

# Feasibility of Complex Operations on Real Time Data Stream at On-Surface Body Sensor for Ubiquitous HealthCare

Chetna Laroija  
Assistant Professor I.T.  
Computer Science Department,  
Jagan Institute of Management Studies,  
Affiliated to GGSIPU Delhi, India

Vijay Bhushan Aggarwal, PhD  
Professor I.T.  
Computer Science Department  
Jagan Institute of Management Studies,  
Affiliated to GGSIPU Delhi, India

## ABSTRACT

Rapid growth in the field of wireless technology, body area network, personal area network and miniaturization in device has led to continuous health monitoring of moving patient which further can result into emergence of technology driven enhancement in contemporary healthcare practices. Can we have an Intelligent Applications in HealthCare (with knowledge base and inference engine) which can work on real time data stream initiating at sensor. For that we need to think of an application which can analyze sensed data stream, based on the past experience or rules specified as its knowledgebase to take decisions with in definite latency or delay. Can we think of an application which could record patient's physiological condition and can automate the dosage induction based on the oddity arising in it? Memory size and energy at sensor provided by different vendors are two major obstacles in the design of such an application. Sensor depletes maximum energy in data transmission which could be reduced in terms of data size and number of communications. In this paper finds three gaps in initiation of such an application namely (i) projections for at sensor UDBMS architecture to upkeep ubiquitous computing, (ii) construction of the SQL code to analyse sensor data stream (based on joint conditional probability) (iii) simulation of above code to compare (CPU time, elapsed time, communication time, communication energy) its execution at sensor with one device at some hop distance. Simulation results have obtained under various length real time sensor data stream are consistent and lead to the deduction that complex operation can be and should be advanced AP or at UI based sensor through SQL queries.

## General Terms

WSN, WBAN, intelligent application, healthcare, PDA, access point

## Keywords

UDBMS architecture, intelligent application, oddity in sensor real time data stream, indexing in UDBMS, MAX Heap with Binary Search

## 1. INTRODUCTION

Our era has immense pressure on quality and quantity of healthcare due to the rise in aging population. To implement healthcare in preventive way and to motivate early detection of diseases ubiquitous computing could be explored. This trend can be captured through the BAN which is mature technology supported by wireless standard IEEE 802.15 launched in November 2007. [1] Wireless body network consists of wearable / on-surface or implantable / in-body (pacemaker) sensors which are distributed on human body to

monitor vital physiological parameters i.e. pulse rate, in-tissue glucose level, body temperature, blood pressure, and even more. This bi-directional body area network is ground for ubiquitous computing in HealthCare domain for in- hospital or at home patient monitoring. Data generated through the these sensors is accumulated into access point or the gateway, which in turn work as repeater to transmit data to doctor's PDA / desktop or the main server through the base station in timely manner before the information conveyed in data becomes obsolete. Raw data generated by these sensors is considerable. Doctors can make decision on the real time stream data communicated to their PDA. WBAN can monitor the impact of one or more than one vital signal on patient body thus can be an effective and efficient solution for early disease detection through long term patient monitoring. With the advancement in the electronics data can be stored or analysed at many locations i.e. PDA, AP, laptop, mobile which are moved even closure to the body sensor. Continuous model of data gathering is suitable to constantly monitor patient's vitals. [2]

Challenges imposed by the ubiquitous healthcare are multidisciplinary. Firstly the wearable sensor to collect patient's physiological data, patient's behaviour. Second the wireless standards to support short range communication with WBAN. Third is intra-body communication between wearable or implantable sensor. Fourth is the way to implement complex operations (through middleware, web-services) to analyze real time data stream from sensors. Fifth is the energy efficient algorithm for complex operations (popularly oddity detection) implementation. Sixth is to move sensor real time data stream Data stream analysis close to the sensor to reduce sensor energy dissipation to transmit data and delay between oddity incidence and decision to work on the criticality indicated by oddity. Seventh is to find the possibility of DBMS to store database at the constraint devices particularly at sensor.

