Implementation of Energy Efficient Fog based Health Monitoring and Emergency Admission Prediction System Using IoT

S. Amudha
School of Computing, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.
E-mail: amudhas@srmist.edu.in

M. Murali
School of Computing, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.
E-mail: muralim@srmist.edu.in

Received October 30, 2020; Accepted November 30, 2020
ISSN: 1735-188X
DOI: 10.14704/WEB/V18SI02/WEB18065

Abstract

With rapid development in Information Communication Technology (ICT), Wearable Sensor Networks with Internet of Things (WSN-IoT) has produced several improvements in the smart world environment. One of the main research challenges in Wearable Sensor is energy, since all the sensor nodes operation depends on battery power consumption. Hence a new middleware has to be introduced between Wearable Sensor nodes and Cloud to reduce latency and Power Consumption problems. Overcrowding in hospital premise, detecting priority of hospital admission for patients, managing and monitoring health status of the patient constantly are daily problems in any health care system. Even though IoT based wearable sensors monitor health status of patients regularly and provide intent treatment in critical stage, but there is some block hole in that such as latency, energy issues and unawareness of medical execution plans and policies to preserve them from sudden attacks such as Heart attack. The proposed work is to implement energy efficient FoG based IoT network to monitor patients’ health conditions from chronic diseases and highlights utility of Deep Learning model for analyzing the health condition of patients and predicting Emergency readmission cases well in advance. This model is also compare with existing machine learning algorithms such as Gradient boosted, Decision tree, Random forest and Logistic regression to achieve more accuracy. This paper introduces preemptive interval scheduling algorithm with predictive analysis for constant monitoring of status for critical patients. By means of comparative analysis done in this work energy efficiency has been achieved prominently.
Keywords


Introduction

With the recent technological development in all fields the human to human and human to machine interaction has been tremendously changed. Cloud Computing is prominent technology nowadays, which is used in almost all application for efficient data processing and analysis. If we include IoT (Internet of Things)-based Cloud computing then the usage of technology with different context will increase [1]. IoT based Cloud can be used in various applications such as Smart Home, Smart Hospital, Smart Environment Monitoring, Smart Manufacturing and Smart Industry [2]. In healthcare applications, wearable wireless sensor elements are playing crucial role in patient health monitoring remotely. Wireless elements which are wearable in nature are fixed in human body so that their health informations are continuously monitored, and in emergency cases proper treatment also taken care by the intended care giver or Nurses. [2,3]. All the collected data are processed by Cloud server which has lots of processing steps such as data collection, transfer, analysis and prediction [4]. This cloud computing server provides high scalable software based on environmental or context conditions. These environments are configured using that software and huge volume of data are processed effectively with less human intervention. [4].

Generally, each patient’s health conditions are monitored by several wearable devices which are attached to their body. In this making a platform based approach dedicated to particular patient is not suitable for all scenarios. So that, we need an alternative method where all wearable nodes are used in uniform platform without worrying about network connection and processing cost etc. To overcome these problems IoT provides a proper solution by enabling things to things communication. Nowadays several healthcare applications are using simple sensor-to-cloud platform for health status analysis which is not suitable for many cases, since there is possibility of data breach happened in outside world. Hospital environment does not allow to store patient data outside [5] which leads privacy and security problems.

Therefore, a new paradigm is required named as fog computing which is introduced by one of the popular company named as Cisco [6], to fulfill the IoT needs and execute those types of data within deadline. In this the FoG nodes acts like a middleware component
between Sensor devices and Cloud Services. It offers tremendous services such as storage, compute, and network activities between cloud services to IoT Devices.

All the processing takes place in EU (End User) that is local processor [7]. Moreover, this new paradigm provides features such as minimum latency and response time hence its highly applicable for emergency environment. This type of new platform provides Internet of Everything (IoE) applications. Different types of application components are working between Sensors and Cloud of between Edge to Cloud server. Example for such type of components are smart gateways, routers and fog devices.[8] To provide efficient computing service between the user in hospital to Cloud server, Fog based cloud computing model is required. These types of services enable efficient fog-based cloud features for Smart Healthcare.

