Energy Efficiency Optimizing Based on Characteristics of Machine Learning in Cloud Computing

Xiao-Bo CAI[1,a], Yuan-Xia JI[2,b], and Ke HAN[3, *]

[1]*School of basic and Information Engineering, Yunnan Agricultural University, Kunming, China*
[2]*Yunnan Agricultural University library, Kunming, China*
[3]*School of Electrical Engineering, Kunming Metallurgy College, Kunming, China*
[a] *xbcai918@qq.com,* [b]*253667802@qq.com*
 * *Corresponding author: hanke81@qq.com*

Abstract: Energy efficiency is one of the most important issues for large-scale server systems in current cloud computing. the main method about the power-performance tradeoff by fixing one factor and minimizing the other, from the perspective of optimal load distribution. However, there still exist several main challenges about Energy efficiency due to the complexities of real cloud computing application scene. The paper adopts machine learning theory to save energy consumption by decrease redundant computation for high energy-efficiency cloud computing environment. give the typical k-means and Page Rank applications, the Experiments show that the presented algorithm can save power consumption apparently. The research combines the machine learning theory and distributed technology, and presents a creative way to challenged problems in energy-efficiency cloud.

1 Introduction

With the rapid increase in the scale and the number of the data centers, the data center energy consumption is growing rapidly. The US Environmental Protection Agency reported that in 2011, the world's data center consumed 18.8 billion kwh of electricity in 2010, about $1.1\% \sim 1.5\%$ of the global total power generation [1]. The green data center has become one of the hot spots. Currently, the data center energy management mainly uses DVS / DVFS or sleep / wake technology, which make the idle nodes with the low energy consumption state [2-4]. The energy saving for data center is mainly on Map Reduce computing tasks. Although these algorithms can reduce the energy consumption on certain extent, but these algorithms are subject to certain restrictions which did not take account of the typical application (machine learning task) running in the data center.

Firstly, we analyze the typical machine learning algorithms to saving the energy consumption of the data center. the machine learning tasks are computationally intensive applications, the energy consumption is mainly in the data analysis and calculation[6-8] .However, these algorithms need to continually classify and analyze the data in the process of machine learning, where there is a lot of redundancy in the process, which leads to unnecessary energy consumption[9-12]. Based on above, we have designed and implemented an energy-saving mechanism for machine learning, the method achieve energy-saving by matching the input data.

The MapReduce Computing Framework is a widely used as a programming model in the data center; there is a lot of work to focus on the study energy consumption and specific control methods of MapReduce model. MapReduce framework is generally divided into two parts: one is distributed storage, the other is distributed computing. The storage node is divided into two parts: one is the hot spot and the other is the non-hot spot which in a low-power state [13]. The different types of data stored in different areas. The effects of different configuration parameters on energy consumption in MapReduce are studied and they also provided enterprise-based benchmarks for MapReduce's energy consumption [14]. running a computational task should use all of the compute nodes and turn off all the compute nodes while the task is completed, which is better energy savings compared to only use one part of the compute nodes[15]. Reducing energy consumption by adjusting or controlling the physical location of virtual machine[16]. The data compression method is given to reduce the system's energy consumption [17]. Using DVFS technology to save energy for computing-intensive applications [18-19]. Different scheduling mechanisms for heterogeneous cluster are proposed, which can enable low power consumption without seriously affecting system throughput [20-21].

This paper Section I describes the related work. Section II analyzes the data center characteristics of energy consumption and the machine learning algorithm. Section III introduces the proposed energy saving algorithm. Section IV shows experimental data and correlation analysis.

2 The Core Idea

The accurate calculation results are not needed usually in the distributed computing by machine learning-oriented algorithm. Such as clustering algorithm, the recommended algorithms and relevance of different users of different commodities, and the last ranking of each point in PageRank, which do not require accurate calculation results. The user allows these calculations have errors within a range. the random initialization make the different results with the same data set. In fact, some programs cannot guarantee the error range of calculation process as k-means, PageRank and proposed algorithm. Usually users tend to specify a convergence value, when the value is smaller than the set value, the calculation is judged to be convergent. The machine learning for the user on the results have a lot of fault tolerance.