Rest of the paper is structured as follows. Section 2 comprises Literature Survey. Section 3 describes the complex operations. Section 4 describes the Query processing. Section 5 describes the proposed architecture and the algorithm for optimizing communication cost. Finally Section 6 gives Simulation settings, conclusions and future scope of this work. Section 7 gives references.

## 2. RELATED WORK

Work has been done to provide real time feedback to the patient and the doctor. Proposed WBAN architecture is to monitor heart activity through ECG sensor and body motion through several motion sensors. Medical server has many

registered users. Second tier is the personal server implemented on the PDA. This provides user interface and the interface to the sensor and the medical server. Personal server is connected to the medical server through the WLAN and to the sensor through the short range wireless standards. Data stream is processed at medical server [3]

Mark A. Hanson, Harry C. Powell Jr., Adam T. Barth, Kyle Ringgenberg have coined a new architecture. Body area sensor network (BASN) is reported to continuously read quantitative data from a various sensors. Body sensors provide data to the body aggregator PDA. Body area aggregator perform functions i.e. fusing data from sensors in the body, user interface, and bridge BASN to higher-level infrastructures. Data stream analysis is done at servers.

Anis ISMAIL , Abd El Salam AL HAJJAR and Ziad Ismail have proposed a framework to support pervasive computing data analysis and mining. Here devices are connected through 802.11 or bluetooth standards. Principle components of the architecture are sensor devices, user, webpages, web services. Web services and web pages are connected to the ubiquitous sources (i.e. notepad, mobile or PDA) through Internet. Web services can use other web services listed in UDDI (Universal Description, Discovery and Integration – web services standard) [4]. Through web services data analysis functions are called.

Anastasios Zafeiropoulos, Dimitrios-Emmanuel Spanos, Stamatis Arkoulis, Nikolaos Konstantinou, Nikolas Mitrou have reported three layered architecture (i) data layer for data collection and aggregation (ii) processing layer – for integration and processing of sensor data, semantic layer -to provide the user with a way to ask for information retrieval from the underlying Knowledge base through the semantic query. Communication between the processing layer and semantic layer is through XML. [5]

The SOA middleware bridges the gap between the high-level requirements of different applications and the hardware constraints of WSNs. Middleware is responsible to select sensors which are capable to supply data to a query. Context information such as location, energy-status, routing tables of sensor is provided by it. [6]

### 3. GAP AREAS TO IMPLEMENT AT SENSOR COMPLEX OPERATION

In this paper we discuss our finding about the gap areas in order to implement complex operation at sensor device for WBAN. This work is to target an application to monitor patient's vitals rather the patient's environment monitoring. WBAN under consideration consists of implantable or on-surface with or without an UI. Either body sensor with UI or the AP at one hop distance from the sensor device (in case of sensor without UI) is used to invoke for real time sensor data stream analysis. Execution of complex operation at body sensor UI or AP will assist to reduce energy consumption (as much of the energy is dissipated in data stream transmission to the server over Wi-Fi, also energy consumption over long range wireless standards is more when compared to that of short range wireless standards), delay in stimuli generation and action taken to brush-off the affect of the criticality indicated by the stimuli. For the above work to be done there are four gap areas-

Gap 1: A low footprint size – UDBMS with tightly coupled algorithm to implement complex operation is required at the sensor device or at AP

Overall architecture of DBMS is explored with focus on query optimization component and indexing scheme.

Gap 2: To implement algorithm for complex operation in SQL only.

Oddity revelation algorithm in SQL works on the basis of the joint conditional probability rather than the joint probability based oddity revelation algorithm proposed earlier  $Q_k = (ci/WC)$  or  $=(ci+cl)/(WW+WC)$ . Here  $Q_k$  has been computed  $= \{(ci/WC)/(cl)\}$  for any  $k \in DB$  or  $DP$ .  $J_q$  is the oddity if it is not in the  $P_{revo}$ . [7]

Gap 3: To check the feasibility of SQL code.

To check that if it is possible to execute above code to reveal the outlier with in one sliding window.

