CHAPTER 1

Edge or Cloud: What to Choose?

Author Arnab K. Paul

Affiliation Virginia Tech, USA

CONTENTS

M

ACHINE LEARNING (ML) models need to be trained on large volumes of data. Traditionally, client applications collect data and transfer it to cloud servers to train machine learning models, with the results returned to the clients. An alternative to this kind of an architecture is where data is trained at the site where it is is collected. Therefore, the data never needs to be uploaded to the cloud server. This is called Edge-based learning which offers the advantages of privacy preservation and reduced network latency. However, machine learning architects need to be aware of various requirements other than privacy to make an informed decision which to choose: Edge or Cloud? There is a lack of actionable information how these two kinds of learning differ in terms of performance and resource utilization. In this chapter, to address this problem, a comprehensive empirical evaluation of these two types of learning is conducted for a widely used clustering learning algorithm. The results will help in designing learning systems in the future and will help mitigate some challenges faced by ML applications.

1.1 INTRODUCTION

IoT (Internet of Things) holds tremendous promise for human society. Current state of the art uses the cloud computing infrastructure as the "brain" for processing and analyzing IoT data, and controlling IoT devices. The low-latency, scalability, and privacy requirements of future IoT applications are motivating the edge computing model: the evolution of the technological landscape that enables in-situ data processing and actuation. The data flow in the present scenario consists of IoT devices as the primary data collectors. These devices are generating increasing data deluge, which needs to be operated over to provide applications with high-quality services. Currently, there are two main approaches to operate over the data, a cloud-based approach in which the collected data is moved across the network to a cloud-based system for processing. Essentially, this approach aims to consolidate the economic utility model with the evolutionary development of many existing approaches and computing technologies, including distributed services, applications, and information infrastructures consisting of pools of computers, networks, and storage resources. The second approach is to train the data closer to the IoT devices – called the edge nodes, sending resulting models to a centralized server. The server operates over the aggregated models and updates models on the edge nodes.

There are well-known trade-offs between cloud-based and edge-

based computation. For example, cloud-based offers additional processing capabilities, with more resources being available at the cloud server. Cloud-based learning may also offer superior values for energy efficiency, as most resource intensive computations are performed at the remote server side. While edge-based offers the decoupling of the model training from the need for direct access to the raw training data. Additionally, the issue of privacy also comes to mind, as the raw data does not leave the confines of the IoT network. Sometimes, latency becomes an important metric to operate over the data. Edge systems will provide lower latency than cloud-based systems.

In order to ease the selection of edge or cloud-based systems for an application, a model is developed in this chapter which takes into account the various factors contributing to the performance of an application. The factors range from system metrics, like CPU and memory to the computation time as well as latency and energy consumption. The model also considers the size of the dataset. The model analyzes multiple runs of machine learning application, namely KMeans Clustering and comes up with a decision parameter which optimizes the factors that deem important in the analyses and lets applications ease the decision-making process to decide which system to select – cloud-based or edge-based, for its computation. Clustering is a machine learning technique that groups items together if they share some characteristics. This technique has been used to solve important problems in diverse domains that include medicine [20, 23, 22], finance [4], social sciences [3], and even search engine optimization [24].

To summarize, this chapter will focus on detailed analysis of the application in both cloud-based and edge-based setup. From the analysis, the optimization model will come up with a list of metrics which are important. Next, based on the inputs from the application, the model comes up with the decision for the application to be built either on cloud or edge.

1.2 BACKGROUND & RELATED WORK

In this section, the technical background required to understand our conceptual contributions is given.

1.2.1 Edge-Based Learning

`Edge-Based Learning` is a ML technique where the training dataset remains within the devices which are placed near the source of the data. All devices communicate with each other to compute the model. This ensure privacy preservation, reduced network latency, and less power consumption. The devices can also use the use the data after it is generated as the data will not be transferred to the cloud. The major disadvantage of this kid of learning is that it is limited by availability of hardware resources [17, 14, 18].

