Scalable Price Prediction Models of Hosting Business Leveraging Big Data with GPU

Samyuktha Muralidharan[1], Savita Yadav[1], Sanghoon Lee[2], and Jongwook Woo[1]*
[1]Department of Information Systems, California State University
Los Angeles 90032, USA
[2]Department of Electrical Engineering, Yonsei University
Seoul 03722, Korea
[e-mail: { smurali2, syadav5, jwoo5}@calstatela.edu, slee@yonsei.ac.kr]
*Corresponding author: Jongwook Woo

Abstract

There are many approaches to use Multiple GPUs to increase the computing power in business applications. We develop various models to predict the price of Airbnb listing using multiple GPUs in Big Data architecture. The Airbnb data set is about 2 GB, which the traditional systems has difficulty to process predictive analysis using machine learning algorithms. Using Big Data cluster in PySpark with resource handlings in YARN and RayDP, we implement and compare the regression models in Decision Tree, Random Forrest, GBT, XGBoost with both CPU and GPU. We measure the performance of the models with the accuracy and computing time. The experimental result shows that XGBoost model has more than 74% RMSE accuracy and the training computing time is up to 94 % faster than other models.

Keywords: Big Data, AI, Spark, RayDP, YARN, XGBoost, RAPIDS, Scalable Predictive Analysis

1. Introduction

Airbnb is an online marketplace for people who want to rent out their homes with people looking for accommodations in that locale.

The paper present regression models to predict the price of the Airbnb Listings. It helps the hosts to know if their property is suitable and to attract customers with the proper prices. The data consists of various Airbnb properties and features. The size of the data set is 1.93 Giga-Bytes. If implementing regression models with the data set greater than hundreds of Mega-Bytes using the traditional systems, it mostly generates a memory error or takes several hours or days to build models. Therefore, we adopt Big Data clusters with Hadoop Spark YARN and Ray that is distributed parallel computing systems and that can store the data set, and build prediction models much faster. It is also linearly scalable to afford bigger data and more computing power by adding more nodes. Additionally, we add GPU chip to the nodes of the clusters in order to accelerate the parallel computing power.

The paper is composed of Section 2 Related Work, Section 3 Distributed Parallel Computing Systems, Section 4 Big Data Architecture and Experimental Results, then finally, Section 5 Conclusion.

2. Related Work

Mitchell, *et al.* implemented an efficient GPU-accelerated XGBoost library with GPU for Higgs dataset containing 10 million instances and 28 features entirely within GPU memory [3]. Kalehbasti, *et al.* developed the models for predicting the Airbnb prices for NYC. The model is built with linear regression, SVR, and neural networks along with feature importance analyses [4]. But, we use Apache Spark and RayDP Big Data clusters to build models including XGBoost. Our approach is linearly scalable to handle massive data sets.

This research was supported by a research credit from Databricks University Alliance

Yadav *et al.* implemented classification models to predict ratings of Airbnb listings using legacy and XGBoost classification algorithms in Hadoop Spark Big Data cluster [2]. Our paper is to build regression models, and we process the data in RayDP cluster as well as in Spark cluster.

3. Distributed Computing Systems

We adopt two distributed parallel computing systems to implement regression models: Hadoop Spark and Ray clusters.

The distributed computing systems has been adopted to increase the computing power with multiple nodes in Data Science and AI. It also utilizes chip-level parallel computing with GPU and leverage the existing computing power of the distributed systems.

3.1 Hadoop Spark Cluster

Hadoop has lead Big Data community since 2006. It is a distributed parallel computing systems to store and process large scale data set with MapReduce computing engine. Spark is in-memory computing engine and built by AMPLab at UC Berkeley in 2012. Spark is 100 times faster than Hadoop MapReduce in theory. Thus, Hadoop community has integrated it into Hadoop cluster. Besides, Spark provides machine learning algorithms and XGBoost. XGBoost model can be built with Rapids and it has much better accuracy and faster computing time with parallel processing even utilizing GPU. Hadoop uses YARN for resource management.

Table 1. Computing Time for Data Reading and Engineering

File Systems	Data Reading (sec)	Data Engineering (sec)
EMR HDFS	5.4	19.9
EMR S3	11.1	22.8
RayDP Linux	18.8	37.5

3.2 Ray Cluster

Ion Stoica at UC Berkley launched RISELab as the successor to AMPLab in 2017. The lab presents Ray open source project as a distributed parallel computing systems. Ray's actor can utilize GPU. Placement Group of Ray is to schedule resources based on gang-scheduling.

RayDP is one of Ray's ecosystems and combine Ray cluster with Spark to prcess large scale data set. It provides XGBoost and deep learning algorithms with Tensorflow and PyTorch.

4. Architecture and Experiments

We implement machine learning models using Decision Tree (DT), Random Forest (RF), Gradient Boost Tree (GBT), and XGBoost to predict price of the Airbnb listings in Spark ML with Spark YARN and RayDP Placement Group.

