batch. A convergence criteria can be used to terminate the learning process. Although this approach has been shown to be effective across several datasets, the focus has been on communication efficiency on image datasets such as MNIST.

Further author information: (Send correspondence to Nirmit Desai)
Nirmit Desai: E-mail: nirmit.desai@us.ibm.com, Telephone: 1 914 945 2612
Dinesh Verma: E-mail: dverma@us.ibm.com, Telephone: 1 914 945 4599

However, for such an iterative federated learning approach to work, each node must have sufficient samples to perform a training iteration with a batch of samples. In many practical scenarios, each node may only have a small number of local observations. As a result, the batch size during an iteration of training must be small. Also, the number of nodes may be very large. Although such a setting is quite challenging for any learning algorithm, a first step in discovering suitable approaches is to investigate the properties of federated averaging when batch sizes are small and number of nodes increase. Interestingly, in the context of convex loss functions as in the case of SVMs, it has been shown that a federated averaging approach is equivalent to a centralized training approach, regardless of the number of nodes, batch size, or the statistical distribution of data across nodes.[5] However, neural network loss functions are non-convex and hence these properties need investigating in the context of federated averaging for neural networks. Recent works have investigated the impact of skewed data on federated averaging, but the number of nodes and batch sizes have not been studied.[6]

This paper investigates the effect of small batch sizes as well as large number of nodes on federated averaging in the context of a human activity dataset USC-HAD.[7] The results are compared with the baseline accuracy achieved with centralized training. While controlling for other hyper parameters, the results show that increasing the number of nodes causes a drop in accuracy, however the drop is not significant. However, reducing the batch size causes a significant drop in accuracy. This behavior is observed under two settings: one where the batch size is adjusted as the number of nodes increase and another where the batch size is kept constant as number of nodes increase. Further, in the extreme case where each node has a only a single training sample, the accuracy is actually lower than the case where each node simply learns a model with local data, without participating in federated averaging.

2. LARGE-SCALE FEDERATED LEARNING

We motivate the problem of federated learning from a large number of distributed nodes in the context of a battlefield scenario and describe a human activity dataset that suitably enables experimental evaluation of federated learning approaches.

2.1 Tactical Edge Scenario

Situational awareness and situational understanding in coalition military operations[8] are critical. A large number of intelligent sensors for collecting and analyzing data are deployed, leading to the concept of *Internet of Battlefield Things (IoBT)*. In the IoBT, the *things* are intelligent computational nodes that communicate with each other and coordinate their operations.[9,10]

The battlefield setting presents a number of barriers to learning from data distributed across the nodes. For example, devices will be distributed and move around over a wide geographical area, often in regions with poor or no communications infrastructure. Communications will need to be wireless and suffer from high latency and high costs. Hence, the current popular model of transferring all data to cloud services for model training will not be feasible.

More importantly, the nodes across the tactical edge may be controlled by autonomous entities including governments, civilians, and partnering commercial enterprises. As a result, flow of raw data is greatly restricted by regulations and policies. For example, in a number of coalition military operations, it may be valuable to detect anomalous movement of a solider or a company based on IMU sensors on the devices carried by them. As there are potentially thousands of such IMU sensors deployed across the battlefield personnel, training an anomaly detection model centrally runs into challenges described above. Further, a single IMU sensor carrying device may have a very small number of anomalous samples to learn from.

2.2 USC HAD Dataset

The USC HAD dataset has been widely used in the tasks related to human activity recognition.[11] The data is gathered from 14 human subjects as they engage in 12 kinds of basic physical tasks such as walking and running.