1.2.2 Cloud Computing

`Cloud Computing` is a distributed computing technique via which resources can be given to applications from a shared pool of resources. The resources can be data storage, compute power or even networks. There is no active management of resources required [19, 7]. This kind of an architecture has major advantages like elasticity, scalability and 'pay-as-you-go' models. For a Cloud-based learning, datasets need to be transferred across networks to the cloud servers thereby increasing risk.

1.2.3 K-Means

`K-Means Clustering` is a very popular ML algorithm where n observations are divided into k clusters based on which observation is closest to the mean. This reduces the variances within a cluster. An approach to compute K-Means is the iterative technique where in every iteration, means are calculated based upon a set of points and then after each subsequent iteration, the distance between the means and the points are minimized until they converge.

Work in The Chapter

This chapter conducts an analysis on the implementation of K-Means clustering technique over both Edge-based learning and Cloud-based learning to decipher some key findings for the resource utilization for both approaches. This chapter will hopefully be able to guide ML designers in selecting a system which best suit their needs.

1.3 EXPERIMENT

Both edge-based learning and cloud-based learning have been analyzed via a clustering algorithm. The input to the algorithm is a dataset containing geographical coordinates. The output is a set of 4 coordinates, which are basically the four clusters into which all points can be divided to the points with the closest distance.

1.3.1 Edge-Based Learning Procedure

In this kind of learning, client edge nodes collect the geographical data and generates a clustering model that can be then sent to a designated edge node acting as the server. The server combines the models from all the client nodes and gives as output the resultant model. The steps are again repeated for new data.

1.3.2 Cloud-Based Learning Procedure

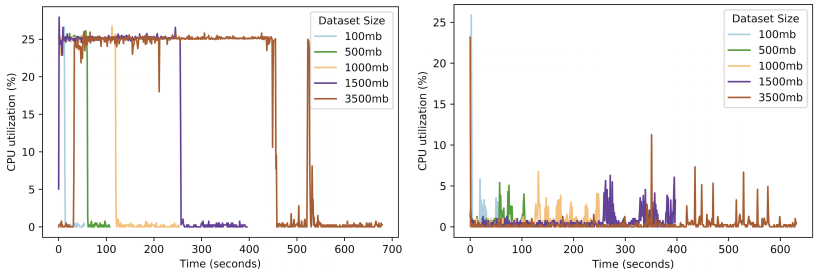
Here, the process differs from the edge-based learning procedure in the sense that clients are only responsible for collecting geographical data. Once the data is collected, it is sent to the clud server where the clustering algorithm is used to generate the resultant model on the overall data. This model is then sent back to the clients.

1.3.3 Experimental Objectives

The major objective of the experiments is to compare both learning approaches with respect to system resource utilization, in particular - CPU, memory, disk, network and energy. The reason is to understand the system behavior for machine learning algorithms to behave under an edge and a cloud setup. The findings will be extremely useful for both system designers as well as machine learning architects to find the learning procedure which best suits their model.

1.3.4 Setup

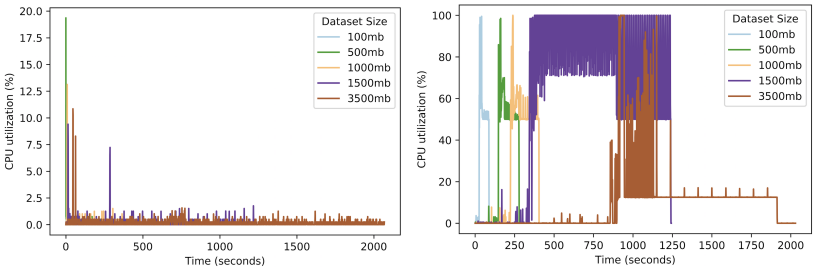
The client nodes for both edge-based learning and cloud-based learning are the same. All are Raspberry Pis 3 Model B, with Quad-Core 1.2 GHz and 1 GB RAM running Raspbian OS version3.0.1. There are 9 client devices for each setup. For edge-based learning, the server is located on a Raspberry Pi having the same configuration. In case of the cloud-based learning, the server is a AWS EC2 instance running



