# Anomaly detection using Dynamic Sliding Window in Wireless Body Area Networks

Smrithy G S[1], Ramadoss Balakrishnan[2] and Nikita Sivakumar[3]

[1, 2] Dept. of Computer Applications, National Institute of Technology, Tiruchirappalli, India
[3] Dept. of CSE, National Institute of Technology, Tiruchirappalli, India
smrithygs1990@gmail.com[1], brama@nitt.edu[2], nikita.siva@gmail.com[3]

**Abstract.** Anomaly detection is one of the critical challenges in Wireless Body Area Networks (WBANs). Faulty measurements in applications like healthcare lead to high false alarm rates in the system which may sometimes even causes danger to human life. The main motivation of this paper is to decrease false alarms thereby increasing the reliability of the system. In this paper, we propose a method for detecting anomalous measurements for improving the reliability of the system. This paper utilizes dynamic sliding window instead of static sliding window and Weighted Moving Average (WMA) for prediction purposes. The proposed method compares the difference between predicted value and actual sensor value with a varying threshold. If average of the number of parameters exceed the threshold, true alarm is raised. Finally we evaluate the performance of the proposed model using a publicly available dataset and has been compared with existing approaches. The accuracy of the proposed system is evaluated with statistical metrics.

**Keywords**: Anomaly Detection, Dynamic Sliding Window, Prediction, Weighted Moving Average, Wireless Body Area Networks

## 1 Introduction

Wireless Body Area Network (WBAN) [2] is a recent advancement in real time healthcare systems. WBAN offers medical professionals the ease of continuous monitoring of patients by enabling them to do it remotely. In contrast to the traditional healthcare systems, modern healthcare systems utilizing WBAN can effectively reduce prolonged stay in hospital, betterment the patient treatments by continuous monitoring rather than occasional assessments, affordable treatment expenditure etc. Fig. 1 shows a typical system model for WBAN in a healthcare system. It consists of wearable and implanted sensors, a base station and healthcare professionals. A WBAN consists of wireless sensors which are used to monitor vital human body actions (e.g., motion sensors) and parameters (e.g., body temperature, blood pressure, pulse (heart rate), and breathing rate (respiratory rate)). These sensors are either embedded (wearable sensors) or implanted (implanted sensors) in the human body. Implanted devices endure from resource constraints such as battery power, storage etc. On the contrary, wearable devices have fewer resource constraints. The sensors transmit the data to the base station which can be a mobile device (smart phone)

having higher computational power, storage capacity and longer transmission range. The base station analyses the received data (different body parameters) and sends to healthcare professionals that refer to the doctors and nurses or other experts.

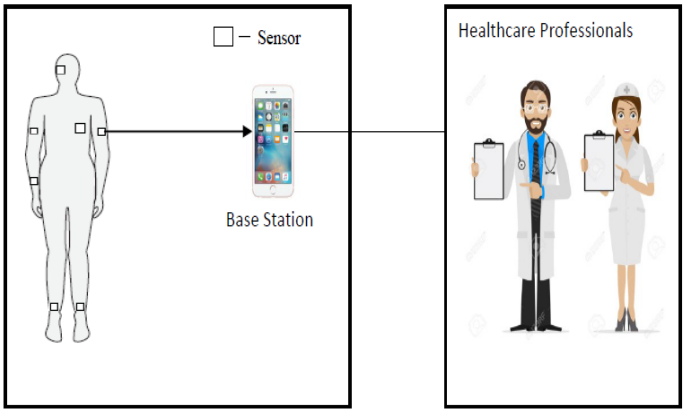


**Fig. 1.** System model for WBAN in a healthcare system.

The accuracy of diagnosis heavily depends on the reliability of sensor data. Sometimes sensor data becomes unreliable due to reasons such as hardware failure, damaged sensors, flaws, mischievous data injection etc. These faulty sensor data may lead to large number of false alarms which are not at all acceptable in a critical scenario like healthcare. Thus there is a need to reduce false alarms by identifying such unexpected observations for reliable monitoring system. Haque et.al [1] proposed an algorithm for anomaly detection in wireless sensor networks for healthcare. The major limitation is they used static size sliding window of historical data for prediction. The proposed work in this paper uses dynamic sliding window to reduce the overhead of considering huge volume of historical data for prediction.

### 1.1 Major Contributions

The contributions of this paper are highlighted as follows:

1. We propose an algorithm for identifying anomalous sensor measurements in WBAN using dynamic sliding window and WMA for prediction.
2. The performance of the proposed method is compared with existing methods [1] in terms of statistical metrics.

### 1.2 Paper Organization

The rest of this paper is organized as follows. Section 2 covers dynamic sliding window. Section 3 describes about WMA for prediction purposes. Section 4 details

the proposed work. Section 5 provides experimental results and discussion. Section 6 concludes the paper.