This paper proposes a policy to achieve the purpose of removing redundant calculations and saving energy consumption by compare the calculated data to determine whether the previous calculation results can be reused. At the same time, the mean error of the calculation results is controlled to a certain extent. In this paper, the the key to the question is to judge the calculation may be redundant calculation. We proposed input data matching degree, input / output correlation and other basic concept concepts.

2.1 Match Degree

In this paper, the twice input data match degree is the similarity degree between two data input and output results . If the two calculations are exactly the same, then the two input exactly match.

2.2 The Correlation between Input and Output

We use correlation to describe the effect of input data changes on the calculated output. Machine learning in the input data can be divided into multiple independent input, their correlation with the calculation results can be individually.

2.3 The Input Matching Module

The module is mainly used to calculate the data matching degree between the input of this task and the before task. This module save d the previous calculation results, the corresponding input value and the the correlation vector.In order to determine whether this calculation is redundant, the system needs to call the Match function to compare the similarity between the calculation and the previous calculation. When the Match function returns true, it means that the calculation can be considered redundant. And the current First, system constructs the difference vector by calling the user-defined Diff function which calculates the difference of the input vector. Then, calculate the vector inner product by the difference degree vector and relevance vector. Finally, the matching degree is compared with the threshold set by the user. If the matching degree is less than the threshold, the matching is

successful. TaskTracker sent the output data address directly to the JobTracker, and notify the energy consumption module matching success.

When the TaskTracker receives the calculated input data, it passes the input data to the input matching module first. If a matching input is found, the stored calculation result is output and the node is set to a low power state. If no match the input, then call the calculation module for Map or Reduce calculation. The following describes the specific design and implementation of each module.

3 Matching Algorithm Design

3.1 System Design and Implementation

We use the correlation to describe the influence of the change of the input data to calculate the output and assign the task According to the situation of energy consumption of the current node. Task Tracker contains the input matching module, the energy consumption module and computing module.

3.2 Task Input

All input data that requires input matching must be abstracted as an input vector, such as most of the machine learning algorithms are based matrices, and each row of the matrix is seen as a dimension of the vector. In this system, we provide a higher level of abstract (InputObj) interface to each input vector type for the user. The users can complete the interface as needed. Such as in k-means, the data in the local data block will not change, we only store and compare the cluster center point coordinates in the process of input matching. Therefore, the input vector should be center Point of all the cluster collection, and clustering center point need to complete the InputObj interface. For the application which have large amount of data or multiple input changes, the user should extract characteristic value or calculation of hash of the input data to a reasonable abstraction to reduce the amount of data storage cache and input matching the amount of calculation.

3.3 Save the Results

It is necessary to the previous calculation results are cached locally is necessary to remove the redundancy calculation. We need to consider the MapTask calculation results 'cache problem in Hadoop, the Reduce Task data stored in the distributed file system by TaskTracker .TaskTracker store the MapTask calculation results in the local directory. When the calculation task of MapTask is completed, the MapTask calculation result will be deleted immediately. But in our system, TaskTracker save the calculation result' output path of input path, When the input matches a stored input, the TaskTracker sends the corresponding calculation result path to the JobTracker. The user can also clear the cached intermediate calculation results manually. The basic algorithm of input matching is as flowing.

```
bool Match(vector<ObjInput> v1,vector
<ObjInput>v2,vector<double>rel)
Vector<double>diff=rel.clone()
For i in 1,…,n
diff[i]=Diff(v1[i],v2[i])
double d=Σdiff[i]*rel[i]
return(d<threhold)
```

4 Experiments and Analysis

We use the power socket to test the configuration of the Intel i7-4770 processor, 32GB memory and 2TB hard drive single-node state of the energy consumption. Then we randomly select n computing points as the end of the connection. We use the custom data generation program to generate 400K data points (205MB), where the average number of connections per point is 20 and the maximum number of iterations is set to 10. First, we conducted an evaluation test on the application of energy consumption.

Energy Efficiency Evaluation

This section describes the energy consumption evaluation results. the input difference threshold is set 0.05 for k-means and PageRank applications.