In this work the problems of overcrowding in hospitals during emergency stage also addressed properly. In the existing hospital scenario hospital admission, handling the severities conditions are long and time-consuming process. To overcome this issues a reasonable solution is the usage of machine learning algorithms to predict EC (Emergency Cases) entry in smart hospitals. In this work continuously gathered information are used for driving knowledge from that and uses this knowledge for administrative purpose. These collected data from different patients are compared and contrasted with different machine learning techniques for predicting the risk of admission from the EC. In order to build the predictive models this work uses logistic regression, decision trees and gradient boosted Algorithms (GBA).

Based on the literature survey conducted in this paper an energy efficient fog-enabled information model is used to prevent energy wastage. To efficiently manage the data of Diabetic patient and to provide healthcare as cloud service this Fog model in used.

This work is organized like this Section 2 is explaining about related works. Section 3 is explaining about proposed model. Section 4 is explaining about experimental setup and result analysis part. Finally section 5 gives about conclusion and future enhancement for analysis. This work is simulated using contiki cooja simulator where the energy utilization of the sensor nodes are calculated in simulated scenarios.

**Related Works**

One of the main applications of Wireless Sensor Network is Healthcare applications. By means of several wearable Sensor nodes to monitor patient health condition in emergency stage is easily done and diagnosis of particular disease also predicted well in advance. The
author Anand Paul and all in paper [9] proposed efficient FoG computing to collect and process health data efficiently for chronic disease analysis. First they explained FoG computing introduction, usage, Components, security and deployment issues and various research challenges in that. This new type of architecture not only eliminates the risk of patient but also reduce latency issues happened between Cloud to Sensor components and vice versa.

In paper [10] the author Wan et al proposed Fog computing for smart factory. They used Particle Swarm Optimization scheduling algorithms to provide efficient scheduling and energy-aware load balancing in various operations in smart factory. The performance metrics considered in this paper are Multiagents, and Candy packaging line. This work was implemented using ESP8266, Raspberry pie and UDOO board. Major contribution of this work is to propose a novel algorithm called as ELBS (Energy aware Load Balancing and Scheduling) based on Fog nodes. They introduced Multi-Agent system which uses Particle Swarm Optimization (PSO) algorithm to achieve load balancing. And for job shop problem they used dynamic scheduling algorithms.

In paper [11] the author presented a system which monitor health status of heart patients using ECG signals in low cost manner. Here they introduced automatic analysis and notification of system models. The system integrates IoT, Fog nodes along with sensor nodes in energy efficient manner. The sensor nodes gather ECG, respiration and temperature of patient and transmit that information remotely to FoG gateway using wireless channels. These informations are accessed by appropriate care-givers by doing automatic decision making and provide real-time notification for immediate attention.

In the research paper [12] Yang, et all introduces novel algorithms for similar Fog networks which is known as Delay energy balanced task scheduling (DEBTS) algorithm. In order to balance energy usage and delay in services, the author introduced cross layer analytical framework. This is framework model implementation not yet done. By reducing service and jitter delays, DEBTS algorithm is used by the author to minimize the overall energy consumption. The performance metrics considers in this work is latency and power consumption using Lyapunov optimization technique. The author Althamary in the paper [13] introduced Popularity based cached placement for Fog nodes which is used for reduction in energy consumption. This work was simulated using matlab, C++ software. They introduced performance metrics here is zipf distribution.

In the research paper [14] Naranjo simulated container based virtualization tool using iFog simulator. The author introduced heuristics framework designed mainly for resource
management which is mainly consider overall energy reduction. In [15] paper the author Conti introduced reinforcement learning method based on Markov model to minimize job loss and optimize server activation policy through Battery Energy Management, Which was done by Matlab tool.