4.1 Data Reading and Engineering

We collect Airbnb listings data set from Kaggle and OpenDataSoft, which is of 1.93 GB [1]. Our experimental clusters are built in Spark EMR and RayDP on Amazon AWS cloud. The cluster is composed of 3 nodes as g4dn.2xlarge instances. The hardware specification of the instance is: 8 vCPUs, 32 GiB memory, 25 GB Network performance. The instance has one NVIDIA T4 Tensor Core GPU that has 16 GB memory, 320 Turing Tensorflow cores and 2,560 NVidia CUDA cores.

Table 1 shows the computing time for Data Reading to read data from the file systems and for Data Engineering to transform the data to the feature columns. In Data Engineering, we select Price column as label. Then, we preprocess the data to remove outliers and handle null values. We split the dataset into a 70:30 ratio for training and testing data.

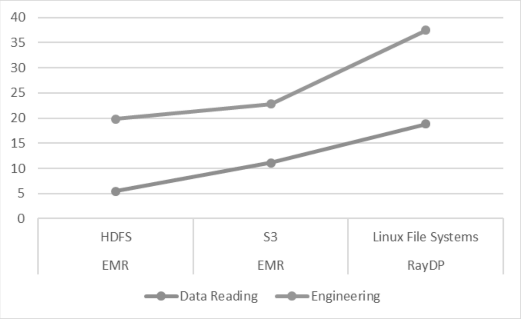

Fig. 1. Computing Time (seconds) for Data Reading and Engineering

We defined a pipeline that consists of a String Indexer, Vector Indexer, MinMax Scaler, and Vector Assembler. It is executed at HDFS and S3 of EMR and Linux file system (EXT4) of RayDP. The experimental result in the table 1 shows that HDFS has the highest performance.

Fig. 1 shows that Data Reading in HDFS is 51% and 71% faster than S3 and EXT4, respectively. For Data Engineering, HDFS is 13% and 47% faster than S3 and EXT4.

4.2 Computing Time for Modeling

We develop models with CPU and GPU in PySpark: DT, RF, GBT, and XGBoost. XGBoost in RayDP skips some data engineering process, which does not reduce the accuracy.
The essential metrics to evaluate the models are RMSE. Additionally, we measure the computing time to build models in multiple nodes with GPU chip for parallel computing. We also measure R2, the coefficient of determination.

Table 2. Performance of the models for price Prediction

Cluster	Models	Training Time (sec)	RMSE	R2
HDFS	XGBoost GPU	15.4	33.42	0.731
S3	DT	31.1	130.4	0.238
S3	GBT	173.2	128.7	0.248
S3	RF	48.2	130.1	0.247
S3	XGBoost CPU	21.5	33.4	0.731
RayDP EXT4	XGBoost CPU	17.9	31.1	NA

Table 2 shows the perforamance of all models. XGBoost models have RMSEs about 31 ~ 33 while other traditional models have 129 ~ 134, which predicts the price over 74 % higher accuracy. For the computing time to build a model, XGBoost with GPU in EMR S3 is 17 % faster than only with CPU, and 43 – 90 % faster than DT, RF, and GBT. XGBoost in HDFS is 14 % faster than in S3. The computing time in the RayDP is 4.3 seconds by skipping feature transformation steps.

5. Conclusions

In this paper, we implement and compare regression models to predict price of Airbnb listings using Big Data and RayDP clusters on distributed parallel computing. The data set is about 1.93 GB that causes days of computing time, and memory issue with the traditional computing systems for regression model. The Big Data, distributed parallel computing systems, can address the issues. Thus, we implement XGBoost models on the clusters that can run both CPU and GPU chip to enhance chip-level parallel computing. The measurement of the comparison is accuracy and computing time for data engineering, and for building a model. We observe that XGBoost models have 74 % higher accuracy and up to 90 % faster than the traditional models. XGBoost using GPU is 17 % faster than using CPU. Furthermore, Hadoop Spark cluster is up to 74 % faster when building models.

References

[1] Airbnb Listings Dataset (2022). Retrieved from https://public.opendatasoft.com/explore/dataset/airbnb-listings/table/?disjunctive.host_verifications&disjunctive.amenities&disjunctive.features

[2] Savita Yadav, Samyuktha Muralidharan, Sanghoon Lee, Jongwook Woo, "Scalable Predictive Analysis for Airbnb Listing Rating", KSII The 13th International Conference on Internet (ICONI) 2021, Dec 12-14 2021, Jeju Island, Korea, pp370-372, ISSN 2093-0542

[3] Mitchell, R., & Frank, E. (2017). Accelerating the XGBoost algorithm using GPU computing. PeerJ Computer Science, 3, e127. Retrieved from https://doi.org/10.7717/peerj-cs.127

[4] Kalehbasti, P. R., Nikolenko, L., & Rezaei, H. (2019). Airbnb Price Prediction Using Machine Learning and Sentiment Analysis. ArXiv. Published. https://arxiv.org/pdf/1907.12665.pdf

[5] Choudhary, P., Jain, A., & Baijal, R. (2018). Unravelling Airbnb Predicting Price for New Listing. ArXiv. Published. https://arxiv.org/pdf/1805.12101.pdf

[6] RAY, https://docs.ray.io/en/latest/ray-core/key-concepts.html

[7] RayDP: Using Spark on Ray, https://docs.ray.io/en/latest/data/raydp.html