## 2 Dynamic Sliding Window Approach

Prediction models usually make use of large amount of historical data or make use of sliding windows with static size. In [3], the authors proposed a dynamic sliding window for traffic prediction in a cloud computing environment. But no work has ever been reported which makes use of a dynamic sliding window for prediction in a healthcare environment. This paper makes use of dynamic window concept for prediction of physiological parameter value for comparison purposes.

**Input:**
$\mathbf{CurSW_{\mu}}$: Mean of Current Sliding Window
$\mathbf{PrevSW_{\mu}}$: Mean of Previous Sliding Window
$\mathbf{CurSW}$: Current Sliding Window
$\boldsymbol{\alpha}$: Significance Level
**Output:**
Successor Sliding Window Size, $\mathbf{SucSW_{size}}$

```
1   Procedure Successor_Sliding_Window_Size
2   Begin
3     SucSW_size = find.size(CurSW)
4     Variation_avg = [(CurSW_μ^2 − PrevSW_μ^2) / (CurSW_μ . PrevSW_μ)]
5     If (Variation_avg > (1 + α))
6       Size = Variance_max / Variance
7       If(CurSW_μ > PrevSW_μ)
8         SucSW_size = SucSW_size + Size
9       Else
10        SucSW_size = SucSW_size − Size
11      End If
12    End If
13    return SucSW_size
14  End
```

The algorithm takes as input mean of the current sliding window, mean of previous sliding window, current sliding window and a significance level. The algorithm predicts the size of the successive sliding window based on the variance between the predecessor sliding window and current sliding window. A lager variance indicates a large variation from the mean and a smaller variance indicates a value closer to the

mean. The variance of the extreme values of a window are considered as the maximum variance in this algorithm [3]. Significance level is Type 1 error which is the probability of rejecting the null hypothesis when it is true. As a preliminary work, we are sticking for a significance value of 0.05. The algorithm starts with finding the size of the current sliding window and is stored in $SucSW_{size}$ . In step 4, the algorithm finds the average variation between the current sliding window and the predecessor sliding window. If the variation is greater than the threshold which is $(1+\alpha)$, the value which should be either added or subtracted with the current sliding window to get the new sliding window size is calculated as shown in step 6. If the mean of the current sliding window is greater than the predecessor sliding window, $Size$ is added to $SucSW_{size}$ , otherwise $Size$ is subtracted to $SucSW_{size}$ to get the new sliding window size of the successor window.

## 3 Weighted Moving Average (WMA)

A Moving Average (MA) [4] is a statistical technique to analyze data points by creating sequence of averages of any subset of numbers. It is also referred as rolling average, running average, moving mean, and rolling mean. Moving Average methods is simple and low complex compared to other complex techniques such as Autoregressive Integrated Moving Average (ARIMA), Neural Networks etc which demands large volume of data. The major advantage of using MA techniques is it can be used for short range time series data .Different types of moving average are Simple Moving Average (SMA), Cumulative Average (CMA) and Weighted Moving Average (WMA). A SMA is the unweighted mean of n past data points. A CMA is the running means of an ordered data stream. A WMA [5] assigns weights to data points that decreases in an arithmetic progression such that the most recent data point in the sample window will get the highest weight. Previous studies have used WMA for static size sliding windows. But no work has been reported that uses WMA in dynamic size sliding windows in WBAN scenario. In this paper, we use Weighted Moving Average for prediction purpose. The mathematical expressions for WMA in this context are explained in this section.

The initial computation of WMA for the first window at time t can be expressed as follows:

$$WMA_t = \frac{\sum_{i=1}^{t} W_i D_i}{\sum_{i=1}^{t} W_i}, W_1 < W_2 < \cdots < W_t \quad (1)$$

where $W_i$ = Weight of $i^{th}$ data point and $D_i$ = Data point at time $i$.

For calculating WMA across successive values (i.e., the sliding window size can be either expanded or diminished according to the output from the algorithm explained in section above) , the following expression can be used

$$WMA_{t+1} = \frac{\sum_{i=\left((t+1)-SucSW_{size}+1\right)}^{t+1} W_i D_i}{\sum_{i=\left((t+1)-SucSW_{size}+1\right)}^{t+1} W_i}, W_{((t+1)-SucSW_{size}+1)} < \cdots < W_{(t+1)} \quad (2)$$

where $\mathrm{SucSW_{size}}$ = Successor Sliding Window Size.

## 4 Proposed Anomaly Detection Algorithm

The main objective of the proposed algorithm is to detect anomalies and to reduce false alarms. In this model we are considering $N$ parameters. The algorithm takes as input actual value of $i^{th}$ parameter at time $t$, predicted value of $i^{th}$ parameter at time $t$, current sliding window of $i^{th}$ parameter. We initialize a counter $Pos$ that represents the benign parameter count, $PF_i$ and $NF_i$ are set to zero.

