A Machine Learning Based Approach for computation offloading in Mobile Edge Computing

S.Sindhura,V.Sindhu, R.Ezhilarasie

1School of Computing, SASTRA DeemedUniversity, Thanjavur

Abstract: In the modern world, mobile devices play a vital role in daily life. Though the technology is developing day by day, mobile devices are still resource constrained (of low potential). Offloading techniques should be used strategically to ensure the optimum utilization of the mobile resources. This paper presents you the study on feasibility of machine learning techniques in mobile offloading frameworks based on some constraints like execution time, CPU load, bandwidth etc. This algorithm predicts whether to do the computation in the local device or offload it to a peer server (which is not as big as cloud but has better resources than the local device) by using machine learning techniques that have high scheduling accuracy. Since there is a lot of congestion to the cloud and the communication cost is also a little high, a peer server is proposed to overcome this problem. By taking million instructions per second and bandwidth as constraints, offloading of an application to another device is done. This peer server, which acts as a cloud network, is not so far from the local server and has better resources when compared to the local network. In this way, more effective results of offloading can be obtained.

Keywords: Offloading, peer server, CPU load, machine learning,computation,cloud

I. INTRODUCTION

In this rapid growing world, technology is enhancing day by day. Every day, billions of users connect to the internet. People generally come across advanced versions of almost all the products in the market like that of augmented reality. In spite of these advancements, the devices are limited in some aspects. People want devices with high configurations especially high battery capacity and they will be happy if unlimited battery is provided but this is practically not possible as of now (future may hold even such surprises for us). At least an effective utilization of battery should be present. By this, it can be concluded that execution time should be reduced. An effective solution for this kind of problem is offloading.

Mobile Cloud Computing is a technology where in the mobile applications are not just restricted to mobile operating system and makes use of cloud technology for better utilization of resources.MCC overtakes the constraints that are present in the software and hardware components of smart devices by making use of cloud .Offloading to cloud is the standard solution. As technology is improving

faster many users are now using internet. Due to this there are many more connections added to the cloud. Not only that, even transmission time and the cloud rent should be taken into consideration. So an alternative way to this can be suggested by introducing intermediate servers that is Edge computing.Important thing to be discussed here is about edge computing device. It can be a nearby PC or any other device which have better resources thaocal devices. By offloading to these servers, the transmission time and the cloud rent will decrease and the computation will also take place quickly. By this, the traffic to the cloud can also be controlled.

So a conclusion is drawn to offload data to the intermediate server. But the question which stands here is, when to offload to the intermediate server and when to run it in the local device? To make this decision, MachineLearning tells whether offloading is better or execution in local device is better. Machine Learning, commonly used word now a days. What is it actually? Machine learning technique is a branch of artificial intelligence (AI). By means of this, a system can learn from previous data and adapt to unseen situations dynamically. A machine learning algorithm is used to predict what to do based on the previous knowledge.

II. Related Works

'Mobile Cloud Computing' also called MCC is combined system comprising of both cloud and mobile environments. It uses applications on local devices that require extensive processing mechanism[4]. Execution in remote systems increases the computation speed of the server. Smart Mobile Devices (SMDs) have less capacity of computing as they are generally limited in memory, speed of processor and lifetime of battery. Therefore, Mobile Cloud Computing (MCC) can take advantage of using computational offloading for enabling intensive mobile applications on SMDs14] .

So, to endorse this problem of load balancing, an adaptive mobile resource management (AMRM) protocol, which does not depend on cloud, was designed and implemented by which delay time is also reduced. The improved job processing speed can be explained based on AMRM and by taking into consideration the idle resources of the mobile devices[10].

Intake of Energy and latency of the system can be decreased to a great extent by means of computation offloading. Based on activities done by the user, device framework decides whether offloading is needed for the computation at the cloud. When the resources or parameters in the device goes beyond the capacity of the device, then it sends the data to the cloud server by making use of sockets that are present in python[11].

Issues that are found to be important are discussed when distributed computing is done on a personal device environment. Such issues are algorithm granularity, differences between speed and cost of different communication protocols, resource usage, and network topology issues, requirements to the platforms as well as security and trust. It is also possible to perform distributed computing in the mobile phone environment [5].

III. Problem Formulation:

Considering Million Instructions per Second (MIPS) as major factor for processing tasks, a classification is done on the outcome of processed data as time efficient, energy efficient and cost efficient. For each model, best possible configuration has to be selected to execute that task by means of machine learning algorithm.

