Efficient Classification Of Cardiotocogram Data By Improved Bi-Model Neural Network

Dr. A. Nirmal Kumar, Kamineni B.T. Sundari, A Mahalakshmi

Abstract: In this paper, an Elaborate literature survey and background study was made on existing clustering and classification algorithms and CTG data classification methods. An Evaluation on Automated Classification Methods for the Clustering and Classification of Cardiotocogram Data was made and the results has been presented. An Outlier Based Bi-Model Neural Network (BM-NN) based Classification System for Improved Classification of Cardiotocogram Data has been presented. A PCA based Improved Bi-Model Neural Network (IBM-NN) based Classification System for Improved Classification of Cardiotocogram Data has been presented.

Index Terms: Cloud, Virtual Data Centers, Openstack, Suspend resume policy.

1 INTRODUCTION

In this work, present the improved classification models which will consider outliers in the data and eliminate them from training phase of the classification process. This model is almost similar to the previous BM-NN Model. But here use the Eigen Vectors of the training data to reduce the dimension of the training data as well as testing data. The proposed idea also considerably improved the performance in classifying Normal, Suspicious and Pathologic CTG patterns[1].

2 LITERATURE SURVEY

Cloud Computing is a term that describes the means of delivering any and all information technology from computing power to compute infrastructure, applications, business process and personal collaboration to end users as service whenever and wherever they need it. Here IaaS is used and the energy consumption is optimized by using three power saving policies to reduce the idle power of the server. Here N policy is used, in that when server machine will be turned on, only when the number of jobs in the queue is more or same as that of the specified N threshold. The three policies used are ISN policy, SI policy and SN policy. In the ISN, policy the server will enter into the busy mode when the job arrives and it will end its busy mode only when it completes all of its jobs. The server will be in the sleep mode until the number of jobs in the queue id is less than that of the specified N threshold value. In the SN, the server will enter into the sleep mode only when there is no job in the specified machine, whereas in the SI policy, the server will be in the sleep mode for only a certain amount of time when the sleeping time of the server ends, it will enter into the idle mode.

- *Dr. A. Nirmal Kumar is currently working as Associate Professor in Department of Computer Science and Engineering, CMR Institute of Technology, Hyderabad, Telangana, India. E-mail: nirmalkumar@cmritonline.ac.in*
- *Kamineni B.T. Sundari is currently working as Assistant Professor in Department of Computer Science and Engineering, Kommuri Pratap Reddy Institute of Technology, Ghatkesar, Telangana, India. E-mail: kaminenibala@gmail.com*
- *A. Mahalakshmi is currently working as Assistant Professor in Department of Computer Science and Engineering, CMR Institute of Technology, Hyderabad, Telangana, India. E-mail: mahalakshmi.valliri09@gmail.com*

Thus, the consumption of idle power is efficiently reduced at the same this, thus more policies are used and in this the beginning cost will be high (Chiang et al. 2015). Here cloud as a service oriented platform dynamic allocation of resources is proposed by means of virtualization technology. In this technology, the resources are allocated dynamically to the system according to the need of the user. In order to allocate resources dynamically to the system and to determine the utilization of uneven multiple resources skewness is used. By using this skewness, the load between the VM is balanced. Thus the overloading between the VMs is prevented by effective load balancing between the machines and at the same time physical machines should be capable enough to handle all the resource needs of the VM, else overloading of the physical machine takes place and the VM performance is decreased (Nagpure et al. 2015). For the past few years", consumption of energy has become the most difficult part in computing environment especially their data centres. Here the energy efficiency is improved in the web servers by determining the daily requests patterns sent to the web servers which indicate the resource provisioning in offline and will be more suitable for management of resources. Here two load distribution algorithms are used. They are relative load distribution, servers are utilized equally and the adaptive load distribution, the nodes are selected based on the power level. Mostly these two algorithms are used to view the usage of the CPU and level of power consumption in the web servers. Thus, at the end, shutting down of the idle nodes will save the energy. It is however difficult to make out this offline methodology without the daily request patterns. Thus in future it should be enhanced in such a way that the workload predictions should be based on online resource management (Chen et al. 2015). The cloud computing has generally reshaped the cloud computing. Though the cloud provides more advanced features, still it is lagging in its operational cost because it is very high for both the public and the private clouds. Green computing is also very important with minimum resource and more demand. A framework is provided that provides enhancements in the cloud architecture. By using less number of VM, power aware scheduling techniques, proper resource management, and the cloud data center can be improved with minimum overhead. Thus, future enhancements should be done in such a way that all the fields that the proper resource management, scheduling