**Input:**
$\boldsymbol{N}$: Number of parameters
$\boldsymbol{AValue_i}$: Actual Value of $\boldsymbol{i^{th}}$ parameter at time $\boldsymbol{t}$
$\boldsymbol{PValue_i}$**:** Predicted Value of $\boldsymbol{i^{th}}$ parameter at time $\boldsymbol{t}$
$\boldsymbol{W_i}$**:** current Sliding Window of $\boldsymbol{i^{th}}$ parameter at time $\boldsymbol{t}$**,** $(i = 1,2, \dots, N)$
$\boldsymbol{Pos}$: Benign Parameters Count, Initially $\boldsymbol{Pos = 0}$
$\boldsymbol{PF_i}$: Positive flag for benign parameter *(i=1,2,…,N)*, Initially set $\boldsymbol{PF_i = 0}$
$\boldsymbol{NF_i}$: Negative flag for abnormal parameter *(i=1,2,…,N)*, Initially set $\boldsymbol{NF_i = 0}$
**Output:**
True sensor alarm or false sensor alarm

```
1   Procedure Proposed_Anomaly_Detection
2   Begin
3     For(i = 1 to N)
4       n = find.size(W_i)
5       For(j = 1 to n − 1)
6         TH_i = find.standarddeviation(W_i[j])
7       End For
8       diff_i = |AValue_i − PValue_i|
9       If(diff_i ≤ TH_i)
10        Retain AValue_i
11        Pos = Pos + 1
12        PF_i = 1
13      Else
14        NF_i = 1
15    End For
16    If( Pos ≥ ( ⌊ (Σ_{i=1}^{N} N) / N ⌋ ) )
17      True Sensor Alarm
18    Else
19      False Sensor Alarm
20      Update all AValue_i = PValue_i in W_i at
          time t with NF_i = 1, ∀i, i = 1, 2, … , N
21    End If
22  End
```

Initially the algorithm finds the size of sliding window of the $i^{th}$ parameter which is represented as $n$.Thus a sliding window $W_i$ with size $n$ has $n$ elements in it. Then the algorithm computes the standard deviation of window $W_i$ for $n-1$ elements which is taken as the threshold value $TH_i$ for the subsequent steps. The absolute difference between actual value and predicted value of $i^{th}$ parameter is represented as $diff_i$. If the absolute difference value is less than or equal to threshold, then actual value is retained in the sliding window and $Pos$ is incremented by one and set positive flag $PF_i$ for benign parameter $(i=1,2,\ldots,N)$ to one. Positive flag for benign parameter $(i=1,2,\ldots,N)$. Otherwise negative flag $NF_i$ to one. Repeat steps 3-15 for all the N parameters. If the positive counter is greater than or equal to the floor value of mean of total number of parameters N in Step 16, then raise true sensor alarm. Otherwise false sensor alarm is generated. Step 20 updates the actual sensor values of all the parameters with negative flag set to 1 with predicted sensor value for future processing.

## 5 Experimental Results And Discussions

The experiment were conducted on a performed on a desktop machine with an core i7-2600 processor with 3.40GHz, windows 7 Professional (64-bit) operating system and 8 GB RAM.

### 5.1 Dataset Organization

For evaluating the proposed work, we have used a publically available dataset from Multiple Intelligent Monitoring in Intensive Care (MIMIC) database (MIMIC DB datasets 221) of Intensive Care Unit patients [6]. MIMIC DB datasets 221 covers logs from 1995-05-18 ,08:00 am to 1995-05-19, 07:33 am with five physiological parameters namely Arterial blood Pressure (ABP), Heart Rate (HR), Pulse, Respiration, Oxygen Saturation (SpO2).

### 5.2 Results

In our experimentation, an anomaly is said to be detected if an alarm is raised in a window containing the anomaly. For experimentation purposes we injected anomalies randomly. For prediction of dynamic window size, we set the significance level $\propto$=0.05 for test runs. In contrast to [1], we utilize dynamic sliding window for prediction purposes which helps to reduce the overhead of considering huge volume of historical data for prediction purposes.
Based on experimentation, we are considering a total of 10449windows. Initially, the sliding window size for each parameter is fixed as 100. Then the model automatically calculates the dynamic sliding window size based on the variance of current and predecessor sliding window. Out of 10449windows, we randomly injected anomalies

in 1449 windows. Thus we have 9000 benign windows out of which 8636 have been correctly classified and 1449 anomalous windows out of which all have been correctly classified. The proposed method correctly classified all the anomalous windows, but misclassified 364 benign windows as anomalous. The accuracy results of the proposed algorithm are based on the following statistical metrics.

$$True\ Positive\ Rate\ (TPR) = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$False\ Positive\ Rate\ (FPR) = \frac{False\ Positive}{False\ Positive + True\ Negative}$$

$$True\ Negative\ Rate\ (TNR) = \frac{True\ Negative}{True\ Negative + False\ Positive}$$

$$False\ Negative\ Rate\ (FNR) = \frac{False\ Negative}{True\ Positive + False\ Negative}$$

Table 1 reflects the overall accuracy statistics for the proposed algorithm when compared with existing approach [1], Mahalanobis Distance (MD) [7], Linear SVM [8] and J48 [9].

