Cloud Computing for ECG Analysis Using MapReduce

Kerk Chin Wee* and Mohd Soperi Mohd Zahid
Faculty of Computing,
Universiti Technology Malaysia,
Skudai, Johor, Malaysia.
*e-mail: kerkchinwee@hotmail.co.uk, e-mail: soperi@utm.my

Abstract— **Electrocardiograph (ECG) analysis brings a lot of technical concerns because ECG is one of the tools frequently used in the diagnosis of cardiovascular disease. According to World Health Organization (WHO) statistic in 2012, cardiovascular disease constitutes about 48% of non-communicable deaths worldwide. Although there are many ECG related researches, there is not much efforts in big data computing for ECG analysis which involves dataset more than one gigabyte. ECG files contain graphical data and the size grows as period of data recording gets longer. Big data computing for ECG analysis is critical when many patients are involved. Recently, the implementation of MapReduce in cloud computing becomes a new trend due to its parallel computing characteristic. Since large ECG dataset consume much time in analysis processes, this project will construct a cloud computing approach for ECG analysis using MapReduce in order to investigate the effect of MapReduce in enhancing ECG analysis efficiency in cloud computing. The project is expected to reduce ECG analysis process time for large ECG dataset.**

Keywords— **ECG, cloud computing, MapReduce.**

I. INTRODUCTION

ECG analysis is a signal analysis process to extract heart condition information of patients from their heartbeat signals. Until now, ECG analysis is the most effective way in cardiovascular disease diagnosis. Traditionally, an ECG report will be printed after the completion of an ECG recording and doctors need to manually analyze printed ECG in order to gather patient heart information. It will suffer doctors and drag down process efficiency if the number of ECG reports is large. For example, an ECG record for 24 hours monitoring of a patient may size up to 100 MB or more. Depending to different needs of ECG analysis, there can be up to hundreds ECG records for different uses which may generate around hundreds MB to 1GB ECG dataset per day. Due to different uses, an ECG record can be a 15 minutes, 30 minutes, one or more hours, or even 24 hours record. Therefore, ECG dataset can be not only big in size, but complex to be analyzed. Due to short time constraint of getting ECG analysis result, an efficient computing paradigm is needed to save more lives at stake. Since the recent ECG devices are equipped with wireless communication technologies such as WIFI or Bluetooth which enable ECG data transmission from ECG devices to other electronic devices such as computers, mobile devices or network terminals, it makes the computerization of ECG analysis process possible.

Although there are many researches for ECG analysis algorithm, ECG personal monitoring and other ECG cloud solutions, there is lack of researches in big data computing for ECG analysis which involves dataset more than one gigabyte. Since public healthcare systems heavily rely on ECG analysis for cardiovascular disease diagnosis, big data computing for ECG analysis is critically needed because a public hospital may generate up to terabytes ECG data over a year. Recently, MapReduce becomes popular paradigm for big data computing due to its parallel computing ability.

In order to investigate the capability of MapReduce in computing large ECG dataset over 10GB mixed from ECG signals with same number of ECG channel but different ECG data file types including header file, data file and annotation file, this research convert custom ECG analysis process into cloud computing process using MapReduce paradigm and investigate its performance. Notice that the ECG dataset mixes with different file types because the annotation files record the time of occurrence for events, data files record ECG signals and header files record details of ECG signals. Although more works must be done to obtain the required information based on different ECG files, the parallel computing characteristic of MapReduce framework is capable to grouping the files efficiently and work on them in parallel in order to shorten the information extraction period.

II. ECG

ECG is the record of the bio-electric potential variation detected via the electrodes throughout the time. The bio-electric potential variation is heart signal with regular circles as a heartbeat. Each heartbeat cycle has a P-wave, QRS complex and T-wave as shown in Fig. 1.