### Table 1 Related Work and its implementation Softwares

<table>
<thead>
<tr>
<th>Author</th>
<th>Type of Execution</th>
<th>Method applied</th>
<th>Performance measurements</th>
<th>Main goal of the paper</th>
<th>Way of approach</th>
<th>Used softwares /hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wan et al [10]</td>
<td>Prototype</td>
<td>ELBS concept</td>
<td>PSO Algorithm, Multiagents</td>
<td>FoG node based ELBS method, Multi-Agent system for dynamic scheduling PSO based Mathematical Modeling for balancing load</td>
<td>Mathematical Modeling</td>
<td>ESP8266, Raspberry pi and UDOO board</td>
</tr>
<tr>
<td>Yang et al. [12]</td>
<td>Simulation</td>
<td>Energy and delay consideration</td>
<td>Lyapunov optimization technique</td>
<td>DEBTS algorithm - minimize energy consumption Cross-layer analytical framework for balancing energy</td>
<td>Analytical Framework</td>
<td>Nil</td>
</tr>
<tr>
<td>Naranjo et al [14]</td>
<td>Simulation</td>
<td>MBFD and MDC, Microsoft RAID, Virtualization concept based on container</td>
<td>PABP-dynamic resource management</td>
<td>Server activation policy is optimized using Reinforcement learning method. Markov based mathematical model to minimize job loss</td>
<td>Heuristic</td>
<td>ifogsim</td>
</tr>
</tbody>
</table>

### Proposed Health Monitoring System using IoT-FoG

The proposed health monitoring work based on IoT-FoG is discussed in this section which is mainly used in Healthcare applications. Fog-enabled cloud computing model used to manage health information of Diabetic patients effectively and identify the health condition and severity level well in advance. Figure 1 depicts the proposed architecture of
Health Monitoring System using IoT-FoG model which consists of multiple components and that are explained further.

**Body Area Sensor Network (Sensors Tier)**

This is first layer. The components in this layer is variety of sensors such as Temperature, activity sensor (MEMS) both left hand and right leg placed, Respiration Sensor and Heart beat sensor etc. This module is used to sense the health data of diabetic patient and transfer the collected information to IoT devices which are fixed in human body.

**1. Wi-Fi Module ESP8266**

This component is called as microcontroller boards know as ESP8266 with inbuilt Wi-Fi connectivity. This is mainly designed to operate as standalone application. Sometime this component act as slave device which is connected to a master microcontroller. This master is used to enable internet connection wirelessly for the intended applications. This component has internal SoC interface, system on chip and flash memory etc. It is operating at 80MHz clock speed and uses 32-bit processor.
2. LM35 Temperature Sensor

LM35 is a temperature sensor which is used to measure the temperature of the diabetic patient and display the value in degree Celsius. This is of type analog in nature. The collected values in this sensors ranges from -55°C to 150°C as well as the output voltage varies from 10mV in response to every increase of °C in temperature. The required power supply is operated from a 5V and 3.3 V supply. The stand by current is less than 60uA. This temperature sensor is connected with ESP8266 UNO node MCU for data gathering and analyzation.

3. Pulse Rate Sensor (Heart Beat Sensor)

Heart Beat Sensor is used to measure heart rate of diabetic patient, it uses well-designed plug-and-play mechanism for collecting the data using Arduino UNO. The sensor clips onto a fingertip or earlobe and they can be powered by internal battery of 5V power supply. All these collected information are transferred to Fog gateway continuously, and then analyzation happened that to identify the data is normal or abnormal. If any emergency data is detected they are immediately transferred to Tinkspeak cloud server for future prediction. Heart shape logo is covered in front of the sensor. Also there is LED light attached in this setup to indicate everything connected properly or not. These sensor shines light into fingertip and reads the amount of light that bounces back, using this method this sensor reads heart rate of the patient.
4. Respiration Sensor & MEMS Accelerator Sensor

Respiration sensors are used to measure oxygen rate of the person. All the collected breathing measurements are transferred to local FoG server through Node MCU which is connected and charged by internal battery. Also this work uses MEMS sensors which are abbreviated as Micro Electro Mechanical System and they developed by any sensor vendors, using technical microelectronic fabrication methods. These techniques create mechanical sensing structures of microscopic size, typically on silicon. These MEMS sensors are used to measure acceleration of patient body movement which is coupled with microelectronic circuits. In this work two acceleration sensors left leg and right hand can be attached for the patient to monitor acceleration in the body.
5. Thing Speak

Internet of Things programmer and coder can create an open source data storage platform called as Thing speak. It is free open source software platform used for storing purpose. Based on the stored results in graphical format with Matlab supportive tools, they are used for data analysis and prediction further. All the collected data are transferred to thing speak cloud server in simultaneous manner. All the uploaded data will be automatically converted to graphical representation for further progressing tasks. This tool uses an API known as “Application programming Interface”, on thing speak which contains string of random character which is combination of alphabets both lower and upper in case, digits and even special characters to identify individual authorized account and ensures that all your forwarded data stored into your account or not.