techniques should be enhanced (Younge et al. 2010). Data centers generally consume most amount of energy which is determined in terms of distribution of power and cooling of the systems. By dynamically adjusting the active machines, the resources can be efficiently used according to the system needs. Generally the data centers will have a wide variety of computer systems with varying energy consumption characteristics. Then, works performed by the system will be generally based on the priority, resource needed and the objective. Failure in any one of these characteristics will lead to a problem, thus here the heterogeneity aware resource management is presented for providing a dynamic work in the clouds. K means clustering algorithm is used for allocating the workload based on the similar characteristics in terms of requirements and resource. Here a novel technique is presented for adjusting the machines dynamically to minimize the power (Zhang et al. 2014). In the epoch of cloud computing, data centres consumes more power that can be reduced by obliterating the number of turned-on servers. A power aware virtual datacenter resource scheduling strategy is using this algorithm that consolidates servers periodically and adjusts the VM placements between consolidation rounds to honor a SLA. This methodology is used to construct and validate a power-aware datacenter simulator. Using this, the simulator performance of scheduling algorithm is analyzed. This strategy is more power-efficient than event-based. Further simulation results show that in general, VM weight adjustment according to workload can enable power-aware schedulers to perform more efficiently under the SLA. This approach improves data center control budget by 35% for accidental workloads resembling web-requests, and progress data center budget by 22.7% for workloads exhibiting stable reserve requirements including ScaLAPACK (Zu et al. 2013). Task consolidation in the cloud has become an important approach to streamline resource usage and in turn improve energy efficiency. Resource utilization directly relates to energy consumption, modeled their relationship and developed two energies-conscious task consolidation heuristics. The cost functions incorporated into these heuristics effectively capture energy-saving possibilities and their capability has been verified by evaluation. The results show that the study should not have only a direct impact on the reduction of electricity bills of cloud infrastructure providers with better resource provisioning in other operational costs (Lee & Zomaya 2012). Monitoring and controlling the behavior of Energy Management System (EMS) by fulfilling its purpose at the same time balancing certain constraints, including cost optimization, system reliability and environmental consideration. Examples of systems are building energy management, micro grid energy management and datacenter energy management. It shows promise in the improvement of energy savings across many domains. EMS is a proof of implementation in a wide variety of uses (Duggan & Young 2012). At Internet Data Centers (IDCs) electricity expense is becoming a major category of operational cost. Here problem addresses the total energy cost minimization of IDCs in the smart grid environment. When the electricity usage is high, the price maker in the electricity market will be high. IDC will affect the price of electricity. By several observations, the price varies with load and consumers. Then by contrive the cost of electricity minimization problem and optimization of electricity is subjected to end-to-end delay constraints. In order to make it more efficient solution method to transform the problem to a quadratic programming, performance is evaluated. The result shows that the total electricity cost of IDCs is minimized by handling the interaction between IDCs and smart grid (Wang et al. 2011). Cloud Computing is used to prove dramatically scalable and virtualized resource as a service over the internet. Datacenters involve in virtualization technology to realize cloud computing. As the rapid growth of server, quantity and scale in the datacenter, it is a great challenge which is directly related to the number of hosted servers and their workloads. It is very important to design and deploy the energy efficient technologies for the data center. Potential consolidation of performance overheads of the server investigates the consolidation strategies and explores the process of live migration. These results show both the two technologies are used to implement the energy saving goals with lower performance overheads (Ye et al. 2010).