A single MotionNode sensor with accelerometer and gyroscope is firmly attached at the subject's front right hip. Front right hip is chosen as the location to wear the sensor because it is one of the top 5 locations where people carry their mobile phones when they are out and about in public spaces. Since MotionNode is a wired

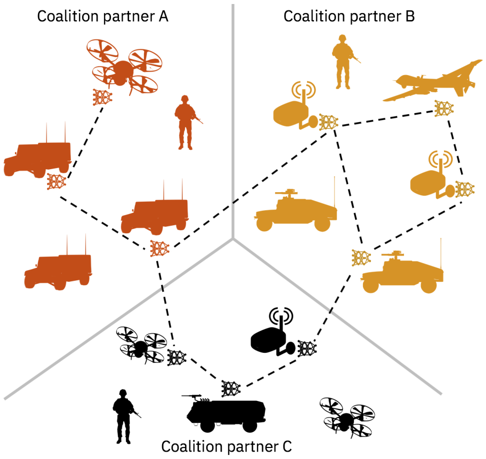


Figure 1. Tactical edge setting in the context of IoBT

device, the MotionNode is connected to a miniature laptop via a long and soft cable to record sampled data. The subject is then asked to perform a trial of specific activity naturally based on ones own style. In order to capture the day-to-day activity variations, each subject was asked to perform 5 trials for each activity on different days at various indoor and outdoor locations. Although the duration of each trial varies across different activities, it is long enough to capture all the information of each performed activity. On average, it took 6 hours for each subject to complete the whole data collection procedure.

2.3 Centralized learning

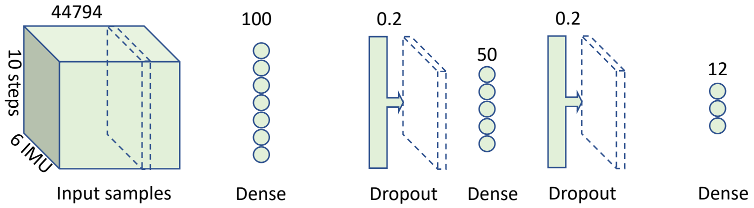
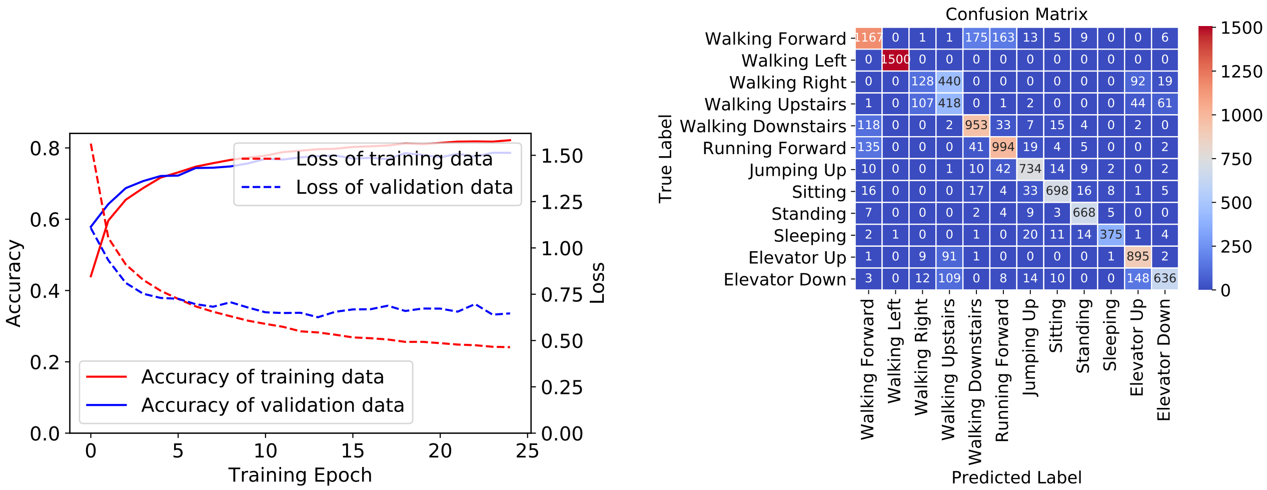


Figure 2. Feed-forward multi-layer perceptron architecture for activity recognition

Figure 2 depicts the architecture of a deep-learning model we employ for activity recognition. It is worth noting that the specific model architecture or a modeling approach is not relevant to the contributions of this paper. The idea of the paper is to show that the model achieves a reasonable accuracy on the activity recognition task when all data is available for training and compare the accuracy when the data is distributed and federated averaging is applied with each node using the same model architecture and parameters.