Currently, the system is for the isomorphic data center, so the power of all nodes can be considered similar, the energy of all nodes of the MapTask Get the power consumed by the Map phase, and then use the power outlet to obtain the real-time power of the ReduceTask node, that is, the power consumed by the Reduce stage, and then the total power consumption of the single iteration is obtained by accumulating. Similarity, the total energy consumption of the whole calculation can be multiplied by the number of iterations by the energy consumed by a single iteration. For the optimized power calculation, we consider two kinds of processing methods respectively.

Free energy-saving method Mode is simply to free the Map Task node that processes the redundant data blocks, the original MapTask used to calculate the time will become idle time, the power calculation can be calculated by the original time Power is replaced by the power of a single node in the idle state. Considering that the data to be processed at the Shuffle stage is the same, its power and the power consumption can be considered the same. Here is the use of alternative methods to estimate the savings of electricity, through the same method of calculation the total energy consumption of the application;

The other is to use the dormant way to deal with the original Map Task calculation time, that is, by dormant way to deal with redundant computing, through Reduce the data request of the Reduce Task or the job assignment request of the Job Tracker to wake up the node, which we call the dormant energy-saving party

In this case, the power of the node is further reduced, but the corresponding hardware support is needed, and the total power is calculated in the same way as the former similarly, the power consumption can be estimated by replacing the original calculated time power with the power in the dormant state.

Finally, by calculating the power consumption of all MapTask and the power consumption of ReduceTask for each iteration, we can get each iteration Figure 1 show the comparison of the power before and after the optimization of k-means and PageRank, and it can be seen that in the k-means application, we also use a simple method based on idle energy saving can achieve 10% of the energy savings, and through the sleep-based energy-saving methods, the original program in 25% of the consumption of electricity can be saved;

Figure 2 shows the case where the k-means applies 20 iterations of total power consumption. Figure 20 shows the case where PageRank applies 10 iterations of total power consumption, where the free energy saving method can save around 5% for k-means applications Power consumption, and through the hibernation-based energy-saving method, nearly 23% of the total k-means application can be saved; for PageRank applications, based on the idle energy-saving method can save 5% of the total power consumption, Energy-saving methods, nearly 20% of the total energy consumption in the entire PageRank application can be saved.

In this paper, we use machine learning algorithm to save the energy consumption of data center .the algorithm eliminate the unnecessary redundancy calculation by matching input and calculation result. This paper analyzes the energy consumption and accuracy. The experimental results show that Error control in a certain range of the remise, the system effectively saves the energy consumption of the system. The next step we will work focus on the system in the heterogeneous environment of the work.

Acknowledgment

References

1. Beloglazov A, Buyya R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers[J]. Concurrency and Computation: Practice and Experience, 2012, 24(13): 1397-1420.
2. Wang XR, Wang YF. Coordinating power control and performance management for virtualized server clusters. IEEE Trans. on
3. Parallel and Distributed Systems, 2011,22(2):245−259. [doi: 10.1109/TPDS.2010.91]
4. Tam D. Facebook processes more than 500 TB of data daily. 2012.
5. http://www.cnet.com/news/facebook-processes-more-than-500-tb-of-data-daily/
6. Low Y, Gonzalez J, Kyrola A, Bickson D, Guestrin C, Hellerstein JM. Graphlab: A new framework for parallel machine learning.
7. arXiv preprint arXiv:1006.4990. 2010.

8. Wang YF, Wang XR, Chen M, Zhu XY. Partic: Power-Aware response time control for virtualized web servers. IEEE Trans. on

9. Parallel and Distributed Systems, 2011,22(2):323−336. [doi: 10.1109/TPDS.2010.79]

10. Qian ZP, Chen XW, Kang NX, Chen MC, Yu Y, Moscibroda T, Zhang Z. MadLINQ: Large-Scale distributed matrix computation

11. for the cloud. In: Proc. of the 7th ACM European Conf. on Computer Systems. ACM Press, 2012. 197−210. [doi: 10.1145/2168836.

