Intelligent Edge Resource Allocation for Distributed Learning

Jared Lynskey, Dr. CS Hong

Kyung hee University, South Korea

jared@khu.ac.kr, cshong@khu.ac.kr

Abstract

Centralized machine learning techniques heavily depend on the reliability, performance and availability of data provided by one machine. New emerging techniques involving distributed learning such as Federated Learning allow for multiple local learning models distributed over several machines which could include Mobile Edge Computing servers. Hence a new dilemma, synchronizing these 'learned' models between MEC servers has arisen. Our paper shows the impact of varying data set sizes, targeted accuracy level, communication bandwidth and MEC computation resources can have on the amount of time it takes to update the global model as well as a technique to alleviate the maximum delay for learning a local model.

1. Introduction

Machine Learning and Deep Learning are currently hot topics in the field of Computer science. However, one dilemma with centralized machine learning is that the data must be accessible to a single machine. In the case of distributed learning cases previously proposed by related work still require the data to be gathered on single a machine first, then split into equals parts and finally redistributed across multiple servers. In our case, we propose an extension to distributed learning/federated learning so that the data samples can remain at the edge while being equally distributed among Mobile Edge Computing (MEC) to reduce the variation in computational delay. As a result, we shall eliminate the excess delay in computing the global model originally caused by MEC servers learning an incredible number of samples, more than the underutilized neighboring MEC servers.

2. Related Work

Papers in the field of distributed learning typically have a homogenous dataset and the batch sizes are typically defined before beginning the learning process as in [1]. In paper [2], they provide a system model including, communication costs for transmitting data wirelessly from user equipment to a single base station and offloading from a single base station to a resourceful backhaul. Furthermore, they apply the system model to propose a solution to solve minimal cost of communication. In [3] the authors consider the problem of Probably Approximately Correct learning from distributed data across multiple servers and analyze fundamental communication complexity questions involved including the upper and lower bounds for machine learning model to reach the targeted accuracy. In [4] the authors discuss the technology that detect, track and disarm possible amateur UAV threats, including the next generation networks 5G.

3. System Model

The system model illustrates the aggregation of local MEC server models. The goal is to minimize the maximum time it takes for all MEC severs to achieve the local model targeted accuracy. The grey broken line shows the accuracy is the average of accuracies across the MEC servers working together on the same learning model. MEC servers are able to directly communicate with each other through a wired channel to send and receive excess data samples to reduce the variation in data sizes between servers (red line).

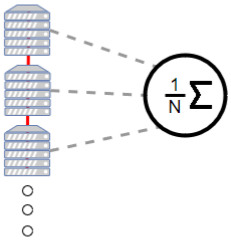

Figure 1. Distributed learning over multiple MEC servers

Optimization techniques such as Stochastic Gradient Descent are used to find the optimal weights in each heterogeneous data set contained across multiple MEC severs. The weights calculated at each local MEC server is then aggregated and averaged in order to generated a global model. In our model, we apply constraints including MEC computational capability and bandwidth between neighboring MEC servers. Our system's model goal is to minimize the time to reach a certain degree of accuracy. Here T is the maximum time to compute the global model among the set of MEC servers. Our goal is to minimize the maximum time.

$$\min(T)$$

T is calculated by adding the time to reach the target accuracy based on the number of Machine learning iterations denoted as I with the communication latency to the reallocate data (if incurred).

$$T = I + C$$

The required accuracy is one of the largest factors that determine the number of interations required. To have a feasible solution, a minimum number of iterations needs to be perform in order to satisfy our accuracy constrain (Err). Each MEC will contain d number of samples, and each MEC will have f available computational resource for a single MEC server. The following function represents the time it takes to converge to a solution that satisfies the constraints in respect to the number of iterations,

$$I = d^{\frac{log_2(1+\frac{1}{\theta})}{f_i}}$$

$$C = d / b$$

$$\text{Subject to. } C < C_{max}$$

$$Err < Er_{max}$$

The following graphs show the r impact of aggregated accuracy and dataset size.