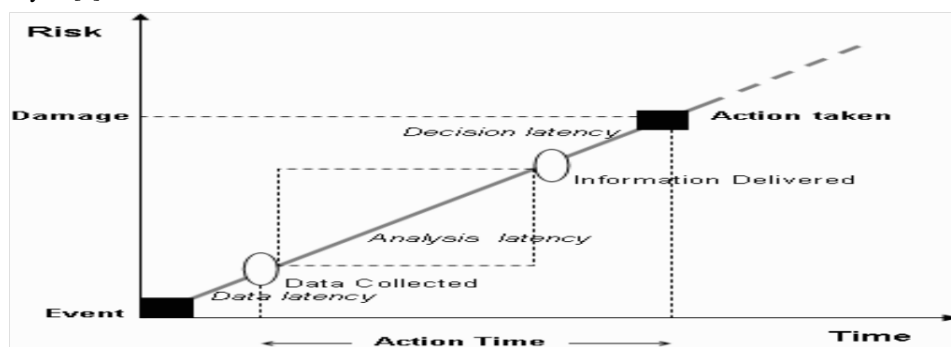


Figure 1: Relation between time when event or Stimuli generated in the sensor network and time of decision or counter action taken [8]

Data latency + analysis latency has to be always less than the sliding window WC [7]

Gap 4: Feasibility of such an SQL code is to be checked in terms of reduction in delay and energy consumption

#### 3.1 Projections for UDBMS Architecture

Memory size and energy at sensors are two major constraints in the design of the sensors UDBMS architecture. MICAz has memory size of 8kb whereas Telos sensor has 10kb RAM. [9]

### 3.1.1 Considerations for UDBMS

#### Architecture at Sensor Device

DBMS architecture components which cause an increase in foot-print size of the DBMS are-

- (i) PL/SQL component
- (ii) Indexes to access records in memory block. We proposed two indexing schemes.

(a) HASHING Index - The order of sensor records are in monotonically increasing order of their timestamp for the window. This ordering constraint of the sensor measurements can be used to create HASHING index. Address of Measure<sub>x</sub> is = F(Measure<sub>y</sub>) + incre, where incre is fixed in array for the sensor data table. These indexes are to be implemented to support equality

JOIN on the measurements from different sensors at a given timestamp to access the health status of the patient based on different biological measures.

(b) MAX HEAP with in Block Binary Search - Index is suitable for the range search and equality search in a table. Max Heap index require  $O(\log_2 n) + k$  search time where n are total number of buckets arranged in the MAX Heap data structure

where each bucket can store k records. Each node in the Max Heap represents the bucket or block.

As an improvement to the Max Heap index proposed is Max Heap with in-Block Binary Search. Search time reduces from  $O(\log_2 n) + k$  for SELECT with range to  $O(\log_2 n) + O(\log$

$\log_2 n)$ . Although with original MAX Heap index SELECT with equality would execute in  $O(\log_2 n) + O(1)$  time as the search time of HASH table is  $O(1)$ . This index is suitable for both SELECT WHERE attribute  $\geq x$  and attribute  $\leq y$ , SELECT WHERE attribute = x

(iii) Implementation of JOIN for execution plan. Memory requirement to run a query depends on the algorithm chosen to implement the operations of the query plan. Tuple – at-a-time strategy to implement joins can save memory consumption for query processing. [10]

(iv) Memory allocation to records in a relation, should be **fixed** not dynamic for this specific applications in HealthCare domain. Only N=10 to 30 records are to be kept by the sensor device before analysis. After analysis the records could be either be transmitted in 50 m range to a resource constraint device or marked deleted.

Memory allocation to any measurement at timestamp Measure<sub>tx</sub> is to be stored at memory address revealed by applying HASH function  $Hf_x(\text{Measure}_x)$ .