978-1-5090-0424-9/16 $31.00 © 2016 IEEE
DOI 10.1109/ACSAT.2015.21

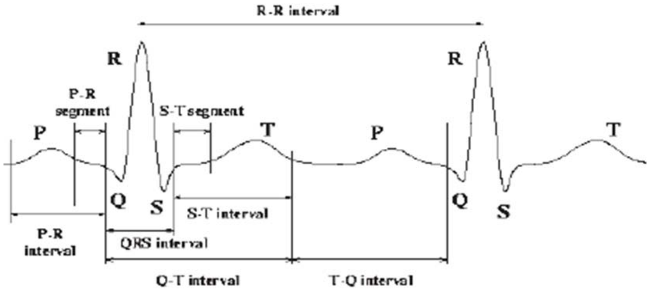


Fig. 1. ECG signal

The ECG features represents identical heart activities respectively such as P-wave is the signal generated by atria depolarization meanwhile QRS complex is the signal generated by depolarization of ventricles. Hence, any abnormal of ECG signal features address heart functionality problem specifically. Notice that QRS complex is critical to identify heart rate from ECG signal because it is significant to identify the number of heart cycle per minute. Therefore, this research focus on extract QRS complex from ECG signal.

III. Cloud Platform – Google Cloud Platforms

Cloud is the optimum solution for platform of computation because it provides flexibility in scaling resources based on the computational needs. There are several cloud platforms provided by big IT companies such as Amazon, Google and Microsoft for users to purchase cloud resources desirably. Among the cloud platforms, this research selected Google Compute Engine (GCE), the IAAS (Infrastructure As A Service) type platform of Google Cloud Platform because it is the most cost efficient platform compared to other IAAS platform such as Amazon Web Services (AWS). According to [10] and [11], at the time being, purchasing a virtual machines (VM) for 2CPU and 6~8GB RAM specification costs $0.104~$0.126 per hour in AWS but only costs $0.07~$0.1 per hour in GCE. Notice that the cost for purchasing a VM in AWS is fixed price but GCE vary with the usage where $0.07 per hour is the price of holding a VM without using it and $0.1 per hour is the price of holding a VM with full usage. In order to purchase GCE services, users are required to have a Google account. The detail of using GCE in this research will be illustrated in section VI part B.

IV. ECG Dataset Of Size More Than One GigaBytes

ECG dataset always be big data in healthcare system. Indeed, an ECG record for 24 hours monitoring of a patient may size up to 100 MB or more. Since an ECG record may has one than one ECG channel, the ECG data size maybe multiplied based on the number of ECG channel. Depending to different needs of ECG analysis needs, there can be up to hundreds ECG records for different uses. In this case, a public hospital can generate around hundreds MB to 1GB ECG dataset per day. Due to different uses, an ECG record can be a 15 minutes record, 30 minutes record, one or more hour record, or even 24 hours record. These ECG records also vary with number of ECG channels. Therefore, ECG dataset can be not only big in size, but complex to be analyzed. Due to short time constraint of getting ECG analysis result, an efficient computing paradigm is needed to save more lives at stake.

V. Hadoop MapReduce

The Hadoop framework used for MapReduce in this research is Hadoop-2.6.0 version. It has default capacity scheduler for task scheduling and YARN architecture for NextGen MapReduce or MapReduce v2. In YARN architecture, there are two important components to able the parallel computation work in MapReduce cluster which are Resource Manager (RM) and Node Manager (NM). There will be only one RM works on master node of MapReduce cluster to schedule the MapReduce tasks according to the conditions of slave nodes in the cluster meanwhile there will be a NM works on each slave node to manage task execution and update the task status in the slave node. The details of YARN architecture is shown in Fig. 2. Since NextGen MapReduce is expected to superior MapReduce v1 provided in Hadoop 0.x or 1.x versions in term efficiency due to framework structuring, this research decides to make it as the research target in Auto-Tuning because still not many MapReduce research involve NextGen MapReduce in their researches.