Each sample in our dataset corresponds to accelerometer and gyroscope IMU readings over a time window of 0.5 seconds with a sampling rate of 20Hz, i.e., 10 readings per sample. With x, y, and z values for accelerometer and gyroscope, and age, weight, and height of the human subject, each sample corresponds to an input vector of 63 dimensions, and a label corresponding to one of 12 activities. We partition the dataset by keeping first four trials from each subject in the train set and the fifth trial in the validation set.

Figure 3 shows the classification accuracy results in the case with centralized data, i.e., no federation, with 25 epochs and a batch size of 64 samples. About 80% of the samples are correctly classified in the validation dataset. The confusion matrix explains why the task is a non-trivial one with similarities between several activities resulting in erroneous classification.



(a) Accuracy and loss, batch=64 (b) Confusion matrix, batch=64

Figure 3. Classification accuracy results in the centralized learning case

2.4 Federated averaging

Having evaluated accuracy of the centralized learning case, we are ready to evaluate and compare the federated averaging approach. We keep the model architecture and hyper parameters the same and experiment with batch sizes as described in the following experiments. The centralized case represents the best-case scenario for federated averaging results, so what about the worst-case scenario? If each node simply trains a model from its local data and validates it against the global validation set, the corresponding accuracy may represent the worst-case performance. Figure 4(a) shows mean accuracy across nodes as the number of nodes varies from 1 to 80, with the accuracy of 1 node identical to the centralized case.

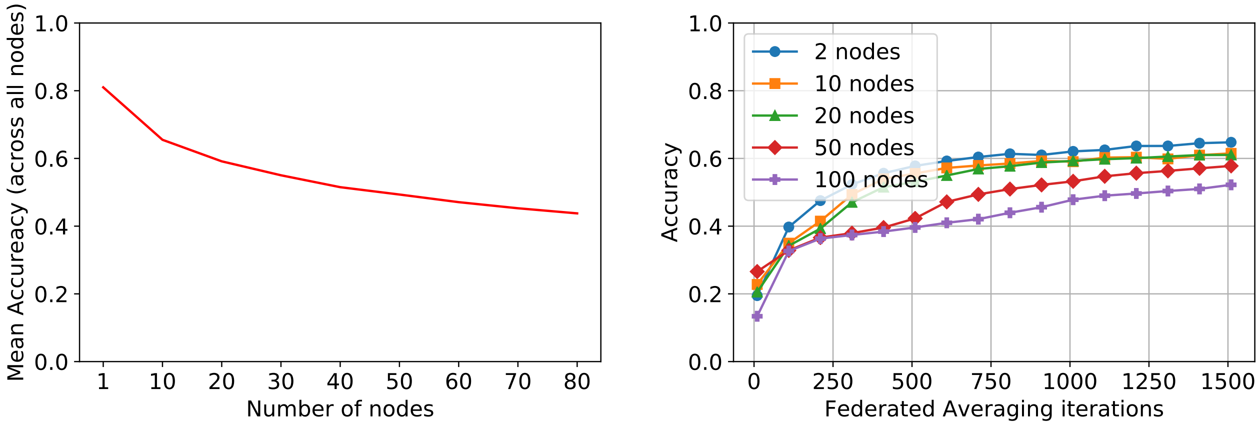


Figure 4. (a) Independent node accuracy, batch=64 (b) Federated averaging accuracy, batch=64

For federated averaging, we adopt the well-known synchronous updates algorithms[4] wherein training proceeds over a number of iterations. In each iteration, each node computes the loss on a mini-batch of local data and updates local weights based on SGD on the loss function. The local weights are reported to a parameter server which waits for reports from all nodes. The parameter server then computes a simple mean over the weights and broadcasts the mean weights to all nodes. The nodes begin the next iteration with the weights received from the parameter server. For learning algorithms with convex loss functions, it has been shown theoretically that as long as the weights are averaged after each mini-batch, the distributed gradient descent provides the same mathematical progression of the weights as centralized gradient descent.[5] Further, training via distributed gradient descent does not depend on the number of nodes of batch size as long as they synchronize after each mini-batch. However, neural network models have highly non-convex loss functions and hence are sensitive to both the batch size as well as the number of nodes as shown in the results below.