(iv) Query optimizer (except Query parser and physical Query plan evaluator)- *Query Optimizer* is outsourced in Antelope. In LittleD query optimization is fixed i.e. parser and laxer are hand written to directly translate SQL code to machine executable byte code. Parsing order is fixed which starts from the FROM clause. [11]

### 3.1.2 Model for UDBMS Architecture

Figure below gives UDBMS model based on the projections in 3.1.1.

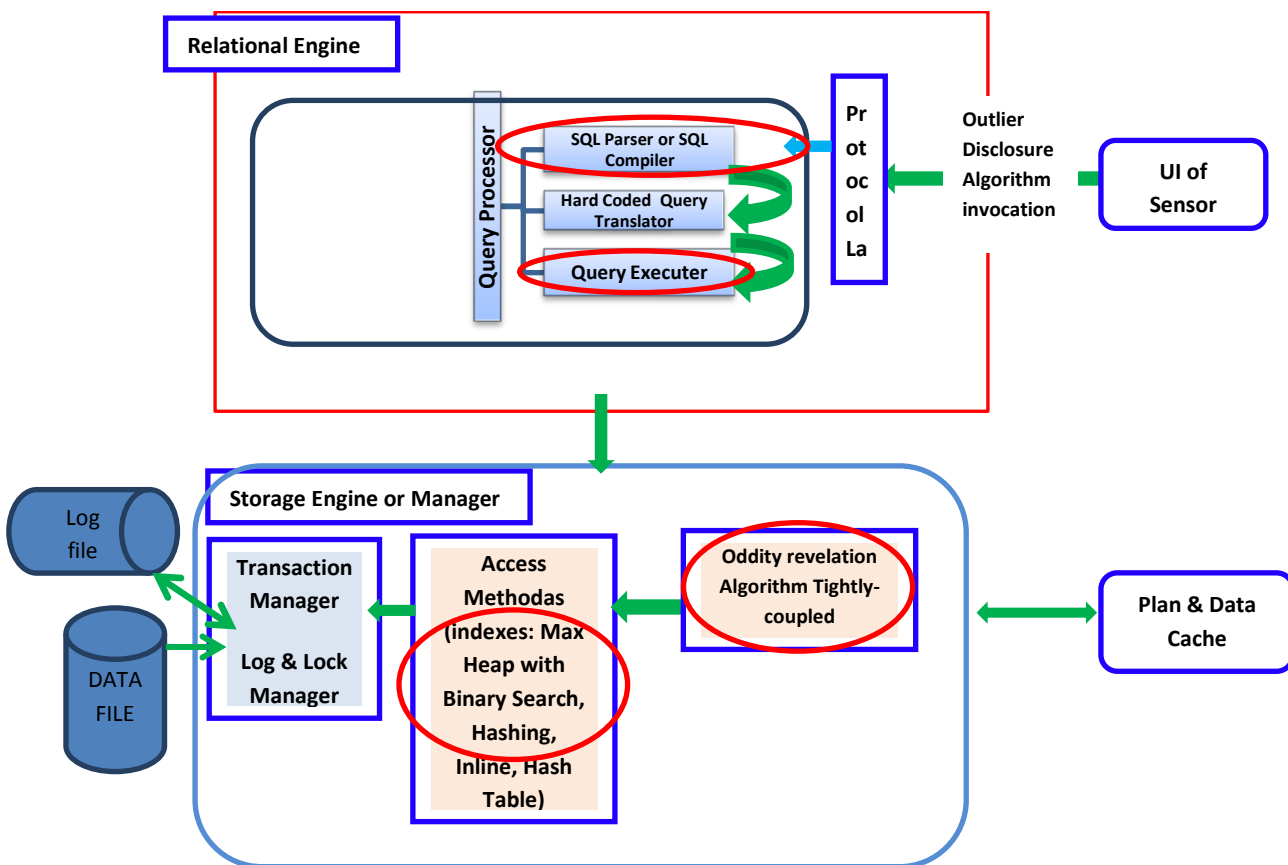


Figure 2: Projections for UDBMS Architecture Model

## 4. SIMULATION & EXPERIMENT RESULTS

### 4.1 Simulation Settings

In order to simulate two algorithms (one based on joint probability and second based on joint conditional probability) to verify the correctness, SQL script is created in SQLite 3.1 on Windows 10 operating system. The input relation size and CPU time is noted. Output of the execution i.e. probability of the outlier record (a combination of glucose level and blood pressure level along with the probability of each of two) is noted to verify the correctness of the algorithm. There could be a provision to plot a graph based on the joint probability of the two body metrics measured (glucose level and blood pressure level) to create a pattern of the sensor measurements over a span of time.