3 RESEARCH METHODOLOGY OF IMPLEMENTATION

Proposed framework executes improved Energy-effective Green Control (EGC) calculation which utilizes SR arrangement for improving vitality effectiveness in lattice registering. Proposed framework utilizes open source cloud execution and OpenStack gives cloud design which is the best to fabricate VI. VI is made in OpenStack utilizing two different ways:

- VI is made legitimately in OpenStack Dashboard
- Using jclouds API, occurrence is made in OpenStack

In the event that a solitary occupation is dispensed to one VI at any given moment, control utilization will be high and reaction time will be postponed. So as to give vitality productivity and low power utilization in a cloud, work is isolated into different little undertakings. These assignments are distributed to a framework of VI. Matrix of VI performs employment to lessen reaction time. SR approach is executed to give an effective green control which distributes VI dependent on the utilization and can powerfully summon the VI on demand. It additionally helps in adjusting the heap between the VI as the activity arrives. The modules incorporate OpenStack arrangement, manufacture and design occurrence, dispatch a case and power the executives in framework registering.

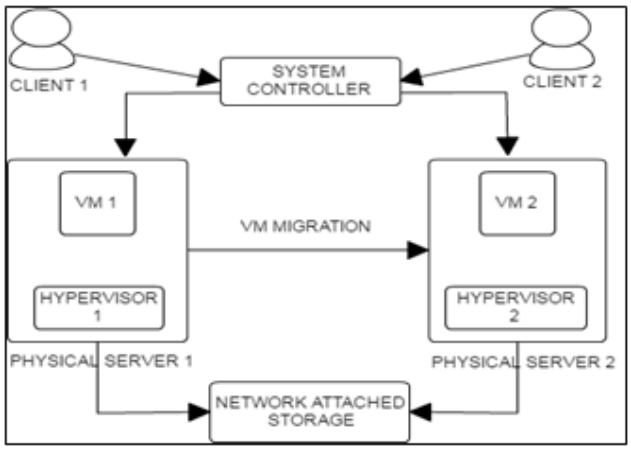

Fig.1 Architecture of VM migration

3.1 Openstack Cloud Setup

In this module, router configuration will be completed. The following setting is done: switch IP address and WLAN settings. After Router setup, virtualBox is introduced. Circle space will be distributed to virtualBox. Presently virtual apparatuses are imported in VirtualBox. Next processor and connector settings should be set. Confirmation step will be done in openStack. To begin server, the accompanying order to be utilized. /Stack. Sh. Server begin and it gives OpenStack IP and validation subtleties in terminal. To stop server, the accompanying order is to be utilized. /Unstuck. Sh.

3.2 Build and Configure Instance

In this module, picture and Instance will be made in openStack dashboard. To open dashboard, open program and type openStack IP. The dashboard gets opened. Give username and secret phrase and the page gets diverted to openStack administrator page. On the left half of the dashboard, under administrator tab, click pictures connect. In picture connect, click make picture catch. A discourse box gets opened. Give name, depiction, select record in picture source and transfer a picture from the framework and select the arrangement as QCOW2. At that point click open checkbox. Presently click make picture catch. The picture gets transferred to openStack. To dispatch an occasion, the accompanying strides to be pursued: at the left half of the dashboard, under venture tab, click cases connect. Snap dispatch an occurrence at the upper right of occasion page. Select accessibility zone as a nova, give occurrence name, season, boot from picture as case boot source and pick picture. Presently click dispatch catch. A VI gets propelled in openstack dashboad.

3.3 Grid Computing and Application Development

In this module, example is propelled utilizing jclouds API. To dispatch an occasion in openStack utilizing jclouds API, give confirmation and openstack-nova supplier to jclouds. At that point give RAM, picture name and example name. Utilizing nodeMeta Data, occasion will be propelled in openStack dashboard. A different IP will be made for every VI. To develop a Grid of VI, administrator needs to give the absolute number of VI to run, beginning VI to be in running state and RAM determination. A network of examples will be made one by one for the absolute number of cases. The base number of occasions to be run is held and other VI is put to rest state. The client can exchange a web application to VI. Presently client can send their web applications in VI. Burden adjusting and SR approach isn't actualized on VI in Grid Computing which may prompt high over-burden and at last crashes the server examples. The power utilization and reaction time will likewise be high.

