A Machine Learning Based Hybrid Approach for Fault Detection in WSN

Linta Aniyan[1], Ann Nita Netto[2]
[1]M.tech(Scholar), Communication Engineering, Department of ECE, Sree Buddha College of Engineering, Pathanamthitta, Kerala, India
[2]Assistant Professor, Department of ECE, Sree Buddha College of Engineering, Pathanamthitta, Kerala, India

***Abstract*— One of the most convenient solutions for detecting the failure in WSNs is the use of machine learning. Since WSNs have limited resources and are usually deployed in inaccessible and autonomous environments, each node in the network must be monitored to avoid adverse effects of faulty nodes on normal network operations. Fault detection in WSN is a challenging problem. The existing approaches for diagnosing faults in sensor networks leads to high complexity and low precision. A new fault detection mechanism based on Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) classifiers are proposed. In order to tune the classifier, feature selection method based on correlation is used so that the important or best features can be selected. Finally, the classification is done by the hybrid method that is combining the prediction score of both Support Vector Machines and K-Nearest Neighbor classifier. The addition of feature extraction will improve the fault detection performance in terms of accuracy and detection rate. So this method outperforms all other conventional methods.**

***Keywords*—Support Vector Machine (SVM); K-Nearest Neighbor (KNN); Detection Accuracy; False Positive Rate.**

I. INTRODUCTION

Wireless sensor networks (WSNs) enable new applications and require a non-conventional model for protocol design due to several constraints. A WSN can be defined as a network of nodes, which can sense the environment and transfer the information gathered from the monitored field through wireless links. Wireless communications are used as a medium for communicating between the nodes. It forms an ad-hoc network with peer-peer communication. Wireless sensors are low power devices with limited computational power, memory, battery, and storage. They are deployed in hostile and harsh conditions to report time-critical events such as landslide monitoring, agricultural monitoring, military operations, infrastructure monitoring, scientific data collection, intruder detection system, navigation, and environmental monitoring.

Smart sensor nodes are low power devices accountable for tight communication, computation, and storage limitation. Large-scale deployment of low-cost sensor nodes in uncontrolled, harsh environments is the inherent property of WSNs. Due to this, they are susceptible to frequent and unexpected errors. The faults in WSN may be due to hardware or software failure. These faults result in erroneous results in normal operation. Most probably the sensor nodes can become faulty and unreliable. The normal operation of a WSN suffers from faulty data since it decreases the judgment accuracy of the base station, it increases the traffic in the networks, and it wastes much-limited energy. Also, the sensors are generally used to control actions, where sensor faults can cause catastrophic events. The erroneous data from faulty sensors might result in wrong interpretation or undesirable alarms. It may lead to life-threatening events to occur as a significant percentage of sensor networks will be involved in safety, critical applications.

For detecting faults in WSNs, the use of machine learning seems to be one of the most suitable solutions. Machine Learning is a system that can study from example through self-improvement. It is an application of artificial intelligence which provides the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning algorithms have been widely used in various fields of research. If collected data from sensors are assumed to be faulty in the presence of faults. Therefore, system normal and faulty behaviour can be modelled from collected data using the machine learning algorithms during the training phase and recognized at runtime. There are various machine learning algorithms for fault detection.

The rest of the paper is organized as follows: Section II is the literature review of this topic. Section III gives a brief idea about different types of fault taxonomy in WSN. The proposed fault detection technique is presented in section IV. Simulation results are given in section V. Finally, conclusions are drawn in section VI.

II. LITERATURE REVIEW

The centralized approach is one of the most common solutions to recognize the faulty sensor node in WSNs. A centralized robust fault detection algorithm is presented in [2] to identify soft faulty sensor node present in the network. In the centralized approach, the central node is responsible for identifying the faulty sensor nodes in WSNs. An HMM is a statistical model in which the system being modelled is assumed to be a Markov process with unknown parameters, and the challenge is to determine the hidden parameters from the observable parameters [4].

A Distributed Fault Identification (DFI) algorithm based on neighbor coordination approach is highlighted in [6]. The detection accuracy and false alarm rate of this algorithm degrade for higher fault probability. In [5], the detection technique is performed at the cloud. The advantage of cloud is especially the potential of computation, the massive storage and software services. A distributed technique SODESN is presented in [8]. It performs well for anomaly detection even in the presence of multiple faults.