**Table 1.** Comparison of proposed method with existing methods

| Metrics | Proposed Method | Existing method [1] | SVM [8] | J48 [9] | MD [7] |
|---|---|---|---|---|---|
| TPR | 100% | 100% | 100% | 100% | 67% |
| FPR | 4.04% | 5.08% | 20% | 30% | 36% |
| TNR | 95.96% | 94.92% | 80% | 70% | 64% |
| FNR | 0% | 0% | 0% | 0% | 33% |

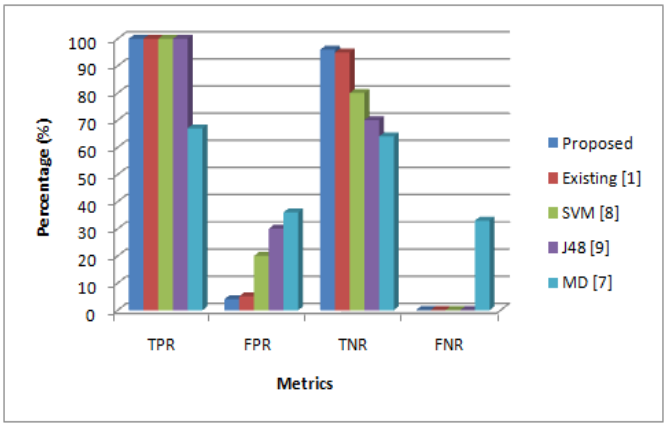


**Fig. 2.** Performance of the proposed method with existing methods.

Fig. 2 depicts the performance of the algorithm in terms of specified statistical metrics for *MIMIC DB datasets 221*. Thus it shows the higher accuracy of our proposed method.

## 6 Conclusions

The proposed method used dynamic sliding window instead of static sliding window which mitigated the overhead of considering huge volume of historical data for prediction purposes. For predicting the sensor value, we used Moving Average (WMA) which is a simple and efficient technique for short range predictions. The proposed method was evaluated with MIMIC DB datasets 221. The proposed method achieved 100% TPR (Detection Rate) which is same as existing method [1], but reduces the FPR by 20.47% when compared with existing method [1].

**Acknowledgments.** This research work was supported by Department of Electronics & Information Technology (DeitY), a division of Ministry of Communications and IT, Government of India, under Visvesvaraya PhD scheme for Electronics & IT.

## References

1. Haque, Shah Ahsanul, Mustafizur Rahman, and Syed Mahfuzul Aziz. "Sensor anomaly detection in wireless sensor networks for healthcare." Sensors 15.4 (2015): 8764-8786.
2. Li, Fagen, and Jiaojiao Hong. "Efficient Certificateless Access Control for Wireless Body Area Networks." IEEE Sensors Journal 16.13 (2016): 5389-5396.
3. Dalmazo, Bruno L., João P. Vilela, and Marilia Curado. "Online traffic prediction in the cloud: a dynamic window approach." In Future Internet of Things and Cloud (FiCloud), 2014 International Conference on, pp. 9-14. IEEE, 2014.
4. C. Murphy, "Moving averages tutorial," http://courses.jmsc.hku.hk/jmsc7008spring2012/files/2010/02/MovingAverages.pdf.
5. S. Dash, "A comparative study of moving averages: simple, weighted, and exponential,"http://www.tradestation.com/education/labs/analysisconcepts/a-comparative-study-of-moving-averages.
6. PhysioNet. Available online: http://www.physionet.org/physiobank/database/mimicdb/ .
7. Salem, O.; Guerassimov, A.; Mehaoua, A.; Marcus, A.; Furht, B. Anomaly Detection in Medical Wireless Sensor Networks using SVM and Linear Regression Models. Int. J. E-Health Med. Commun. 2014.
8. Salem, O.; Guerassimov, A.; Mehaoua, A.; Marcus, A.; Furht, B. Sensor Fault and Patient Anomaly Detection and Classification in Medical Wireless Sensor Networks. In Proceedings of 2013 IEEE International Conference on Communications (ICC), Budapest, Hungary, 9–13 June 2013; pp. 4373–4378.
9. Liu, F.; Cheng, X.; Chen, D. Insider Attacker Detection in Wireless Sensor Networks. In Proceedings of 26th IEEE International Conference on Computer Communications, Anchorage, AK, USA, 6–12 May 2007; pp. 1937–1945.