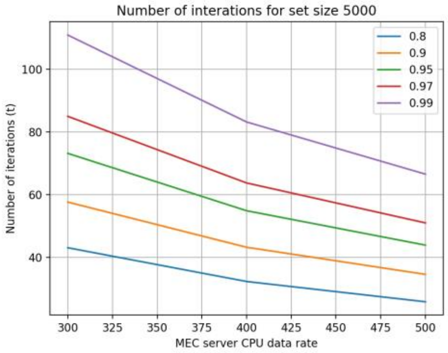

Figure 2. Effect of target error rate on the time to reach target accuracy.

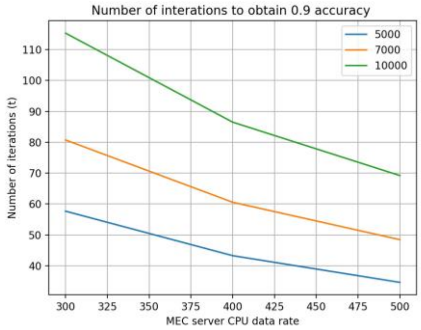

Figure 3 Effect of data size and target error rate on the time to compute.

We propose to solve the excessive delay for MEC servers by reducing the maximum data size subject to the constraints of data communication.

4. Simulation

To prove that our proposed method reduces the time to achieve the target accuracy, we conducted a simulation in Python with and without the process of reallocation of data samples. Each MEC server follows a normal distribution with a variation in the number of data samples. Our constraints, includes the delay due to reallocation and MEC server performance. C denotes the communication delay this is the delay incurred when sending data to a neighboring MEC server. For our simulation we required an accuracy of 0.95 and used a normal distribution for data sample sizes with a mean of 200 a variance of 40 samples. The power of the MEC server was not considered in this simulation and remained constant among the tests performed.

5. Evaluation

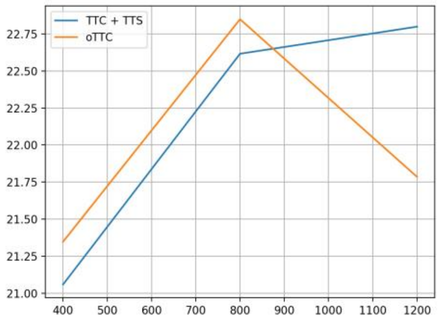

Figure 4. When the variation of the data sets is extreme, reallocating the data can save time.

'oTT' is the delay of Original time to compute without offloading. TTC + TTS is delay for computing plus time to send for reallocated data. Our proposed method improves the time to reach the targeted accuracy when there is a large amount of variance among the data set sizes stored on each MEC server. Our algorithm is even more advantageous when transmitting data to underutilized MEC severs is abundant and cheap. For MEC servers with data set sizes that are close to the mean set data set size, there is no need to reallocate data since the extra cost to send the data will outweigh the benefit of a reduction in variance.

6. Conclusion

In conclusion we have shown that our algorithm can improve federated learning by reducing the maximum time it takes to reach a targeted accuracy for a local model. The effectiveness of our proposed method depends on high variation between MEC servers and low bandwidth cost at the backhaul.

7. Future Work

Our next goal is to apply a constraint for the level of offloading subject to power consumption and renewable energy resources available for each iteration. Furthermore, apply this technique to a real world problem such as computer vision object detection among UAVs performing surveillance. Also to explore opportunities to increase performance when there is little to no variation in data size.

8. Acknowledgement

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2015-0-00567, Development of Access Technology Agnostic Next-Generation Networking Technology for Wired-Wireless Converged Networks) *Dr. CS Hong is the corresponding author.

9. References

[1] S. Kinkiri and W. J. C. Melis, reducing data storage requirements for machine learning algorithms using principle component analysis, 2016

[2] H. Zhang, J. Guo, L. Yang, X. Li, H. Ji, Computation Offloading Considering Fronthaul and Backhaul in Small-Cell Networks Integrated with MEC, IEEE Conference on Computer Communications Workshops 2017

[3] M. Balcan, A. Blum, S Fine, Y Mansour, Distributed Learning, Communication Complexity and Privacy, JMLR: Workshop and Conference Proceedings vol 1 (2012)

[4] D. Solomitckii, M. Gapeyenko, V. Semkin, S. Andreev, Y. Koucheryavy, Tehnologies for Efficient Amateur Drone Dectection in 5G Millimeter-Wave Cellular Infrastructure, 2018