Whenever we are creating Thing speak account two types of API keys are generated. First key is known as read API key and second one is known as Write API key, this write API key is important one and also used in Arduino code for writing data to thing speak server. Below diagram shows the general workflow of communication model from sensor devices to Think speak cloud server. All the real time collected patient datas are transferred to Arduino board which is having internal Wi-fi features in it. Then they are transferred to Thingspeak server through ESP8266 Node MCU board in serial Communication way. Then the gathered information is represented in graphical manner using Matlab software which are remotely accessed by the intended Doctor system in cloud.

![Figure 7 Work Flow of the Proposed Model](http://www.webology.org)

6. Data Holder & Data Sources

This is data Holder module which is used to upload patient’s data to local server using IoT based Fog gateways such as mobile, Laptop or Tablet etc. To give more security and
prevent data loss, the owner also keep one copy of the data in their local system and keep duplication of the data to local server. Different sources of the information such as all types of sensors which are explained previously are used for collecting patient health information in 24 hours. And transmits that collected information to Fog gateway using Arduino UNO board. After each transmission sensor energy is analyzed.

### Table 2 Real Time Data Collection from Diabetic Patient

<table>
<thead>
<tr>
<th>Id</th>
<th>Disease</th>
<th>Age</th>
<th>Sex</th>
<th>Temperature</th>
<th>Respiration</th>
<th>Heart Beat</th>
<th>MEMS1</th>
<th>MEMS2</th>
<th>Readmitted Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>50</td>
<td>1</td>
<td>36</td>
<td>12</td>
<td>119</td>
<td>151</td>
<td>173</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>45</td>
<td>0</td>
<td>37</td>
<td>20</td>
<td>116</td>
<td>140</td>
<td>160</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>50</td>
<td>1</td>
<td>29</td>
<td>30</td>
<td>131</td>
<td>150</td>
<td>178</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>34</td>
<td>1</td>
<td>43</td>
<td>12</td>
<td>121</td>
<td>150</td>
<td>160</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>34</td>
<td>1</td>
<td>28</td>
<td>20</td>
<td>123</td>
<td>140</td>
<td>173</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>34</td>
<td>1</td>
<td>34</td>
<td>40</td>
<td>123</td>
<td>162</td>
<td>155</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>45</td>
<td>1</td>
<td>26</td>
<td>45</td>
<td>122</td>
<td>176</td>
<td>161</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>55</td>
<td>0</td>
<td>39</td>
<td>41</td>
<td>125</td>
<td>146</td>
<td>170</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>45</td>
<td>1</td>
<td>33</td>
<td>42</td>
<td>138</td>
<td>144</td>
<td>174</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>45</td>
<td>1</td>
<td>34</td>
<td>48</td>
<td>129</td>
<td>142</td>
<td>149</td>
<td>4</td>
</tr>
<tr>
<td>11</td>
<td>3</td>
<td>46</td>
<td>1</td>
<td>30</td>
<td>11</td>
<td>129</td>
<td>152</td>
<td>154</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>47</td>
<td>1</td>
<td>43</td>
<td>12</td>
<td>129</td>
<td>161</td>
<td>148</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>3</td>
<td>48</td>
<td>1</td>
<td>40</td>
<td>20</td>
<td>129</td>
<td>136</td>
<td>145</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>45</td>
<td>1</td>
<td>32</td>
<td>27</td>
<td>133</td>
<td>138</td>
<td>145</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>45</td>
<td>1</td>
<td>34</td>
<td>28</td>
<td>132</td>
<td>173</td>
<td>175</td>
<td>5</td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>45</td>
<td>1</td>
<td>27</td>
<td>30</td>
<td>136</td>
<td>167</td>
<td>163</td>
<td>5</td>
</tr>
<tr>
<td>17</td>
<td>4</td>
<td>34</td>
<td>1</td>
<td>30</td>
<td>32</td>
<td>138</td>
<td>137</td>
<td>160</td>
<td>6</td>
</tr>
<tr>
<td>18</td>
<td>4</td>
<td>34</td>
<td>1</td>
<td>41</td>
<td>34</td>
<td>100</td>
<td>159</td>
<td>159</td>
<td>6</td>
</tr>
<tr>
<td>19</td>
<td>4</td>
<td>34</td>
<td>1</td>
<td>28</td>
<td>35</td>
<td>122</td>
<td>137</td>
<td>137</td>
<td>6</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>45</td>
<td>1</td>
<td>35</td>
<td>36</td>
<td>116</td>
<td>147</td>
<td>147</td>
<td>3</td>
</tr>
<tr>
<td>21</td>
<td>3</td>
<td>45</td>
<td>1</td>
<td>33</td>
<td>11</td>
<td>121</td>
<td>161</td>
<td>161</td>
<td>3</td>
</tr>
<tr>
<td>22</td>
<td>3</td>
<td>45</td>
<td>1</td>
<td>22</td>
<td>10</td>
<td>124</td>
<td>159</td>
<td>159</td>
<td>3</td>
</tr>
<tr>
<td>23</td>
<td>3</td>
<td>45</td>
<td>1</td>
<td>38</td>
<td>27</td>
<td>98</td>
<td>150</td>
<td>150</td>
<td>2</td>
</tr>
<tr>
<td>24</td>
<td>3</td>
<td>47</td>
<td>1</td>
<td>35</td>
<td>33</td>
<td>103</td>
<td>149</td>
<td>149</td>
<td>2</td>
</tr>
<tr>
<td>25</td>
<td>3</td>
<td>48</td>
<td>1</td>
<td>18</td>
<td>38</td>
<td>99</td>
<td>159</td>
<td>159</td>
<td>2</td>
</tr>
<tr>
<td>26</td>
<td>0</td>
<td>50</td>
<td>1</td>
<td>36</td>
<td>12</td>
<td>118</td>
<td>177</td>
<td>177</td>
<td>1</td>
</tr>
<tr>
<td>27</td>
<td>0</td>
<td>50</td>
<td>1</td>
<td>37</td>
<td>20</td>
<td>120</td>
<td>154</td>
<td>154</td>
<td>0</td>
</tr>
<tr>
<td>28</td>
<td>0</td>
<td>34</td>
<td>1</td>
<td>36</td>
<td>12</td>
<td>113</td>
<td>164</td>
<td>164</td>
<td>0</td>
</tr>
</tbody>
</table>