To find the feasibility of the above said algorithms at sensor device or at resource constraint device with in 50 m of range of the sensor device, algorithm is deployed over a WLAN supported network with architecture as given below. Local device has SQL SERVER 2014 on Windows 10 with jdk1.8 and the remote device has SQL SERVER 2008 R2 with jdk1.5 and windows 7.

Network topology is as given in the figure 1 below.

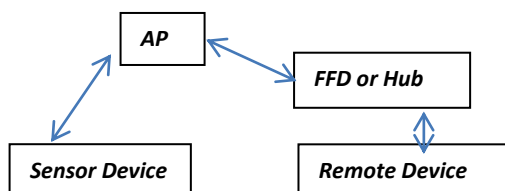


Figure 3: Network Topology for Experiment

It is assumed that the algorithm has to find the outlier record out of every sliding window of size  $n=10$  glucose and blood pressure sensor measurements. Also sensor just keep  $n=10$  records at a time.

Experiments are conducted on glucose level data taken from UCI Machine Learning Repository site. Data set is selected by filtering out data records with glucose measure ignoring the data records with insulin dosage. Dummy data of Blood pressure has been created with both glucose and blood pressure measures at same timestamp.

Table 1: Network setting

Number of devices considered	2
Device 1: Local device	SQL SERVER
Device 2: Remote device	SQL SERVER 2008 R2,
Hop distance of device 1 from AP	1
Hop distance of device 2 from AP	2
WSN standard	Wi-Fi

Table 2: Data set specifications

Number of data points	Average of 4 readings (value -BP/sugar, timestamp) - 10, 25, 50, 100
Sugar value	as per UCI Machine Learning Repository Diabetes data set [12]
Blood Pressure value	Random

### 4.2 Result Analysis

#### 4.2.1 Result based on Response time at Local vs. Remote devices

Algorithms were executed on a device considered to be local device where the data stream is projected to originate. This device (with or without UI sensor) primarily has DBMS to fetch or process data stream stored as database in this DBMS. Remote system is simulated through console java application in jdk1.8 along with DBMS to fetch and process the data stream from local device. A console java application UI is created at the local device too to compare the transmission delay with that of the remote device for  $x$  bits of data stream. For the local system response time of the algorithm includes total of CPU time and waiting time taken when algorithm is posted in the job queue. In the case of remote system response time is sum total of the CPU time, waiting time incurred in the job queue, communication time to transmit result data set from the remote system SQL SERVER application to the jdk1.8 on local system, I/O time.

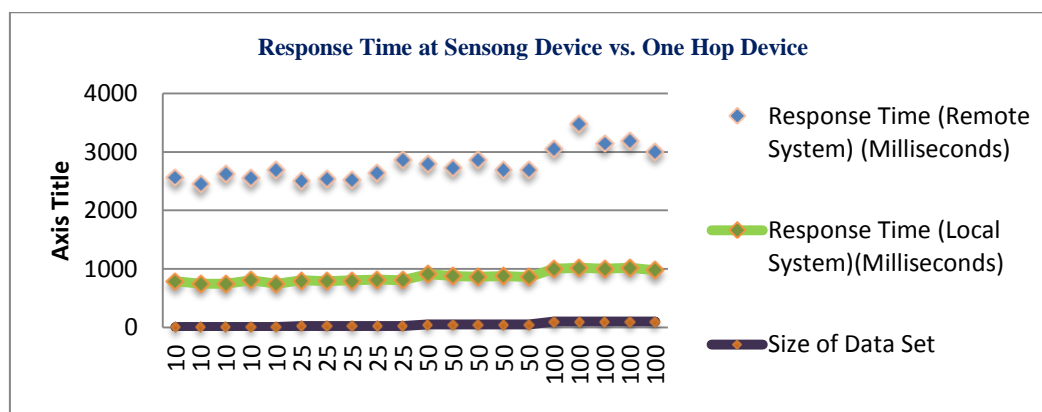


Figure 4: Response time comparison of the algorithm on local and remote machine.