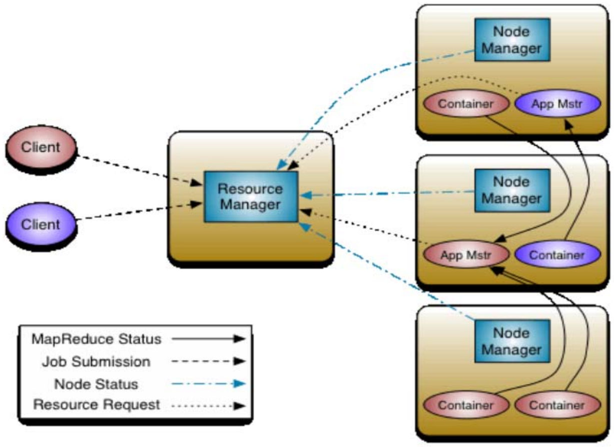


Fig. 2. Apache YARN architecture

In view of MapReduce paradigm, there is almost no different between MapReduce v1 and NextGen MapReduce which processes as shown in Fig. 3. However, YARN architecture make the MapReduce framework changes by changing the software architecture which are:

i. from jobtracker to resource manager
ii. from tasktracker to node manager

These changes are due to the demands of enhancing the capacity of Hadoop framework in better schedule, keep tracking and fast react to MapReduce tasks. Notice that MapReduce job is a client request for program such as a request of ECG analysis meanwhile MapReduce tasks are the distributed works from a MapReduce job to make it process in parallel such as reading ECG file to get QRS complex data. The number of MapReduce tasks depends on the number of splits per data in a MapReduce job.

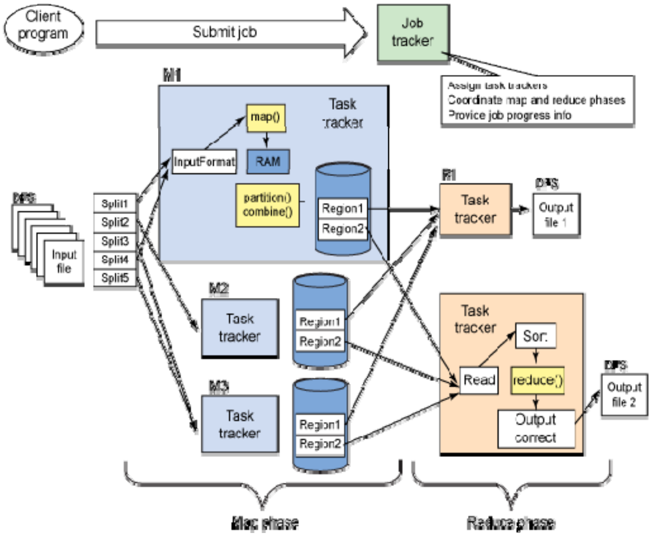


Fig. 3. MapReduce paradigm

VI. Related Works

Wang et al. (2014) proposed a solution for ECG mobile computing that using smartphone as ECG analysis tool and cloud as the training agent for ECG analysis model as shown in Fig. 4.

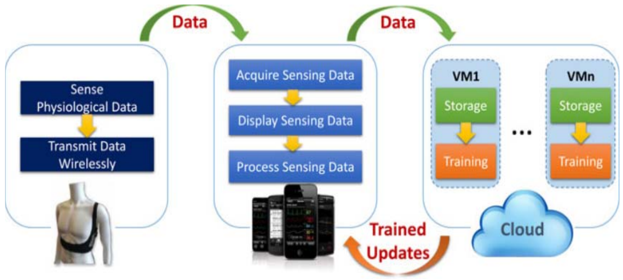


Fig. 4. Mobile Cloud Model proposed by Wang et al. (2014)

The research gives impressive result which speedup the analysis processing averagely 37 times and save around 88% of mobile device energy compared to only use mobile devices for analysis. However, this research also reveals the truth that the limitation of computational resources of mobile device in term of processing power, energy resource such as battery life and data storage gives a great challenge to mobile based healthcare system. Therefore, instead of using mobile computing, cloud computing is more suitable for ECG analysis.