12. Williams C. What is a green data center. 2011. http://www.mnn.com/green-tech/computers/stories/what-is-a-green-data-center

13. Kliazovich D, Bouvry P, Khan SU. DENS: Data center energy-efficient network-aware scheduling. Cluster Computing, 2013,16(1):

14. 65−75. [doi: 10.1007/s10586-011-0177-4]

15. Koomey J. Growth in data center electricity use 2005 to 2010. 2011. http://www.analyticspress.com/datacenters.html

16. Zhou L, Li J, Li F, et al. Energy consumption model and energy efficiency of machine tools: a comprehensive literature review[J]. Journal of Cleaner Production, 2016, 112:3721–3734.

17. Yuan L, Zhan JF, Sang B, Wang L, Wang HN. PowerTracer: Tracing requests in multi-tier services to save cluster power

18. consumption. Technical Report, Institute of Computing Technoly, Chinese Academy of Sciences, 2010.

19. Chase JS, Anderson DC, Thackar PN, Vahdat AM, Doyle RP. Managing energy and server resources in hosting centers. In: Proc. of

20. the 18th Symp. on Operating Systems Principles. 2001. [doi: 10.1145/502034.502045]

21. Chen G, He WB, Liu J, Nath S, Rigas L, Xiao L, Zhao F. Energy-Aware server provisioning and load dispatching for

22. connectionintensive Internet services. In: Proc. of the NSDI. 2008. 337−350. http://dl.acm.org/citation.cfm?id=1387613

23. Chen Y H, Krishna T, Emer J S, et al. Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks[J]. IEEE Journal of Solid-State Circuits, 2017, 52(1):1-12.

24. Demaine E D, Lynch J, Mirano G J, et al. Energy-Efficient Algorithms[J]. 2016:321-332.

25. Lu T, Chen MH, Andrew LL. Simple and effective dynamic provisioning for power-proportional data centers. IEEE Trans. On Parallel and Distributed Systems, 2013,24(6):1161−1171. [doi: 10.1109/TPDS.2012.241]

26. Chen YP, Ganapathi AS, Fox A, Katz RH, Patterson DA. Statistical workloads for energy efficient mapreduce. Technical Report,

27. No. UCB/EECS-2010-6, University of California at Berkeley, 2010.

28. Lang W, Patel JM. Energy management for mapreduce clusters. Proc. of the VLDB Endowment, 2010,3(1-2):129−139.

29. Yao Y, Huang L, Sharma A, et al. Data centers power reduction: A two time scale approach for delay tolerant workloads[C]//INFOCOM, 2012 Proceedings IEEE. IEEE, 2012: 1431-1439.

30. Ye K, Huang D, Jiang X, et al. Virtual machine based energy-efficient data center architecture for cloud computing: a performance perspective[C]//Proceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing. IEEE Computer Society, 2010: 171-178.

31. Pedram M. Energy-efficient datacenters[J]. Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on, 2012, 31(10): 1465-1484.