This experiment reveals the benefit to execute oddity revelation at the sensor device rather than at nearby resource constraint device. Deployment of outlier disclosure algorithm at sensor device gives 37% to 45% in terms of time.

#### 4.2.2 Algorithm execution time (CPU) on Local vs. Remote device compared to ping one way time

To support the results obtained from the above experiment, CPU time taken by the algorithm (in microseconds), total time (CPU time + Elapsed time) taken by the algorithm is compared with the time (millisecond) taken by the ping command (ICMP) conveying x no of bits.

Average of min CPU time of algorithm execution has been noticed to be even zero microseconds. In contrast ping has average min time of 1millisec one way when compared for 300 bytes or 2400 bits of data.

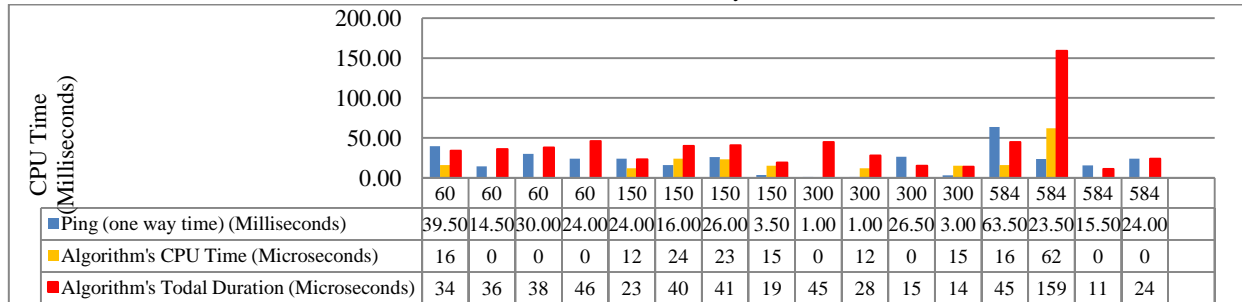


Figure 5: Ping one way time compared to Algorithm CPU and Total duration

#### 4.2.3 Algorithm CPU time compared to ping one way time

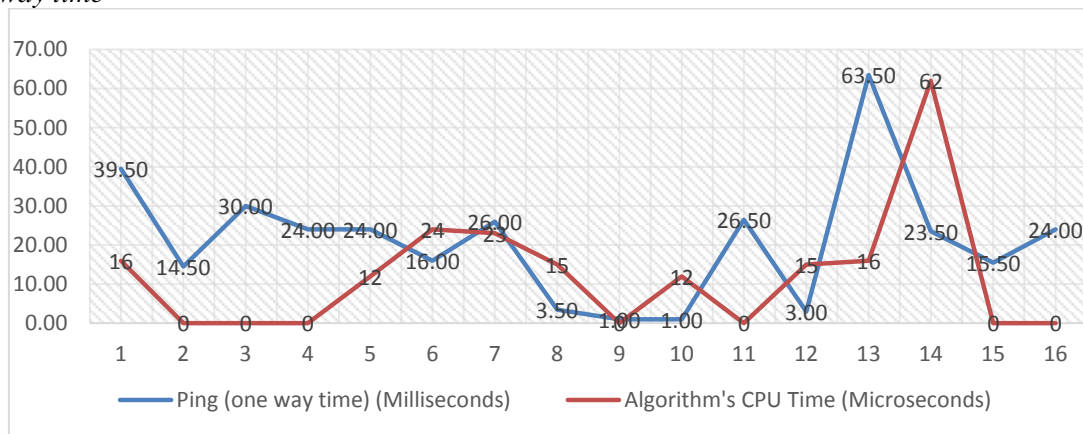


Figure 6: Ping one way time compared to algorithm CPU time

ICMP header has fixed length of 32 bits. Study reveals that even transmission of 32 bits of data takes 1 milliseconds of time compared to minimum recorded nonzero CPU execution time i.e. 12 microseconds. It show 83% gain in algorithm execution at the machines where data originates.