Sahoo et al. (2014) first proposed distributed computing approach for ECG analysis using MapReduce for ECG feature extraction. The research shows the execution time for ECG analysis greatly reduced compared to using desktop and the execution time also reduce with the increase of processing nodes in cloud which in turn prove the capability of MapReduce in enhancing cloud computing efficiency. In the research, 3.2GB ECG signal for 4 ECG channels requires 33 minutes to be processed by desktop but only need 1.57 minutes to be processed by MapReduce using 4 nodes. Although Sahoo et al. (2014) shows the capability of MapReduce in speedup ECG analysis for signals with same channel number, the research do not investigate the capability of MapReduce in ECG analysis with mixed ECG signals which provided by this research.

VII. Methodology

A. *Material and Tools*

The ECG dataset for this research is a mixed collection of ECG dataset from three ECG databanks provided by Physionet, the online ECG dataset provider. The ECG databanks are European ST Databank which consists of 90 records of 30 minutes ECG data with total size 488MB, Long term AF Database which consists of 84 records of 21 hours ECG data with total size 3.7GB and Long term ST Databank which consists of 89 records of 24 hours ECG data with total size 6GB. The details of ECG dataset are stated in Table 1.

TABLE 1. ECG Dataset Details

ECG data size	10.2GB
ECG record numbers	263
ECG record components	.hea, .dat and .ann files
ECG dataset composition	1. 90 records of 30 mins ECG data 2. 84 records of 21 hours ECG data 3. 89 records of 24 hours ECG data

As mentioned in section III, the tool for cloud computing in this research is GCE platform. This research only uses Virtual Machines (VM) provided by GCE platform for MapReduce master and slave nodes. Meanwhile, the file system for Hadoop framework is Hadoop Distributed File System (HDFS) which locates data among VMs in MapReduce cluster and enable the data to be shared among cluster. The details of GCE cluster setup will be illustrated in part B.

B. Google Compute Engine (GCE) Setup

GCE setup starts from creating a Google account if do not have it. Once having a Google account, the research starts from setup a MapReduce cluster by purchasing VMs with specifications listed in TABLE 2. Next, Oracle Java 8 Java Virtual Machine (JVM) is installed in every VMs. Then, Hadoop-2.6.0.tar.gz file is downloaded into master node and extracted into home directory. After that, adding Hadoop environment variable in .bashrc file under home directory, and edit core-site.xml, hdfs-site.xml and mapred-site.xml under Hadoop-2.6.0 folder before creating folders for datanode and namenode in order to setup default Hadoop framework. Finally, copy the Hadoop 2.6.0 folder in master node to other slaves by gcloud feature and add Hadoop environment variables in the .bashrc file of slave nodes.

TABLE 2. VM Specification

Data disk type (Master and worker nodes)	standard persistent disk
Data disk size in GB (Master and worker nodes)	100
Machine type (Master and worker nodes)	n1-standard-2 (2vCPUs, 7.5GB RAM)
Machine OS	Ubuntu 14.04 LTS
Enable request between cluster nodes	Yes
Enable http request	No
Enable https request	No

After the cluster is setup, wfdb library provided by Physionet website is installed as the ECG library which read European Data Format (EDF) ECG data for ECG analysis. Finally, the runnable jar file for MapReduce program and ECG dataset are uploaded to the cluster for MapReduce process.

C. ECG Analysis Approach

The ECG analysis approach for this research is ECG feature extraction for QRS detection. The approach first read ECG information from header files of EDF format ECG by wfdb library function, followed by using high-pass and low-pass filters to obtain QRS complex from ECG signal (.dat file), before save it into QRS file (.qrs file) as shown in Fig 3. The filters work to remove unwanted components such as P-wave, T-wave and noises from ECG signals in order to get the QRS complex as residual signal.