32. Gao P X, Curtis A R, Wong B, et al. It's not easy being green[J]. ACM SIGCOMM Computer Communication Review, 2012, 42(4): 211-222.
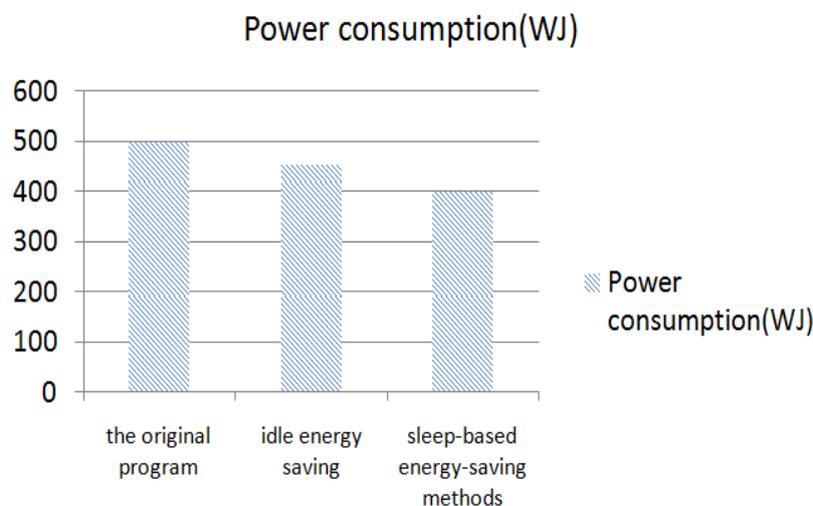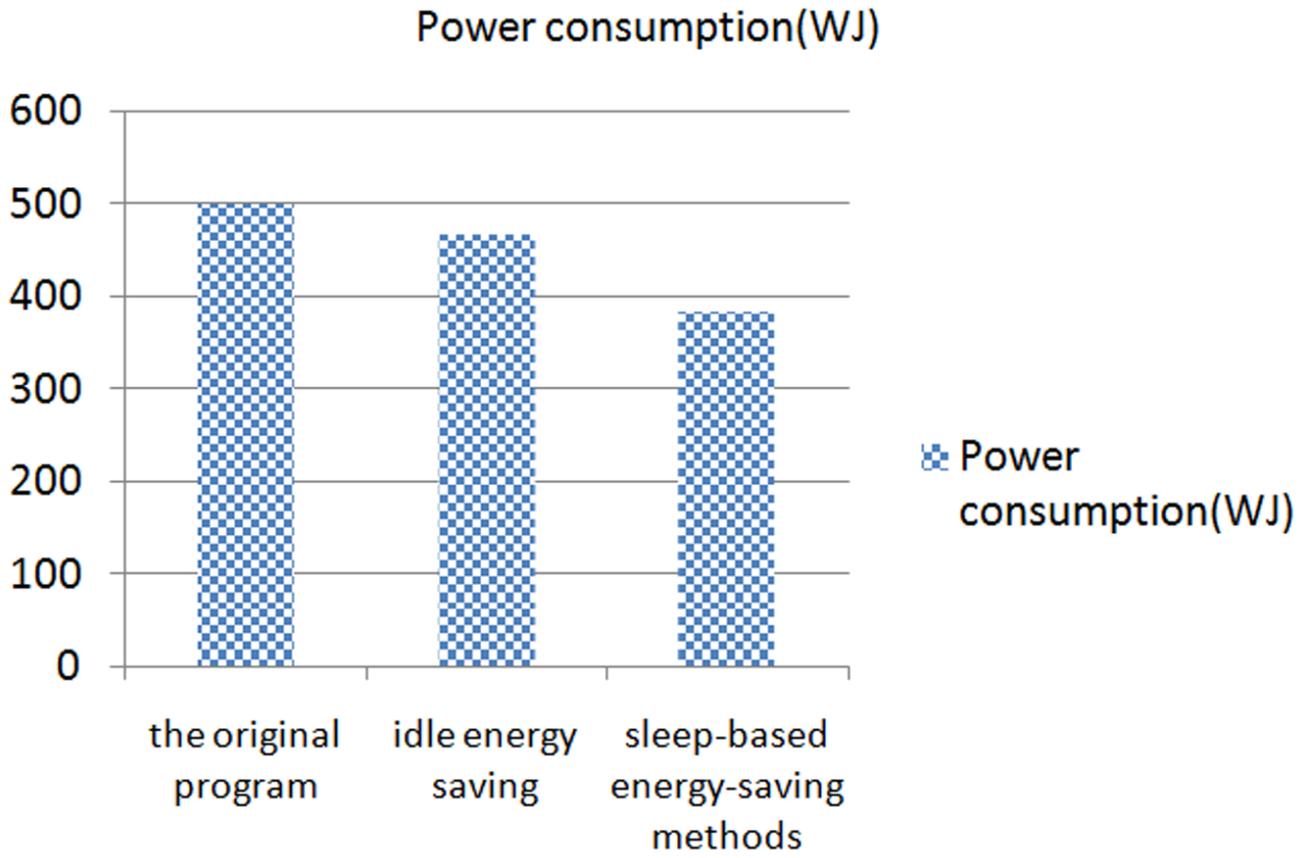
Figure 1. Total energy consumption of k-means

Figure 2. Total energy consumption of PageRank