In time efficient model, MIPS and bandwidth are taken as input and execution time taken by the task in each configuration has to be predicted to get the best suitable configuration for obtaining time efficiency. Coming to cost efficient model, prediction of the execution time and cost per unit time has to be done for a given task and the best efficient configuration to finish it using less cost has to be found out. Lastly, for energy efficient model, the execution time and power consumed per unit time and the configuration that is efficient in terms of power has to be computed. A proper machine learning algorithm has to be used to get the most efficient output.

The algorithm is implemented in a peer infrastructure. It has two major components: a mobile device that is, our local device, and a remote device for the applications to be offloaded. The main objective behind using this is the optimization in the execution of task and to make the application run as quick as possible. To attain our goal, the main requirements that are to be considered are Million Instructions Per Second (MIPS), Bandwidth, Time, Cost and Power.

This problem can be classified based on three major constraints Time, Cost and Power. Based on user‘s means of efficiency, best suitable or appropriate configuration system has to be taken.

A. Time Efficient algorithm

Inputs:

1. MIPS:

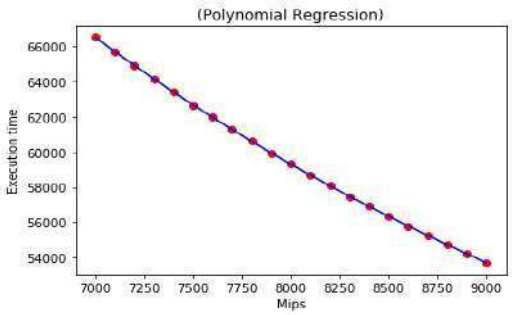
MIPS measure randomly the number of instructions that a computer can execute in one second. It generally corresponds to the speed of CPU. By taking it as an input a faster and efficient execution of the processes can be obtained. If the MIPS of the system is more, the lesser will be the time of execution.

2. Bandwidth:

It corresponds to the distance between the local system and remote system .Remote system is nothing but peer system. If bandwidth is more, it takes more time to transmit data from our mobile device to the peer system. But if the configuration is very high, distance doesn‘t matter much. So, it will also make changes in finding the execution time of a task.

Output:Execution time:

Based on MIPS of the system and bandwidth, the system which has the best configuration for executing a given task is considered and then we can find time efficient system. For the model that is time efficient, the predictions of execution time as per the MIPS value is found through Regression model and the graph between them is obtained as follows:



MIPS	Bandwidth(in kbps)	Execution Time(in ms)
7500	500	66741.844
7800	900	62613.051
8300	400	62269.895
8600	700	58924.483
8900	200	64999.041

Execution Time Vs MIPS

B.Energy efficient algorithm

Input: MIPS

Output:

Power consumed per unit time:To get an optimization with respect to energy, power consumption plays a vital role. Based on speed of the processor, power consumed by the system varies. The relationship between power and MIPS is almost of the following form:Power = MIPS^2/10^8

From power, energy is obtained .so predicting the value of power can find a way for energy efficient algorithm

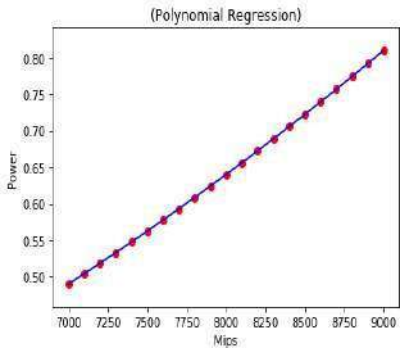
Execution Time

Similar to the first model, the execution time of a task in a system can also be found.Energy consumed = power consumed per unit time * time taken

By using that relationship, the predicted energy values of a task or an application can be found out and thus most effective configuration is found out to get the maximum efficiency and consistency in the problem. For the algorithm we got the execution time for the MIPS values as per the below values: For the model that wants the results to be Energy efficient, the output is obtained as followed by the Regression model as per the predictions. Graph is as follows:

Table of MIPS , power and Exec time

MIPS	Execution Time(in ms)	Power(in mWatts)	Energy(in
7500	62674.574	0.5625	35254.448
7800	60601.504	0.6084	36869.955
8300	57487.148	0.6889	39602.896
8600	55795.709	0.7396	41266.507
8900	54211.481	0.7921	42940.914



MIPS vs Power

C. Cost efficient algorithm

Input: MIPS

Outputs:

1. Cost per unit time

Based on the speed of processor and the specifications or features of the system, cost factor varies continuously. Taking them into consideration, cost per unit time for processing a given task arefound out.