3.4 Power Management Grid Computing

Administrator will distribute occupations to VI. This activity will put on FCFS line and can be served in framework figuring condition. Occupation is part into little undertakings and designated to framework of running VI. The memory for every VI is observed ceaselessly to counteract over-burdening. Edge esteem will be checked with memory use and if any VI surpasses, it will be accounted for to cloud administrator. Burden adjusting is accomplished by setting off the SR Policy to stack another VI which is in the rest state. This guarantees uniform circulation load among all the VI that helps in avoiding high memory use which will definitely impact control utilization.

4 RESULTS AND DISCUSSION

The performance evaluation is carried by number of experiments and those are listed below. The first module clarifies the establishment of Kernel Virtual Machine (KVM) (Akiyama et al. 2014). The initial step is to check whether the CPU bolsters the equipment virtualization (Kumar & Schwan 2008). The command utilized to check the equipment support is egrep-c '(vmxlsvm)'/proc/cpuinfo. The subsequent stage is to check whether the processor is 64 bit kernel. The command utilized for this rationale is egrep-c 'lm'/proc/cpuinfo. On the off chance that the output got is 0, then the processor is not 64bit kernel. On the off chance that the output obtained is 1 or more than 1, then the processor is a 64bit kernel. The following stage is to introduce 5 vital bundles to help the KVM. Running a 64 bit kernel on the portion is not required to make the accessibility of more than 2GB of RAM for the VM which the client will make (Shetty et al. 2012).

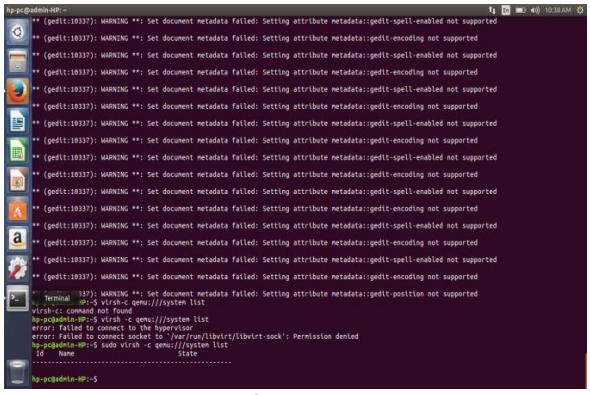

Fig.2 *KVM*

Figure.2 shows KVM installation. A 64-bit kernel should be utilized. On a 32-bit kernel establishment, there will be a confinement of 2GB RAM for a given VM. The above order causes the client to check whether the given VM bolsters 64bit part or not. On the off chance that 0 is appeared as output, it implies that the CPU is not 64-bit. If 1 or higher, then it is a 64 bit kernel. To check whether the running kernel is lesser than or equivalent to 64-bit, the following command

uname –m is used. If the output is x86_64, at that point it shows that the running kernel is a 64-bit kernel. In the event that the client sees other than that, at that point the client is running a 32-bit kernel. The 32 bit kernel should not be utilized in light of the fact that it won't bolster the VM. The five packages that are to be introduced are

- qemu-kvm
- libvirt-bin
- bridge-utils
- virt-manager
- qemu-system

The qemu-kvm is utilized to run the information and yield in the virtualization. The instrument that is utilized to associate with the product for virtualization is the libvirt. Libvirt is useful in running the qemu-kvm. It underpins numerous wide scope of hypervisor. The virt-manager is extended as the VM supervisor. VM supervisor is the user interface for driving VM. The bridge-utils associates the qemu-kvm with the libvirt-bin. The progressions with respect to the formation of VM, keep alive break must be changed in the design document. The design document can be opened by utilizing the command sudo vim /etc/libvirt/libvirtd.conf. After the establishment of all the 5 packages, make the important changes in the setup document and after that the establishment is checked utilizing the accompanying command $ virsh -c qemu:///system list.