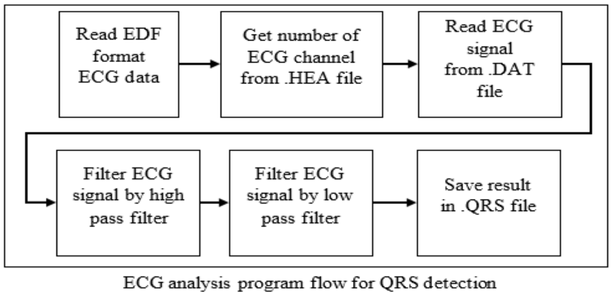

Read EDF format ECG data
Get number of ECG channel from .HEA file
Read ECG signal from .DAT file
Filter ECG signal by high pass filter
Filter ECG signal by low pass filter
Save result in .QRS file
ECG analysis program flow for QRS detection

Fig. 3. QRS detection algorithm for ECG analysis

Since ECG analysis program should perform the QRS detection for each record in ECG dataset, the QRS detection algorithm in Fig. 3 is performed recursively throughout the ECG dataset as shown in Fig. 4. Notice that Fig. 4 is the flow chart of custom ECG analysis program for QRS detection.

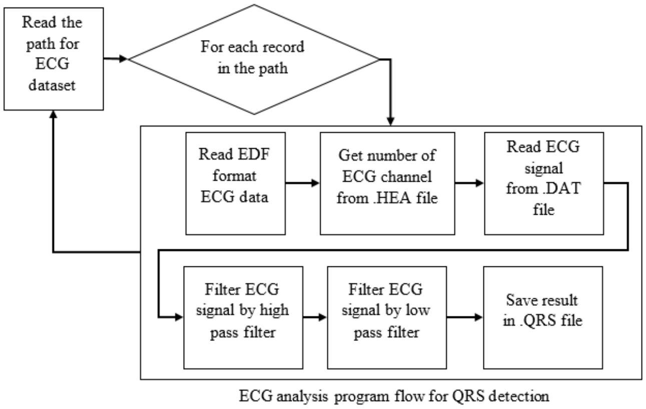
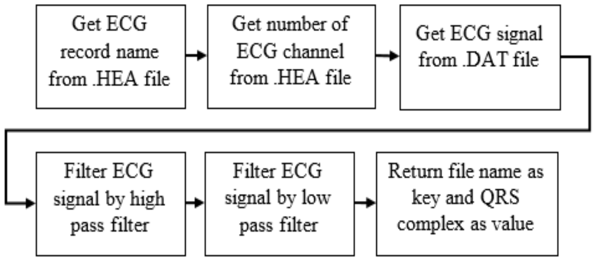

Read the path for ECG dataset
For each record in the path
Read EDF format ECG data
Get number of ECG channel from .HEA file
Read ECG signal from .DAT file
Filter ECG signal by high pass filter
Filter ECG signal by low pass filter
Save result in .QRS file
ECG analysis program flow for QRS detection

Fig. 4. ECG Analysis Program using QRS detection algorithm throughout ECG dataset

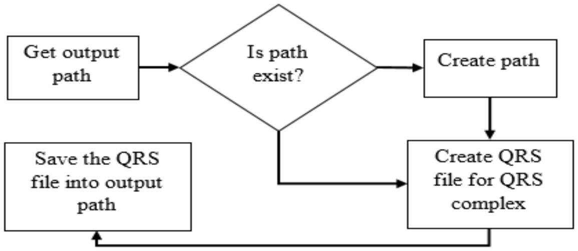
D. MapReduce Algorithm for ECG analysis

MapReduce is the parallel computational framework to perform ECG analysis in parallel. Typically, MapReduce algorithm is divided into map and reduce functions. Map function is usually the main computation needed to be executed in parallel on multiple mappers as map tasks meanwhile reduce function is the result reduction process running on one or more reducers as reduce tasks. In this research, the map function is QRS detection as shown in Fig. 5 and reduce function is QRS detection result writing is shown in Fig. 6.



Get ECG record name from .HEA file
Get number of ECG channel from .HEA file
Get ECG signal from .DAT file
Filter ECG signal by high pass filter
Filter ECG signal by low pass filter
Return file name as key and QRS complex as value

Fig. 6. Map function algorithm for ECG analysis



Get output path
Is path exist?
Create path
Save the QRS file into output path
Create QRS file for QRS complex

Fig. 7. Reduce function algorithm for ECG analysis

MapReduce process works by first breakdown the large ECG dataset around 10.2GB into small workloads of size around 64MB by resource manager, before the resource manager schedules and assigns workloads to mappers as map tasks. After all map tasks completed, the immediate results released by mappers are scheduled and assigned to reducers as reduce tasks. After all reduce tasks are finished, the QRS detection results are collected as .qrs files in cloud storage bucket of GFS. The whole process of MapReduce is shown in Fig. 7. Notice that worker nodes in GCE cluster are assigned as mappers and reducers in MapReduce process.