Figure 4(b) shows the accuracy of training via the federated averaging approach with 1600 iterations as number of nodes vary from 2 to 100 and the batch size is kept constant at 64 samples, same as the centralized case. The number of iterations correspond to the total number of training steps in the centralized case. Clearly, with these parameters, federated averaging accuracy is significantly worse than the centralized training accuracy and the accuracy drops further as the number of nodes increase. Also, given the plateau of accuracy curves, it is clear that a larger number of iterations would not yield higher accuracies. Clearly, the federated averaging algorithm's properties and its implementation need a deeper investigation.

To investigate the effect of batch size further, we set the batch size such that for each iteration of training across all nodes, the total number of samples is identical. Hence, for global batch size of 64, nodes=2 implies batch size=$\frac{64}{2}$, nodes=10 implies batch size=$\frac{64}{10}$, and so on, getting down to the batch size of 1 for 50 nodes. As shown in Figure 5(a), lower batch sizes yield significantly lower accuracy, approaching the independent accuracy achieved by nodes in Figure 4(a).

If smaller batch sizes do not work, perhaps the results are better when batch size is increased in the federated case? Indeed, with a larger constant batch size of 256 samples as number of nodes increase, the federated averaging accuracy approaches close to the centralized training accuracy as shown in Figure 5(b). Although number of nodes still cause a small drop in accuracy, 256 is a much better batch size than 64 for the federated averaging case. Further, batch size of256 is no better than the batch size of 64 for the centralized case, can be compared across Figures 6 and 3(a), which is an interesting result in this paper. Perhaps in the case of federated averaging, the quality of weight updates computed on each node is more important than the number of batches of samples on the node. Although this is an intuitive explanation, further investigation is needed in substantiating it.

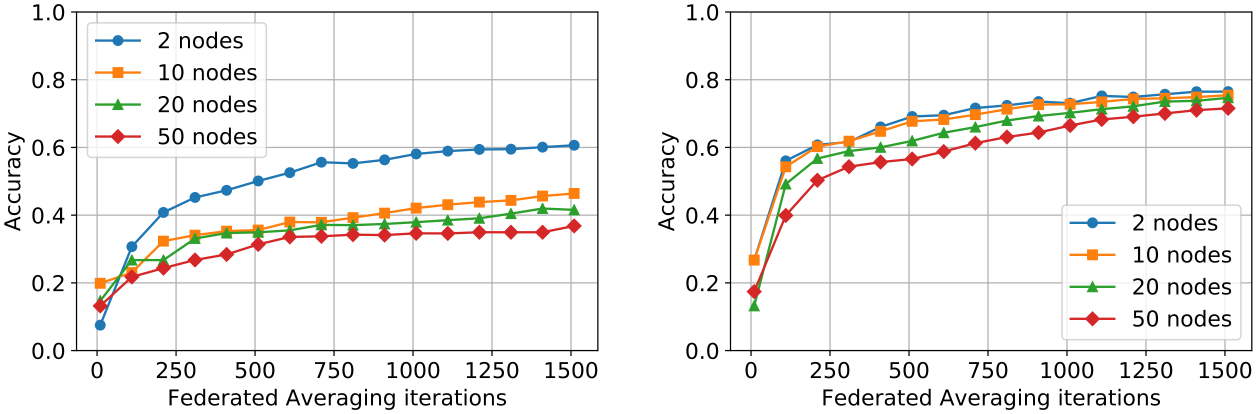


Figure 5. (a) Federated averaging, small batches: batch(2)=32, batch(10)=6, batch(20)=3, batch(50)=1 (b) Federated averaging accuracy, batch=256