5 CONCLUSION

The energy competence is achieved in openstack cloud by means of enhanced EGC algorithm by means of using grid computing to allocate the jobs to the grid of VI by subdividing the single job into multiple jobs by means of using divide and conquer algorithm and the jobs will be selected based on the jobs which comes first which is determined by using the FCFS algorithm. By using the SR policy, the VI from the sleep mode is waken to busy mode when any one of the VI has reached its threshold level and in need of another VI thus here the energy is efficiently used by waking the VI only when there is a need and since the job is divided by grid computing the response time which will be less.

REFERENCES

[1] Ahmad, RW, Gani, A, Hamid, SHA, Shiraz, M, Xia, F & Madani, SA 2015, „Virtual machine migration in cloud data centers: a review, taxonomy and open research issues", The Journal of Supercomputing, vol. 71, no. 7, pp. 2473-2515.

[2] Al-Shayeji, MH & Samrajesh, MD 2012, „An energy-aware virtual machine migration algorithm", In IEEE International Conference on Advances in Computing and Communications, pp. 242-246.

[3] Buyya, R, Broberg, J & Goscinski, AM 2010, „Cloud computing: Principles and paradigms", John Wiley & Sons, vol. 87.

[4] Chiang, YJ, Ouyang, YC & Hsu, CHR 2015, „An efficient green control algorithm in cloud computing for cost optimization", IEEE Transactions on Cloud Computing, vol. 3, no. 2, pp. 145-155.

[5] Dalvi, R & Madria, SK 2015, „Energy efficient scheduling of fine- granularity tasks in a sensor cloud", In International Conference on Database Systems for Advanced Applications, Springer, pp. 498-513.

[6] Duggan, GP & Young, PM 2012, „A resource allocation model for energy management systems", In IEEE International Systems Conference SysCon, pp. 1-3.

[7] Fakhfakh, M, Cherkaoui, O, Bedhiaf, IL & Frikha, M 2009, „High availability in IMS virtualized network", In First IEEE International Conference on Communications and Networking, pp. 1-6.

[8] Liu, Z, Qu, W, Liu, W & Li, K 2010, „Xen live migration with slowdown scheduling algorithm", In IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies, pp. 215-221.

[9] Piyare, R & Lee, SR 2013, „Towards internet of things (iots): Integration of wireless sensor network to cloud services for data collection and sharing", International Journal of Computer Networks & Communications, vol. 5, no. 5, pp. 59-72.

[10] Shetty, J, Anala, MR & Shobha, G 2012, „A survey on techniques of secure live migration of virtual machine", International Journal of Computer Applications, vol. 39, no. 12, pp. 34-39.

[11] Travostino, F, Daspit, P, Gommans, L, Jog, C, De-Laat, C, Mambretti, J & Wang, PY 2006, „Seamless live migration of virtual machines over the MAN/WAN", Future Generation Computer Systems, vol. 22, no. 8, pp. 901-907.

[12] Ye, K, Jiang, X, Huang, D, Chen, J & Wang, B 2011, „Live migration of multiple virtual machines with resource reservation in cloud computing environments", In IEEE 4th International Conference on Cloud Computing, pp. 267-274.

[13] Zhang, P, Sun, H & Yan, Z 2013, „A novel architecture based on cloud computing for wireless sensor network", In Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering Atlantis Press. pp. 472-475.

[14] Zhang, Q, Zhani, MF, Boutaba, R & Hellerstein, JL 2014, „Dynamic heterogeneity-aware resource provisioning in the cloud", IEEE transactions on cloud computing, vol. 2, no.1, pp. 14-28.

[15] Zu, Y, Huang, T & Zhu, Y 2013, „An efficient power-aware resource scheduling strategy in virtualized datacenters", In IEEE International Conference on Parallel and Distributed Systems, pp. 110-117.