III. TYPES OF FAULTS IN WSN

Faults are extremely local in the network and it only affects a few components of the network. Generally, the entire WSN is not faulty. Faults only affect a small number of

components of the network instead of the entire network. Due to the increasing reliability of the networks, the growth of the faults is slower than that of network size. Hence, the detection has to be based on the locality than the entire global system. Therefore, by fault locality, faults can be broadly classified into two categories: 1) Data-centric and 2) System-centric.

Let the data originating from a sensor node be modelled as a time series, *d(n; t; f(t))*, where *n* is the node id, *t* is the instant of time in which the value was sensed, and *f(t)* represents the value sensed by node *n* during the time *t*. The *f(t)* can be modelled as $\alpha + \beta x + \eta$; Where α an additive constant called offset, β is a multiplicative constant called gain, *x* is the non-faulty sensor value at time *t*, and η is the noise in the data. In an ideal case, *f(t)* will be *x* but in real-world cases a fault-free node will have $f(t) = x + \eta$.

A. Data-Centric Perspective

In data- centric perspective, characteristics of sensed data are considered for determining fault. They are also called as soft faults. It is classified into different categories:

*1) Offset Fault***:** Offset fault refers to a deviation in sensed data by an additive constant from the expected data. This might occur due to the improper calibration of the sensor. An offset fault is modelled as

$$x' = \alpha + x + \eta$$

where $x' \in f(t)$ and α is the constant value that gets added to the normal reading.

2) Gain fault: Gain fault occurs when the rate of change of sensed data does not match with expectation over an extended period of time. Thus a constant value gets multiplied to the non-faulty sensor data. Gain fault is modelled as

$$x' = \beta x + \eta$$

where $x' \in f(t)$ and β is the constant value that gets multiplied to the normal reading.

3) Stuck-at fault: A fault is said to be a stuck-at fault when the difference or the variance of data from the data series of a node is zero. A stuck-at fault can be either transient or persistent. A stuck-at fault is modelled as

$$x' = \alpha$$

where $x' \in f(t)$ and α is the constant value that is sensed.

4) Out of bound fault: A fault is said to be an out of bound fault if the data lies beyond the thresholds for the problem requirement. A node is said to have out of bounds error if $x' > \theta$ and $x' < \theta_1$ where $x' \in f(t)$, θ and θ_1 are the application thresholds.

5) Data loss fault: A fault is said to be data loss fault if sensed data is missing from the time series for a given node. This means that the sensed data is a null value. If $f(t) = \Phi$ & $t > \tau$, where Φ is null set and τ is the maximum time required for sensing, then it is called data loss error.

6) Random Fault: Random fault is an instant error, where data is perturbed for an instance. It is defined as various negative or positive fast peaks which can affect the data.

B. System-centric perspective

In system-centric perspective, characteristics and properties of the system that is used in WSN are considered for determining fault. They are also called as hard faults. It can be categorized into different categories:

1) Calibration Fault: Calibration is a major cause for faults in WSN. Calibration errors give rise to gain faults and offset faults.

2) Battery failure: Battery failure is the main cause of faulty data. Transmission of faulty data by the sensors is mainly due to the depletion of batteries.

3) *Hardware failure:* Communication and hardware failures occur due to the impairment of hardware components of WSN. These are mostly permanent faults that require the replacement of faulty hardware. As WSNs are deployed in harsh conditions hardware errors are quite frequent.

IV. PROPOSED SYSTEM MODEL

This section deals with the classifiers used in this research. Here SVM and KNN technique are applied to classify received sensor data.

A. Classification

Classification is the problem of identifying to which of a set of categories a new observation belongs, on the basis of training data samples. Classification appears as an appropriate technique for decision making.

1) Support Vector Machine

A region separation algorithm called SVM (Support Vector Machines) is used for classification. It consists of finding an optimal hyperplane that separates the data into two classes. The principle of this technique is defining a decision function $f: X \rightarrow \{-1,1\}$, which have a simple set of data $\{(x_i, y_i);\ x_i \in X$ and $y_i \in (-1,1)\}$. For every new point $x \in X$, this decision function allows to predict its belonging to the right class ((-1) or (+1)). SVM is applicable for both linear and non-linear classification.