3. CONCLUSIONS

Motivated by coalition military operations with a large number of distributed computational and sensing nodes with local data, we investigated the properties of federated averaging on a human activity dataset. Specifically, we found that the quality of learning in the federated averaging setting is highly sensitive to batch sizes, which is not the case with the centralized learning setting. While the number of nodes participating in federated averaging also affects the accuracy, the drop in accuracy with increasing number of nodes is not significant. In summary, federated averaging may be a suitable tool for learning from highly distributed data but needs a deeper investigation in choosing the right hyper-parameters.

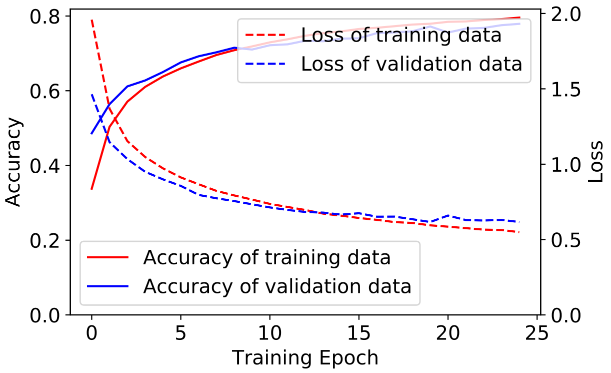


Figure 6. Centralized learning accuracy, batch=256

ACKNOWLEDGMENTS

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF-16-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are Authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

[1] Verma, D., Bent, G., and Taylor, I., "Towards a distributed federated brain architecture using cognitive IoT devices," in [*The Ninth International Conference on Advanced Cognitive Technologies and Applications*], (2017).

[2] Lecun, Y., Bengio, Y., and Hinton, G., "Deep learning," *Nature* **521**, 436–444 (5 2015).

[3] Goodfellow, I., Bengio, Y., and Courville, A., [*Deep Learning*], MIT Press (2016). http://www.deeplearningbook.org.

[4] McMahan, H. B., Moore, E., Ramage, D., and y Arcas, B. A., "Federated learning of deep networks using model averaging," *CoRR* **abs/1602.05629** (2016).

[5] Tuor, T., Wang, S., Leung, K. K., and Chan, K. S., "Distributed machine learning in coalition environments: Overview of techniques," *2018 21st International Conference on Information Fusion (FUSION)* , 814–821 (2018).

[6] Verma, D., White, G., Julier, S., Pasteris, S., Chakraborty, S., and Cirincione, G., "Approaches to address the data skew problem in federated learning," in [*Defense and Security Symposium - Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*], **11006**, 1100650, International Society for Optics and Photonics (2019).

[7] Murad, A. and Pyun, J.-Y., "Deep recurrent neural networks for human activity recognition," *Sensors* **17**(11) (2017).

[8] Preece, A., Cerutti, F., Braines, D., Chakraborty, S., and Srivastava, M., "Cognitive computing for coalition situational understanding," in [*2017 IEEE SmartWorld*], 1–6, IEEE (2017).

[9] Kott, A., Swami, A., and West, B. J., "The internet of battle things," *Computer* **49**(12), 70–75 (2016).

[10] Suri, N., Tortonesi, M., Michaelis, J., Budulas, P., Benincasa, G., Russell, S., Stefanelli, C., and Winkler, R., "Analyzing the applicability of internet of things to the battlefield environment," in [*Military Communications and Information Systems (ICMCIS), 2016 International Conference on*], 1–8, IEEE (2016).

[11] Zhang, M. and Sawchuk, A. A., "Usc-had: A daily activity dataset for ubiquitous activity recognition using wearable sensors," in [*ACM International Conference on Ubiquitous Computing (Ubicomp) Workshop on Situation, Activity and Goal Awareness (SAGAware)*], (September 2012).