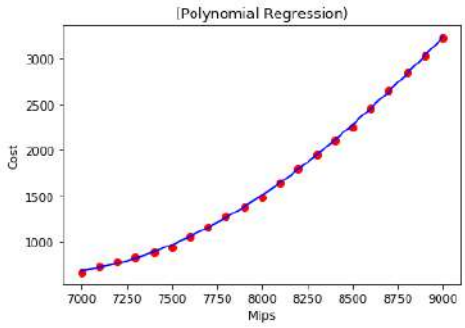
2. Execution time

From the first model, the execution time of the task is obtained in the following way:

Cost = Cost per unit time * Execution time

The relationship between power and MIPS is described earlier. By calculating time needed for given task, the Energy consumed by each of the given task or application can be known and best suitable configuration for that particular process can be estimated. Some of the predicted values are given below: Finally for the cost efficient model, the predictions are made in the same way and the finest outcome is considered. The value of MIPS and cost per unit time varies as follows:

Some of the predictions of the values of cost efficient model are explained in the following table:



MIPS Vs cost graph

MIPS	Cost(in paise)
7500	815.089
7800	1042.087
8300	1621.020
8600	2075.984
8900	2602.163

MIPS vs cost

For our problem statement, all the possible machine learning algorithms are used. The best suitable one for the above models is Polynomial Regression. A custom value for cost is taken and it is related to MIPS. Cost doesn‘t linearly vary with respect to MIPS. By using peer systems, the optimized outcome according to the type of model we require can be achieved. Let us consider some random values for inputs and verify how the values changes in outputs. The observations are taken down in the table below:

3 different observations are considered for a detailed study of how the values are changing with respect to the given inputs.

For Time, as MIPS value increases that is as the speed of processor increases, the time required to run the task gradually decreases. It is not a linear decrease but it is a polynomial decrease.

Coming to the Energy, it varies both on Execution time and also the power. From the graph, it was clear that Power consumption increases as the value of MIPS increases linearly. Execution time varies with MIPS polynomially as per the first graph. Energy is obtained as the product of Power and Time. If Power is more and MIPS is less, then the value of time is more and as a result, Energy consumed is also more.

If MIPS is very large and Power is less, then time consumed is less and the Energy consumed is also less comparatively.

3. For Cost, the value again depends on MIPS. As the value of MIPS increases, Cost also increases. They are proportional to each other. They are not varying linearly but changes polynomially . From the third graph, we can infer that .

Coming to preferences, the order of tasks in all the three models are as follows:Time – 3 1 2

Energy – 2 1 3

Cost – 1 2 3 .Let us consider some other observations

Table 4 MIPS, Energy and Cost Table 5 MIPS, Bandwidth and Execution time

From the values it can be inferred that

If MIPS is too low and bandwidth is too high , Time is too high.

If MIPS is too low and bandwidth is also low, then least amount of energy is consumed.

If MIPS is too slow and bandwidth is also low ,then cost is very high that is maximum amount of cost has to be kept.

So, the constraints have to be taken based on the type of model in which we want optimization.

MIPS	Energy(in microJoule)	Cost(in paise)
9876	48248.941	4698.0780
4567	19187.817	5332.631
12345	49497.245	9031.007
9654	47090.191	4181.248
3667	13908.931	9802.382
7111	33179.089	671.562
6533	30106.585	812.941
9843	48079.887	4620.23

Task no.	MIPS	Bandwidth(in kbps)	Execution time(in ms)
1	9876	567	53422.871
2	4567	998	101788.403
3	12345	765	35581.5411
4	9654	890	52264.255
5	3667	345	109674.704
6	7111	254	73768.792
7	6533	465	74805.036
8	9843	312	56456.755

IV. Results and Discussion

As of now implementation of framework in mobile device and creation of an application to test our algorithm is done. Now inputs can be given accordingly and get corresponding outputs. According to the outputs, offloading it to a remote machine or execution in the local device itself has to be done. If offloading of data is done, data has to be transmitted to the suitable system configuration as predicted

by the algorithm in an efficient and optimized manner and get the results as per the application that is created.

For time sensitive applications, the system that completes the task in least possible time with all the available resources is taken. Similarly, as far as cost is concerned, the best applicable configuration that gives results for legible cost of transmission is taken into account. And for the efficiency in case of energy consumption, the configuration of the system that involves less energy intake and faster execution is most desirable one.Total time for execution = 2 * Time for Transmission + Execution time at peer server.Time for transmission is proportional to the bandwidth through which the data is being propagated. It is also difficult to provide long battery lifetimes with more clock speed but it is not a factor as per our problem.The comparative approach for deciding the optimization regarding the processing of tasks or applications will help us to get more and more efficiency in achieving outputs.