(a) % CPU Utilization in Edge-Based Clients

(b) % CPU Utilization in Edge-Based Server

Figure 1.1: Time-series graphs for CPU %Utilization for Clients and Servers in Edge-Based Learning



(a) % CPU Utilization in Cloud-Based Clients

(b) % CPU Utilization in Cloud-Based Server

Figure 1.2: Time-series graphs for CPU %Utilization for Clients and Servers in Cloud-Based Learning

Ubuntu Server 16.04 LTS, with 8 GB of RAM, with 2 CPU cores. The dataset sizes vary from 100 MB to 3.5 GB. To measure power, a "watts up? Pro" [9] power monitoring device is used. To estimate the power consumption of the AWS instance, energy estimate from Kurpicz et at. [15] is used.

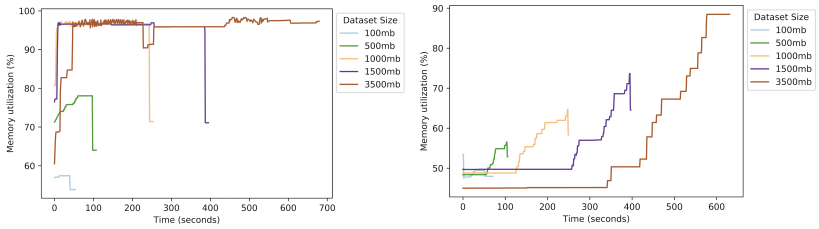
1.4 ANALYSIS

This section has the evaluation results for both edge-based and cloud-based learnings.

1.4.1 CPU Utilization

The CPU utilization (in percentage) is shown in Figures 1.1 and 1.2. It is seen that for clients, CPU utilization is much more for edge-based learning that cloud. The trend is the reverse in servers. This is because, for edge-based learning, individual models are generated in the clients and then passed onto the server, but in cloud the clients are only responsible for data collection.

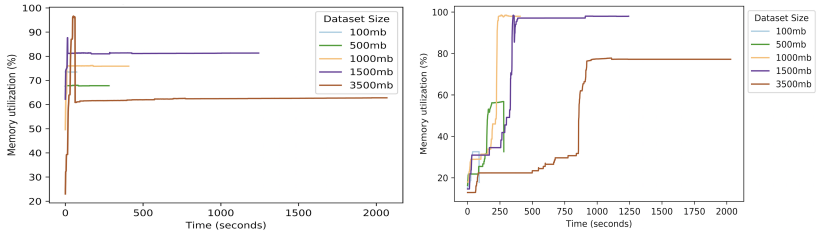
1.4.2 Memory Utilization



(a) % Memory Utilization in Edge-Based Clients

(b) % Memory Utilization in Edge-Based Server

Figure 1.3: Time-series graphs for Memory %Utilization for Clients and Servers in Edge-Based Learning



(a) % Memory Utilization in Cloud-Based Clients

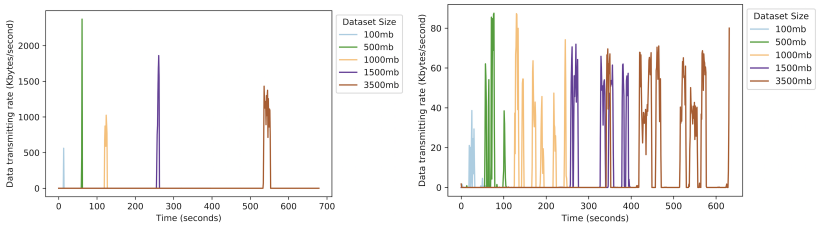
(b) % Memory Utilization in Cloud-Based Server

Figure 1.4: Time-series graphs for Memory %Utilization for Clients and Servers in Cloud-Based Learning

The behavior for memory utilization is similar to that of compute utilization as seen in Figures 1.3 and 1.4. For edge-based learning, memory usage is increased when the model is computed on the clients and then the usage deceases after the model is given to the server, after

which the memory usage of server starts increasing. For cloud-based learning, memory usage for clients is high at the beginning when data is collected, after which it gets low and the server's memory utilization increases after that due to the computation of the entire model.