In the case of linear classification, this algorithm finds a hyperplane that separates at best the samples of two classes. The function f is linear in x_i with the following general form: $f(x_i =< w,\ x_i >) + b$.

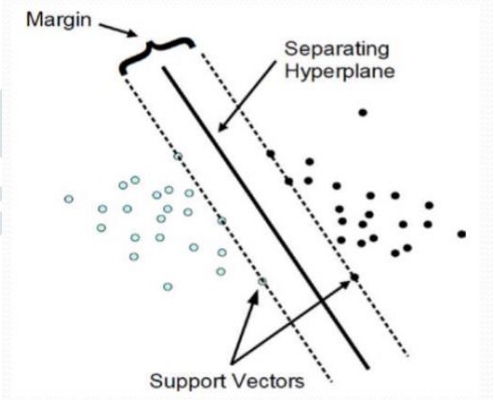

Figure 1: Linear classification

In the case of nonlinear classification, the separator hyperplane of the previous section is not valid. Thus, non-linear SVM is applied. The basic idea is to find a space with the biggest dimension where the projection of examples are linearly separable such as Hilbert space H based on a scalar product that can be replaced by a kernel function of the starting space (space of observations).

2) K-Nearest Neighbor

A K-Nearest Neighbor (KNN) is also known as instance-based learning in comparison to model-based learning because it is not studying any model. It is an effective method, but simple

for classification and is able to solve complex problems. Basically, the training process is learning all the training data. For a data record *t* to be classified, its k-nearest neighbors are taken, and this forms a neighbourhood of t. Usually, majority voting among the data in the neighbourhood is used to determine the classification for t with or without consideration of distance-based weighting. However, to apply KNN we need to choose an appropriate value for *k*, and the success of classification is dependent on this value.

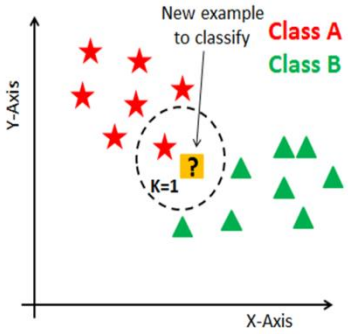

Figure 2: KNN

K-Nearest Neighbor classifiers are based on closeness. When given an unknown tuple, a k-nearest neighbour classifier searches the pattern space for the k training tuples that are closest to the unfamiliar tuple. Closeness is defined in terms of a distance metric such as Minkowski distance. To predict a new data point, we found the closest K neighbors from the training set and let them vote for the final prediction. Voting among these data is used to decide the classification for the data point.

B. Proposed Scheme

In the proposed system, classification is done by the hybrid method that is combining the prediction score of both SVM and KNN classifier. The proposed data learning solution is based on the data learning technique. A decision function is used in real time to classify any new data. A labelled dataset is used as a learning database. The labelled WSNs dataset prepared consists of a set of sensor measurements where different types of faults are injected into it. Data consists of temperature measurements.

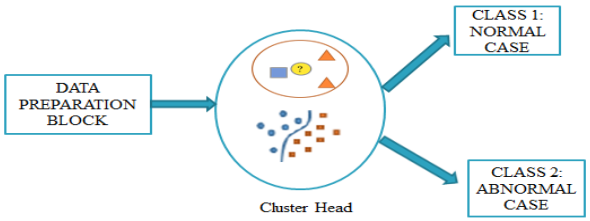

Figure 3: Proposed system fault detection

For each new data measurement, a new observation vector is constructed by a data preparation block. Then the SVM and KNN techniques are applied to this new observation. It belongs to the normal data if the result is positive and otherwise it is considered as a faulty case.