All the gathered data are represented in table. This is real time sensors data of Diabetic patient at different time periods.

**Fog Computing Layer**

The sources of information coming from wearables components are analyzed here. This layer locally performs data aggregation and analysis tasks. After prediction based on
type of data either normal or abnormal then the collective information are passed to thing speak cloud server for future storage and retrieval purpose. Here Fog server distribute the processing to several fog nodes, hence huge volume of data transfer happened smoothly without any overlapping in that. Using efficient task scheduling algorithms the task distribution happened properly among the fog nodes.

1. Data Aggregation

In this paper, a novel preemptive task distributed scheduling algorithms is discussed, which is used to distribute the task among several fog nodes. After task distributed, next process is data aggregation. This task consists of three main parts: schema mapping, duplicated data removal, and efficient data fusion. Schema mapping will ensure that aggregated data are making sense properly, and also provide data flow properly. Next step is duplication data removal which is used to remove redundant data. Finally relevant data are obtained. By means of this step overlapping can be prevented. Apart from this, to ensure security in Fog layer, false data injection is avoided by implementing local filter algorithms to fog devices. Final component is data fusion in which the whole information’s are gathered and put together as one entity.

2. Data Analyzer and Emergency Sector

Data analyzer and emergency sector module is newly added in Fog layer. In this module, the patient enter into their system using their credentials such as user name and password. After successful login the authorized user will search for data as well as existing patient Records. In this module, the server can perform the following operations such as View All admitted and Emergency Patients details to particular Hospital, and real time health data collected from various sensors.