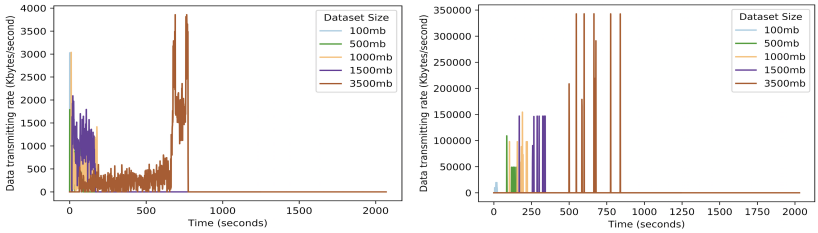
1.4.3 Data Transmission Rate



(a) Data Transmission Rate (KB/sec) in Edge-Based Clients

(b) Data Transmission Rate (KB/sec) in Edge-Based Server

Figure 1.5: Time-series graphs for Data Transmission Rate (KB/sec) for Clients and Servers in Edge-Based Learning



(a) Data Transmission Rate (KB/sec) in Cloud-Based Clients

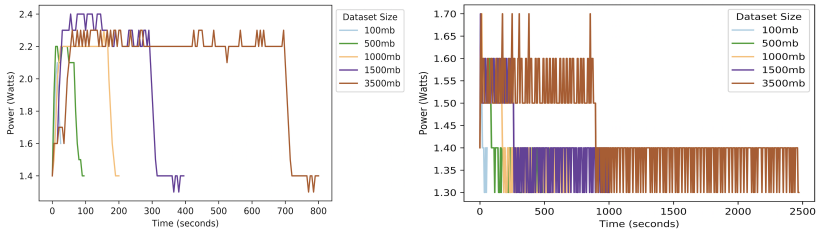
(b) Data Transmission Rate (KB/sec) in Cloud-Based Server

Figure 1.6: Time-series graphs for Data Transmission Rate (KB/sec) for Clients and Servers in Cloud-Based Learning

The data transmission rates in KBytes per second are shown in Figures 1.5 and 1.6. Cloud-based learning achieves much higher rates in data transmission over the network than edge-based learning. Edge-based learning clients have a smaller phase of data transmission over a smaller period of time than cloud-based clients, which are more continuous. This trend is reversed in case of servers where cloud server is much more discreet than edge-based server. But the edge-based server

has less data to transmit and therefore the data transmission rates are drastically lower in edge server than the cloud server.

1.4.4 Power Consumption



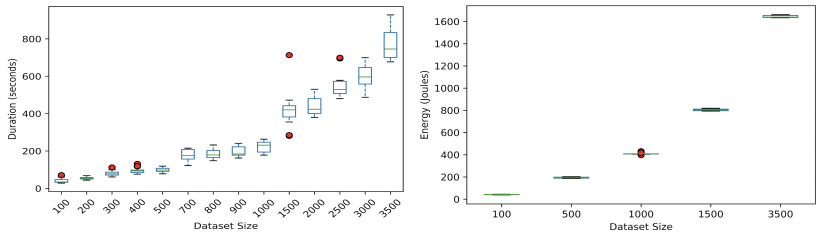
(a) Power Consumption in Edge-Based Clients

(b) Power Consumption in Cloud-Based Clients

Figure 1.7: Time-series graphs for Power Consumption for Clients in Edge-Based and Cloud-Based Learnings

The time series of power consumption, shown in Figure 1.7 shows that clients in edge-based learning consumes more power than in cloud-based learning. This is because in edge-based learning, clients are responsible for generating the model from the dataset but for cloud-based learning, clients are only responsible for transmitting the data collected.