## 5. CONCLUSION

Early detection of the outlier even at sensor device will clearly low down the energy consumption at sensor, round trip propagation delay in outlier detection and the size of asynchronous data transmitted from the sensor. This may also alert the patient and help him to escape critical body situations. Real time simulation results reveals that the algorithm proposed could be implemented at the energy constraint devices as time taken to execute outlier disclosure script on a window size of 50 records is time to ping a machine. Moreover outlier disclosure algorithm implemented through simple SQL queries.

Results have shown that oddity revelation through joint conditional probability based algorithm is possible through the SQL code to be executed without PL/SQL compiler. Feasibility of the oddity revelation algorithm is possible for 10 to 100 records window size.

## 6. REFERENCES

- [1] B. Yu, "Wireless Body Area Networks for HealthCare: A Feasibility Study," University of Florida, Florida, 2009.
- [2] G. V. Luis Javier, T. C. Alicia and B. ., Cláudia Jacy, "Routing Protocols in Wireless Sensor Networks," Sensors, 2009.
- [3] O. Chris, M. Aleksandar, C. Sanders and J. Email, "System Architecture of a Wireless Body Area Sensor Network for Ubiquitous Health Monitoring," Journal of Mobile Multimedia, pp. 306 - 326, 2006.
- [4] A. E. S. A. H. Z. I. Anis Ismail, "A New System Architecture for Pervasive Computing," International Journal of UbiComp (IJU), pp. 22-37, 2011.
- [5] D.-E. S. S. A. N. K. N. M. Anastasios Zafeiropoulos, "DATA MANAGEMENT IN SENSOR NETWORKS USING Semantic Web Technologies," Nova Science Publisher, 2011.
- [6] J.-F. M. P. C. L. Jesús Rodríguez-Molina, "Combining Wireless Sensor Networks and Semantic Middleware for

- an Internet of Things-Based Sportsman/Woman Monitoring Application,” *Sensors*, 2013.
- [7] L. Chetna and A. Dr. V.B., “Intelligence based Outlier Disclosure for UDBMS at Sensor :Wireless Body Sensor Network,” *American Journal of Engineering Research*, pp. 213-220, 2016.
- [8] H. Y. Simon Fong, “The Six Technical Gaps between Intelligent Applications and Real -Time Data Mining Critical Review”.
- [9] P. F. Jose Cecilio, “Wireless Sensor Network: Concepts and Components,” in *Wireless Sensors in Heterogeneous Networked Systems*, Coimbra, Springer International Publishing, 2014.
- [10] A. D. Nicolas Tsiftes, “A Database in Every Sensor,” *SenSys’11*, ACM, pp. 316-329, 2011.
- [11] G. R. Douglas, “LittleD:A Relational Database Management System for Resource Constrained Computing Devices,” UNIVERSITY OF BRITISH COLUMBIA, pp. 1-38, 2014.
- [12] M. Khan, “UCI Machine Learning Repository,” Aug 1993. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/Diabetes>.
- [13] C. L. C. Komalavalli, “ Energy Efficient Protocol in Wireless Sensor Network: Reactive On-Demand Routing,” vol. 2, no. 1, 2010.
- [14] C. L. C. Komalavalli, “Metadata Challenge for Query Processing over Heterogeneous Wireless Sensor Network,” vol. 3, no. 4, 2011.
- [15] E. ZURICH, Switzerland, 2007.
- [16] R. L. Graeme Douglas, “LittleD: a SQL database for sensor nodes and embedded applications,” *SAC '14 Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pp. 827-832, 2014.