V. SIMULATION RESULT

The proposed technique is evaluated and, it is compared with the most recent fault detection techniques in WSNs. The performance like detection accuracy and false positive rate are analyzed by simulation of the algorithm in MATLAB. This comparison is based essentially on two measures. The first metric is the Detection Accuracy (DA) which is defined as:

$$DA = \frac{\text{Number of faulty nodes detected}}{\text{Total number of faulty nodes}}$$

The second comparison measure is called the False Positive Rate (FPR). False positive rate is the proportion of normal nodes (non-faulty) that are reported as faulty, and it is also known as the false alarm rate (FAR). It is the ratio of the non-faulty nodes diagnosed as faulty to the total number of fault free nodes. It is defined as:

$$FPR = \frac{\text{Number of non} - \text{faulty node diagnosed as faulty}}{\text{Total fault free nodes}}$$

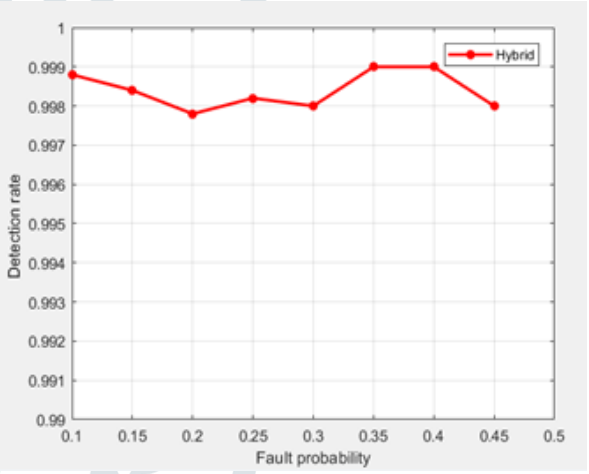

Figure 4: Detection accuracy of the hybrid approach

The hybrid approach shows higher DA as compared to the other techniques. Its contribution is important in case of big fault probability 35% to 50%. However, it provides a similar DA compared to existing fault detection technique in case of a small rate of fault probability 5% to 15%. This fact makes hybrid method the most effective technique of fault detection in WSNs in case of deploying sensor in hazardous or unmonitored environments. From figure 5, it is clear that the hybrid technique shows higher detection accuracy.

The HMM technique's performance decreased once the faults were increased. When the number of faults increases, there will be more risk of having faulty data which closely looks like the data of normal functioning. These data are called bordering data in the field of classification and data analysis. Often, these data are the cause of recognition failure.

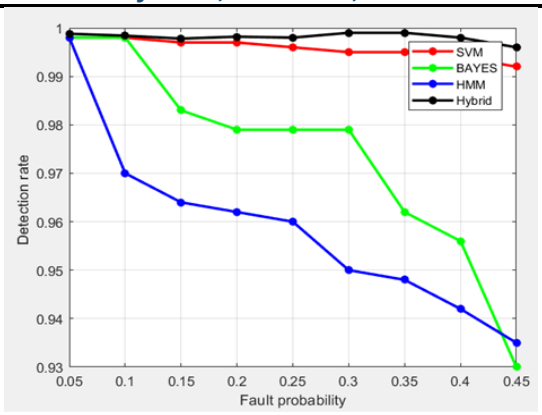

Figure 5: Detection accuracy of Hybrid approach compared to SVM and Bayes

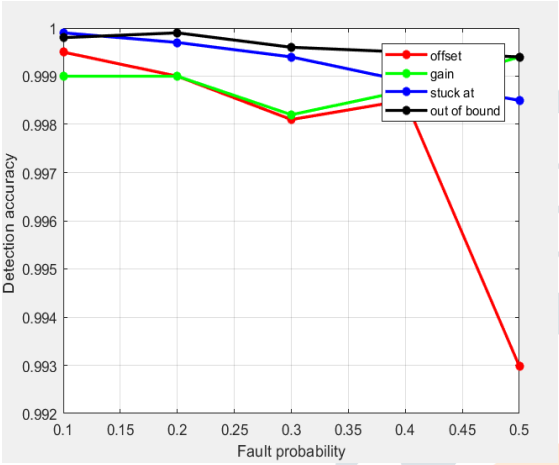

Figure 6: Detection accuracy of Hybrid approach according to fault type

For a learning base retrieved in a random way from the original database. This effect has more influence on the results of classification and therefore also on fault detection. For these reasons, the detection rate decreases when the fault rate increases. Nevertheless, SVM and KNN statistical learning approaches have shown more resistance to this effect.