1.4.5 Energy Consumption

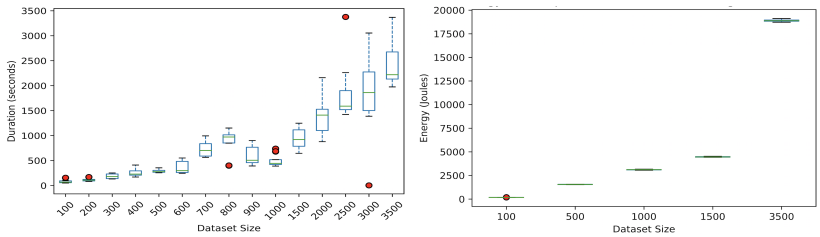


(a) Duration of Jobs in Edge-Based Learning

(b) Energy Consumption in Edge-Based Learning

Figure 1.8: Duration and Energy Consumption in Edge-Based Learning

Energy consumption is calculated by multiplying the duration of a job with power consumption. Power consumption was discussed in the



(a) Duration of Jobs in Cloud-Based Learning

(b) Energy Consumption in Cloud-Based Learning

Figure 1.9: Duration and Energy Consumption in Cloud-Based Learning

previous section. Here, the duration and energy consumption of edge-based and cloud-based learnings are shown in Figures 1.8 and 1.9. As can be seen in the figures, the energy consumption in cloud-based learning is multiple times higher than edge-based learning. This is a very important factor for designers to include when thinking about green computing.

1.4.6 Summary

Edge-based learning consumes less energy and due to the distributed setup of the model generation, also has a lower duration of each job. However, this results in higher CPU and memory usage of client edge nodes in edge-based learning than cloud-based learning. Due to AWS network being bteer than local network, the data transmission rates are higher in cloud-based learning than edge-based learning. However, data transmission needs to happen for a longer period of time in cloud-based learning than edge-based learning due to localized learning.

1.5 FINDINGS

This section focuses on discussing the overall findings for this chapter.

1.5.1 Edge-Based Learning

Edge-based learning acts upon localized data. Clients are responsible for data collection and generating a model for the collected data. This model is then sent to the server for aggregation. The major motivation for edge-based learning is data privacy, as sensitive data will not

leave edge nodes where the data is collected. Also, the overall energy consumption and time taken to generate the model is much low.

1.5.2 Cloud-based Learning

For cloud-based learning, all data is transferred from the data collection client nodes to the centralized cloud server where the machine learning model is generated. Data privacy is hampered in this approach. Also, the overall energy consumption is much higher. However the CPU and memory usage of clients are much less, which will elongate the lifetime of client edge nodes. But this also means that edge nodes are not used to the fullest. Also, data transmission rates are much higher because of an improved network interface in the cloud.

1.5.3 Comparison

The edge-based learning architecture is faster and consumes lesser energy than the cloud-based architecture when performing the same operations over data sets of the same size. However, client devices use more energy in edge-based learning due to model computation at the edge node. System architects and machine learning developers need to take into account the higher resource utilization on the clients for edge-based learning. Another important consideration is the server for the edge-based learning can be of a lower specification because of the lower resource utilization for edge-based learning. Cloud-based learning will incur higher costs due to network management for transmitting data from the clients to the cloud server.

1.6 CONCLUSION

This chapter focuses on giving a detailed analysis of the two types of learning: edge-based and cloud-based. Both learnings use k-means clustering algorithm to be evaluated.Both offer trade-offs in terms of resource utilization, energy consumption, network latency, data transmission and data privacy. While, edge-learning approach helps in lower energy consumption, cloud based learning helps in lower resource utilization for client nodes. Developers and architects can take help from this analysis to be better informed while selecting an approach for the machine-learning workloads.

Bibliography

[1] Khaled Alsabti, Sanjay Ranka, and Vineet Singh. An efficient K-Means clustering algorithm. 1997.

[2] Preeti Arora, Shipra Varshney, et al. Analysis of K-Means and K-Medoids algorithm for big data. *Procedia Computer Science*, 78:507–512, 2016.

[3] Tim Brennan and William L Oliver. Emergence of machine learning techniques in criminology: Implications of complexity in our data and in research questions. *Criminology & Pub. Pol'y*, 12:551, 2013.