3. Preemptive Interval Scheduling Algorithms

Preemptive interval scheduling algorithm is used to classify the patient based on health parameters received from different types of sensors and its RSSI (Received Signal Strength) also consider. The sink node analysis the vital sign parameters then it determine patient vital sign information based on Threshold values. Further analysis happened based on the threshold input conditions.
RSSI Data Analysis

![RSSI Data Analyzer Module](image)

**Figure 8 Data Analyzer Module**

<table>
<thead>
<tr>
<th>WBSN</th>
<th>Normal</th>
<th>Abnormal</th>
<th>RSSI value (dBm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature (Tap)</td>
<td>$\text{TL} &lt; \text{Tap} &lt; \text{TH}$</td>
<td>$\text{Tap} &lt; \text{TL}$</td>
<td>$\text{Tap} &gt; \text{TH}$</td>
</tr>
<tr>
<td>Respiration (RSap)</td>
<td>$\text{RSL} &lt; \text{RSap} &lt; \text{RSH}$</td>
<td>$\text{RSap} &lt; \text{RSL}$</td>
<td>$\text{RSap} &gt; \text{TH}$</td>
</tr>
<tr>
<td>HeartBeat (HBap)</td>
<td>$\text{HBL} &lt; \text{HBap} &lt; \text{HBH}$</td>
<td>$\text{HBap} &lt; \text{HBL}$</td>
<td>$\text{HBap} &gt; \text{HBH}$</td>
</tr>
<tr>
<td>MEMS1 (ME1ap)</td>
<td>$\text{ME1L} &lt; \text{ME1ap} &lt; \text{ME1H}$</td>
<td>$\text{ME1ap} &lt; \text{ME1L}$</td>
<td>$\text{ME1ap} &gt; \text{ME1H}$</td>
</tr>
<tr>
<td>MEMS2 (ME2ap)</td>
<td>$\text{ME2L} &lt; \text{ME2ap} &lt; \text{ME2H}$</td>
<td>$\text{ME2ap} &lt; \text{ME2L}$</td>
<td>$\text{ME2ap} &gt; \text{ME2H}$</td>
</tr>
</tbody>
</table>

Where ap -> Active Patient detail, L -> Low value input, H->High value parameter

**Preemptive Scheduling Algorithms for Patient Classifications**

**Input:** Patient Parameters (Tap, RSap, HBap, ME1ap, ME2ap)
Low parameter threshold (TL, RSL, HBL, ME1L, ME2L)
High parameter Threshold (TH, RSH, HBH, ME1H, ME2H)
If (input parameter)

Check($\text{TL} < \text{Tap} < \text{TH} \&\& \text{RSL} < \text{RSap} < \text{RSH} \&\& \text{HBL} < \text{HBap} < \text{HBH} \&\& \text{ME1L} < \text{ME1ap} < \text{ME1H} \&\& \text{ME2L} < \text{ME2ap} < \text{ME2H}$)
Set Flag 1 # Normal Patient

Check (Tap<$\text{TH} \&\& \text{RSap} < \text{RSH} \&\& \text{HBap} < \text{HBH} \&\& \text{ME1ap} < \text{ME1L} \&\& \text{ME2ap} < \text{ME2L}$)
Set Flag 2 # Unnormal Lower level condition

Else check (Tap>TH $\&\& \text{RSap} > \text{TH} \&\& \text{HBap} > \text{HBH} \&\& \text{ME1ap} > \text{ME1H} \&\& \text{ME2ap} > \text{ME2H}$)
Set Flag3 # unnormal upper level conditions.
The above algorithms can be executed in Fog gateways for finding normal and abnormal patients based on Real time health parameters retrieved from the above mention sensors.

**Deep Learning Introduction**

This work is explaining about the concept of innovative set of deep learning algorithms which requires minimum efforts of human involvement. Deep learning architecture has produced tremendous improvement in system performance and it has working with both text data and time series data. If the healthcare applications are powered by big data with intelligent then every processing in healthcare can be automated easily. In this work volume of different data of diabetic patient are collected, hence Big data comes into picture. The collected informations are analyzed by deep learning algorithms to find useful patterns and trends from them. In this way, the deep learning algorithms become most popular in analyzing patient’s data generated from wireless body area networks (WBANs).

1. **Logistic Regression**

   In machine learning, logistic regressions build models of probability with various factors for binary decisions. It gives logical analysis with probability by taking various condition parameters in consideration. The predictive analysis is monitored and updated with interval pre-emptive scheduling.

2. **Support Vector Machine (SVM)**

   This is supervised algorithms used for classification and regression analysis for the collected data of diabetic patient. After giving an SVM model sets of labeled training data for each category, they’re able to categorize new text. A support vector machine takes the collected data points and outputs the hyper plane that best separates the tags. This line is the **decision boundary**: anything that falls to one side of it we will classify as blue, and anything that falls to the other as red.