The DA of the proposed technique is in the average of DA after injecting each type of faults on our dataset. These faults described previously are: offset, stuck-at, out of bounds, gain and random fault. A comparison of DA according to the fault type is given in Figure 6.

In Figure 7, a comparison of FPR between Hybrid approach, SVM technique, Bayes, cloud, SODSEN, and HMM is given.

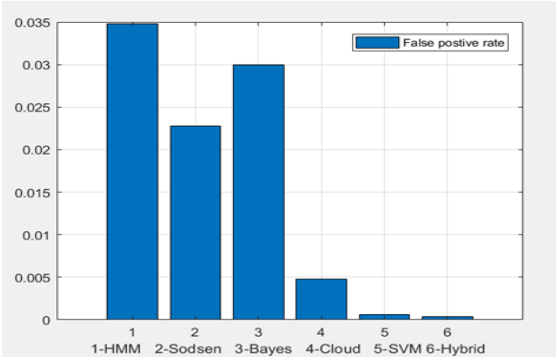

Figure 7: False Positive Rate

It is clear that the hybrid approach provides a significant and important improvement of False Positive Rate as compared to others. From the simulation result, it is evident that the performance of the algorithm is much better as compared to other fault detection techniques.

VI.CONCLUSION

Since WSNs have limited resources and are usually deployed in inaccessible, uncontrolled, and autonomous environments, each node in the network must be monitored to avoid adverse effects of faulty nodes on normal network operations. Indeed, the whole process is executed at the sink node where there is no problem of resource limitation. After establishing the decision function it is sent from the sink node to the cluster head. In this paper, a fault diagnosis mechanism based on combining SVM and KNN classifier is used among sensor observations in wireless sensor networks. Based on the verified relation between SVM and KNN, this new technique improved the SVM algorithm for classification by taking advantage of the KNN algorithm according to the distribution of data samples in a feature space and gives higher prediction accuracy than the SVM alone. At the same time, the rate of accuracy is also increased. Classification techniques can be used to non-stationary or dynamic data. Predicting fault is more efficient to prevent errors than detecting while fault has been occured. This can be very useful to prevent the occurrence of faults.

REFERENCES

[1] Salah Zidi, Tarek Moulahi, and Bechir Alaya, "Fault detection in Wireless Sensor Networks through SVM classifier,". Journal of Latex Class Files, Vol. 6, No. 1, June 2017.

[2] Panda, Rama Ranjan, Bhabani Sankar Gouda, and Trilochan Panigrahi, "Efficient fault node detection algorithm for wireless sensor networks," High Performance Computing and Applications (ICHPCA), 2014 International Conference on. IEEE, 2014.

[3] Feng, Zhen, Jing Qi Fu, and Yang Wang, "Weighted distributed fault detection for wireless sensor networks Based on the distance," Control Conference (CCC), 2014 33rd Chinese. IEEE, 2014.

[4] Warriach, Ehsan Ullah, and Kenji Tei, "Fault detection in wireless sensor networks: A machine learning approach, "Computational Science and Engineering (CSE), 2013 IEEE 16th International Conference on. IEEE, 2013.

[5] Yang, Chi, et al. "A time efficient approach for detecting errors in big sensor data on cloud," IEEE Transactions on Parallel and Distributed Systems 26.2 (2015): 329-339.

[6] Panda, Meenakshi, and Pabitra Mohan Khilar, "Energy efficient distributed fault identification algorithm in wireless sensor networks," Journal of Computer Networks and Communications 2014 (2014).

[7] Duarte Raposo, Andre Rodrigues, Jorge Sa Silva and Fernando Boavida, "A Taxonomy of Faults for Wireless Sensor Networks," Springer2017

[8] Obst, Oliver, "Distributed fault detection using a recurrent neural network," Proceedings of the 2009 International Conference on Information Processing in Sensor Networks. IEEE Computer Society, 2009

[9] Rong Li, Hua-Ning Wang, Han He, Yan –Mei Cuiand Zhan-Le Du, "Support Vector Machine combined with K-Nearest Neighbors for Solar Flare Forecasting," Chin. J. Astrophys. Vol. 7 (2007), No.3, 441-447

[10] https://en.wikipedia.org/wiki/Support_Vector_Machine

[11] https://en.wikipedia.org/wiki/Supervised_learning