V. Conclusion

In this paper, a machine learning based runtime application for mobile transferring framework is proposed. The idea of using peer systems brings out an effective result is actually very good. It is really difficult to run large files of data in mobile device which have less resources than that are needed to get optimized output. With the help of Machine learning algorithm, the constraints can be predicted and the results for all possible conditions and configurations can be found out. This way of finding the best suitable source for our inputs helps us in many ways. Performance of the task has increased to much larger extent by using this mechanism. All the necessary services are provided to get adequate results. The time taken to run the process is reduced to a larger extent. Applications are classified into 3 types to get a better offloading with respect to each and every constraint individually.

There is a hope that this experiment will inspire new ideas about innovative applications which are more appropriate for this kind of technology. A list of some of the research areas associated with offloading, and descriptions of some infrastructures and solutions that address these research areas is given. As a future work, some other constraints can be added and it can be made a more efficient one .This will really create a drastic change in the field of mobile computing devices . Then as many resources as possible can be used by the device as per the necessities.

References:

1. S M Azharul Karim, John J Prevost," A Machine Learning based Approach to Mobile Cloud Offloading", Computing Conference 2017 18-20 July 2017 | London, UK
2. Amal Ellouze, Maurice Gagnaire, Ahmed Haddad," A Mobile Application Offloading Algorithm for Mobile Cloud Computing", 2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering
3. Bowen Zhou, Amir Vahid Dastjerdi, Rodrigo N. Calheiros, Satish Narayana Srirama, and Rajkumar Buyya," A Context Sensitive Offloading Scheme for Mobile Cloud Computing Service", 2015 IEEE 8th International Conference on Cloud Computing
4. Vandana, Rajeev Kumar Bedi, Sunil Kumar Gupta," OFFLOADING TECHNIQUES IN MOBILE CLOUD COMPUTING: A REVIEW",Technical Research Organisation,India
5. Balázs Bakos, Szabolcs Fodor and Jukka K. Nurminen," Distributed computing with mobile phones: an experiment with Mersenne prime search", Nokia Research Center, P.O.Box 392, H-1461 BUDAPEST, Hungary and Nokia Research Center, P.O.Box 407, FIN-00045 NOKIA GROUP, Finland
6. Salvador Melendez and Michael P. McGarry," Computation Offloading Decisions for Reducing Completion Time", University of Texas at El Paso El Paso, TX 79968
7. https://www.researchgate.net/publication/273259885
8. Muhammad Shiraz·Abdullah Gani· Azra Shamim·Suleman Khan· Raja Wasim Ahmad," Energy Efficient Computational Offloading Framework for Mobile Cloud Computing", Centre for Mobile Cloud Computing Research (C4MCCR), Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia
9. Karthik Kumar·Jibang Liu·Yung-Hsiang Lu· Bharat Bhargava," A Survey of Computation Offloading for Mobile Systems", Bhargava Purdue University, West Lafayette, IN, USA
10. Byoungwook Kim , Hwirim Byun , Yoon-A Heo and Young-Sik Jeong ," Adaptive Job Load Balancing Scheme on Mobile Cloud Computing with Collaborative Architecture",
11. Heungsik Eom, Pierre St Juste, Renato Figueiredo, Omesh Tickoo, Ramesh Illikkal, Ravishankar Iyer," MachineLearning-basedRuntimeSchedulerforMobileOffloadingFramework", 2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing
12. Bowen Zhou," Code Offloading and Resource Management Algorithms for Heterogeneous Mobile Clouds",submitted for degree of Doctor of Philosophy, THE UNIVERSITY OF MELBOURNE
13. Pooja V. Bhokare Chitra J. Patil," Mobile Computation Dynamic Offloading using Cloud ", International Journal of Engineering Trends and Technology (IJETT) – Volume 8 Number 6- Feb 2014
14. Muhammad Shiraz·Abdullah Gani· Azra Shamim·Suleman Khan· Raja Wasim Ahmad," Energy Efficient Computational Offloading Framework for Mobile Cloud Computing", Centre for Mobile Cloud Computing Research (C4MCCR), Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia.
15. Radhika Loomba B.Tech.(Hons.)," A Colloborative Mobile Sensing Framework using Mobile Cloud Computing.