[4] Fan Cai, Nhien-An Le-Khac, and Tahar Kechadi. Clustering approaches for financial data analysis: a survey. *arXiv preprint arXiv:1609.08520*, 2016.

[5] Truong Vinh Truong Duy, Yukinori Sato, and Yasushi Inoguchi. Performance evaluation of a green scheduling algorithm for energy savings in cloud computing. In *2010 IEEE international symposium on parallel & distributed processing, workshops and Phd forum (IPDPSW)*, pages 1–8. IEEE, 2010.

[6] Kamil Figiela, Adam Gajek, Adam Zima, Beata Obrok, and Maciej Malawski. Performance evaluation of heterogeneous cloud functions. *Concurrency and Computation: Practice and Experience*, 30(23):e4792, 2018.

[7] Ian Foster, Yong Zhao, Ioan Raicu, and Shiyong Lu. Cloud computing and grid computing 360-degree compared. *arXiv preprint arXiv:0901.0131*, 2008.

[8] John A Hartigan and Manchek A Wong. Algorithm as 136: A K-Means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

[9] Jason M Hirst, Jonathan R Miller, Brent A Kaplan, and Derek D Reed. Watts up? pro ac power meter for automated energy recording, 2013.

[10] Anil K Jain, Richard C Dubes, et al. *Algorithms for clustering data*, volume 6. Prentice hall Englewood Cliffs, 1988.

[11] Anthony D Josep, Randy Katz, Andy Konwinski, Lee Gunho, David Patterson, and Ariel Rabkin. A view of cloud computing. *Communications of the ACM*, 53(4), 2010.

[12] G Kalpana, Puligadda Veereswara Kumar, Shadi Aljawarneh, and RV Krishnaiah. Shifted adaption homomorphism encryption for mobile and cloud learning. *Computers & Electrical Engineering*, 65:178–195, 2018.

[13] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. An efficient K-Means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7):881–892, 2002.

[14] Jakub Konecný, H Brendan McMahan, Felix X Yu, Peter Richtarik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[15] Mascha Kurpicz, Anne-Cecile Orgerie, and Anita Sobe. How much does a vm cost? energy-proportional accounting in vm-based environments. In *2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, pages 651–658. IEEE, 2016.

[16] He Li, Kaoru Ota, and Mianxiong Dong. Learning iot in edge: deep learning for the internet of things with edge computing. *IEEE Network*, 32(1):96–101, 2018.

[17] Brendan McMahan and Daniel Ramage. Federated learning: Collaborative machine learning without centralized training data. *Google Research Blog*, 3, 2017.

[18] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, et al. Communication-efficient learning of deep networks from decentralized data. *arXiv preprint arXiv:1602.05629*, 2016.

[19] Peter Mell, Tim Grance, et al. The nist definition of cloud computing. 2011.

[20] HP Ng, SH Ong, KWC Foong, PS Goh, and WL Nowinski. Medical image segmentation using K-Means clustering and improved watershed algorithm. In *2006 IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 61–65. IEEE, 2006.

[21] Maciej Pawlik, Kamil Figiela, and Maciej Malawski. Performance evaluation of parallel cloud functions. 2018.

[22] Peng Gang Sun, Lin Gao, and Shan Han. Prediction of human disease-related gene clusters by clustering analysis. *International journal of biological sciences*, 7(1):61, 2011.

[23] Wolfgang Vogt and Dorothea Nagel. Cluster analysis in diagnosis. *Clinical Chemistry*, 38(2):182–198, 1992.

[24] Ellen M Voorhees and Narendra K Gupta. Facilitating world wide web searches utilizing a multiple search engine query clustering fusion strategy, January 26 1999. US Patent 5,864,845.

[25] William Voorsluys, James Broberg, Srikumar Venugopal, and Rajkumar Buyya. Cost of virtual machine live migration in clouds: A performance evaluation. In *IEEE International Conference on Cloud Computing*, pages 254–265. Springer, 2009.