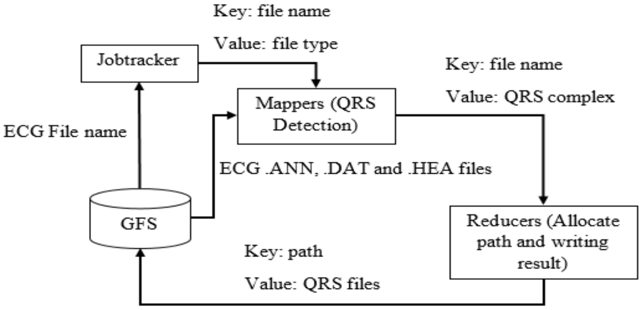


Fig. 7. MapReduce framework for ECG analysis

In case of ECG analysis, the dataset is the collection of small ECG data files which size from 5KB to 100MB. In order to make Hadoop to perform MapReduce for multiple small ECG files, the input format for Hadoop must be set on CombinedFileInputFormat and the split operation for single file should be disabled. This will make the ECG analysis process in MapReduce similar to pipelining for ECG analysis of large amount of input data.

VIII. Result

The result of ECG analysis processing time is shown in Table 3 and Table 4 whereas Table 3 is the ECG analysis processing time of same ECG dataset vary with cluster size and Table 4 is the ECG analysis processing time of same cluster size vary with ECG dataset size. This research compare ECG analysis processing time among the approaches of custom method and MapReduce with different number of VMs in the cluster. Custom approach is the approach that run ECG analysis program on one VM without using MapReduce meanwhile MapReduce approach is the approach that run ECG analysis under MapReduce paradigm with different number of VMs. Due to the current status of VM in term of disk utility, RAM usage and so on, the ECG analysis processing time for all cases are vary all the time during experiment. Thus, the ECG analysis processing time for all cases are taken by average of 10 runs. Notice that all VMs in the cluster for whether custom approach or MapReduce approaches have same specification in order to ensure the result consistency.

TABLE 3. Job complete time of MapReduce for ECG Analysis vary with cluster size

Cases	Average process time (sec)
No MapReduce (1 VM)	3445.9
MapReduce with 1 node	488.7
MapReduce with 2 nodes	259.3
MapReduce with 3 nodes	185.1
MapReduce with 5 nodes	131.1

TABLE 4. Job complete time of MapReduce for ECG Analysis vary with dataset size

ECG Dataset size (GB)	Average process time (sec)
1	91.4
4	246.0
10	488.7

IX. Discussion

The result shown in Table 3 indicates the MapReduce approach has much higher computational efficiency than custom approach which reduce ECG analysis processing time from 3445.9 secs to 488.7 secs with the same resource. This proves MapReduce is capable to better schedule the resource for ECG analysis than custom computing approach. The result also reveals that ECG analysis processing time for MapReduce approach reduces with the increase of number of VMs in cluster. This is because the more VMs in the cluster, the more work nodes to run ECG analysis in parallel and the more ECG data to be processed in the same time, the faster ECG analysis process can be completed. At last, the result also shows that ECG analysis processing time with MapReduce approach is not directly. For example, suppose MapReduce with 2 nodes should need half of ECG analysis processing time compared to MapReduce with 1 node, but it takes longer time. This is due to the performance of MapReduce with more than one VMs maybe affected current condition of cluster network and efficiency of task scheduling algorithm. The result in Table 4 shows that ECG analysis processing time increases with ECG dataset size. This is due to large ECG dataset size results in more MapReduce tasks are needed to be processed for ECG analysis to be completed. Notice that, MapReduce works stable regardless the size of ECG data because Hadoop will divide the data into splits of almost same size before sharing them among the work nodes although it may sometime affected by current VM status. This is proven by mixing large size ECG data (51~66MB per signal file) from Long Term ST Dataset, with smaller size ECG data from Long Term AF Dataset and European ST Dataset.