3. **Decision Tree**

   **Decision Trees** are type of Supervised **Machine Learning** where the data is split based on certain parameters and is going to explain about the output and its corresponding input in the training data. Using algorithmic approach this decision tree identifies the different
way to split the data based on conditions. If the coming output produces extra value to the
decried output means then over fitting problems will happened, it can be used for both
classification and regression tasks. The main goal of this tree is to create prediction model
that predicts the desired output based on classification rule set. This classification is of
two step process. First step is learning step, in this the model is developed based on
training dataset. Second step is prediction step, in this based on classification rule set the
decision happened.

4. Gradient Boost Machine

A created GBM integrates various associative decision trees and makes the final
predictions with the method boosting. These take input from logistic analysis based
decision trees considering various factors like age, critical diseases, time, and severity
conditions.

Cloud Data Center

This data center performs data pre-processing which is used to convert the huge volume
of data collected from Big Data server into user specific format, during workload
execution. Apart from this using data analytics tools filtering of data also processed in
this. All the collected time series data based on patient health status are analyzed properly
and suitable decision making are done by the system automatically. This process is used
to recommends appropriate medication and also enables the suitable Doctors to give
proper check-up based on the continuous opinion of healthcare providers and doctors.
These processed informations are stored into the server for future analysis and admission
prediction happened based on historical analysis of healthcare data.

Experimental Result Analysis

Results of all the collected sensors signals are analyzed using Machine Learning
algorithms such as Logistic Regression, Decision Tree, SVM and Gradient Boost
Machine learning algorithms in Fog gateway. Based on the emergency condition the
particular alert messages as well as mail are sent to respective care giver, Nurses, Doctor
and relatives. Based on their historical data the hospital server do some analyzation and
report the details such as how many times the particular patient readmitted in the
emergency stage and their full history details refer figure 9, for easy and effective cross
checking. This will reduces the waiting time in emergency ward and do the correct
prediction well in advance in critical stage. The figure 10 displays the authorized users
personal information for verification and validations.
Figure 9 Hospital Server to view Emergency Patient Admitted Details

Figure 10 Hospital Server for finding Authorized Users

The comparative analysis of various machine learning algorithms and its graph analysis in display in the below figure 10.
Figure 11 Correlation Values for different sensor inputs

Table 4 Different Machine Learning Algorithms Analysis

<table>
<thead>
<tr>
<th>S. No</th>
<th>Algorithms Name</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>KNN</td>
<td>72.9</td>
</tr>
<tr>
<td>2</td>
<td>SVM</td>
<td>74.03</td>
</tr>
<tr>
<td>3</td>
<td>LR</td>
<td>76.23</td>
</tr>
<tr>
<td>4</td>
<td>DT</td>
<td>70.7</td>
</tr>
<tr>
<td>5</td>
<td>GB</td>
<td>77.3</td>
</tr>
<tr>
<td>6</td>
<td>RF</td>
<td>79.5</td>
</tr>
<tr>
<td>7</td>
<td>Artificial Neural Networks</td>
<td>93.00</td>
</tr>
</tbody>
</table>
This analysis depicts that compare to other algorithms Artificial Neural Network model (Binary Class Sequential Classification) provide 93% accuracy for emergency admission prediction based on the real-time health data gathered from different types of sensors. The data gathering units provide the following Correlation property value for the different type of inputs collected from variety of Body Sensor at different time interval.

**Conclusion and Future Enhancement**

This paper is explained about FoG computing usage in Healthcare applications and also the utilization of machine learning algorithms for predictive analysis. In this work, three machine learning models are involved for predictive analysis of hospital admission based on their health conditions. Individual models are trained properly based on health status of patient data and emergency admission prediction happened based in three different machine learning algorithms such as logistic regression, decision trees and gradient boosted algorithms. Overall, the Random Forest algorithm performed the best when compared to logistic regression, decision trees and other algorithms. Also the machine learning algorithms adapts with preemptive interval scheduling in graph of patients’ status with accuracy. In future the same scenario can be implemented for outdoor environment also and the same system can be deployed to various edge devices and investigate the real-time behavior of the new adaptable system.
Acknowledgments

This work was supported by the Software Engineering Department, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.

References