X. Conclusion And Future Work

In conclusion, MapReduce is proven to be able to speed up the ECG analysis computation. In fact, a collection of 89 records of 24 hours ECG data, 84 records of 21 hours ECG data and 90 records of 30 minutes ECG data can be processed within 12 to 30 minutes by a cluster of 3 to 5 VMs. Since it may be faster if the number of VMs increases, a large cluster of 10 or more VMs may process 10GB ECG data within 10 minutes which may save a lot of lives at stake. In order to further improve the performance of MapReduce in ECG analysis, the future work of this research will focus on enhancing Hadoop MapReduce.

Acknowledgment

The authors would like to thank Ministry of Education Malaysia and Universiti Teknologi Malaysia (UTM) for supporting this research under Vote no 4F375.

References

[1] SAHOO, S. S., JAYAPANDIAN, C., GARG, G., KAFFASHI, F., CHUNG, S., BOZORGI, A., CHEN, C. H., LOPARO, K., LHATOO, S. D. & ZHANG, G. Q. (2014). Heart beats in the cloud: distributed analysis of electrophysiological 'Big Data' using cloud computing for epilepsy clinical research. Journal of the American Medical Informatics Association, 21, 263-271

[2] SEENA, V. & YOMAS, J. (Year) A review on feature extraction and denoising of ECG signal using wavelet transform. Devices, Circuits and Systems (ICDCS), 2014 2nd International Conference on, 6-8 March 2014 2014. 1-6.

[3] VELIC, M., PADAVIC, I. & CAR, S. (Year) Computer aided ECG analysis - State of the art and upcoming challenges EUROCON, 2013 IEEE, 1-4 July 2013 2013 Zagreb, Croatia. 1778-1784.

[4] VANEGHI, F. M., OLADAZIMI, M., SHIMAN, F., KORDI, A., SAFARI, M. J. & IBRAHIM, F. (Year) A Comparative Approach to ECG Feature Extraction Methods. Intelligent Systems, Modelling and Simulation (ISMS), 2012 Third International Conference on, 8-10 Feb. 2012 2012. 252-256.

[5] MAZOMENOS, E. B., BISWAS, D., ACHARYYA, A., TAIHAI, C., MAHARATNA, K., ROSENGARTEN, J., MORGAN, J. & CURZEN, N. (2013). A Low-Complexity ECG Feature Extraction Algorithm for Mobile Healthcare Applications. Biomedical and Health Informatics, IEEE Journal of, 17, 459-469.

[6] XIAOJUN, Z., XIULI, M. & YANG, L. (Year) An adaptive threshold algorithm based on wavelet in QRS detection. Audio, Language and Image Processing (ICALIP), 2014 International Conference on, 7-9 July 2014 2014. 858-862.

[7] GUNARATHNE, T., TAK-LON, W., QIU, J. & FOX, G. (Year) MapReduce in the Clouds for Science. Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on, Nov. 30 2010-Dec. 3 2010.

[8] LI, M., XU, G.-H., WU, L.-F. & JI, Y. (Year) Performance Research on MapReduce Programming Model. Instrumentation, Measurement, Computer, Communication and Control, 2011 First International Conference on, 21-23 Oct. 2011 2011b. 204-207.

[9] GAIZHEN, Y. (Year) The Application of MapReduce in the Cloud Computing. Intelligence Information Processing and Trusted Computing (IPTC), 2011 2nd International Symposium on, 22-23 Oct. 2011 2011 Wuhan, Hebei, China. 154-156.

[10] https://cloud.google.com/compute/?utm_source=google&utm_medium=cpc&utm_campaign=2015-q1-cloud-japac-my-gce-bkws-freetrial&utm_content=en&gclid=CN_n0aGv88cCFVQpjgodLUoLvA#pricing

[11] https://aws.amazon.com/ec2/pricing/