International Journal of Applied and Advanced Multidisciplinary Research (IJAAMR)
Vol. 2, No. 1, 2024: 67-78

Fault Detection in Wireless Sensor Network Based on Deep Learning Algorithms

Pragati Mahale[1]*, Sejal Khopade[2],
AISSMS Institute of Information Technology
Corresponding Author: Pragati Mahale pragati.mahale@aissmsioit.org

ARTICLE INFO

Keywords: Wireless Sensor Network, Fault Detection, Convolution Neural Network, Convex Hull, Naive-Bayes, Energy Efficiency.

Received : 20, November
Revised : 22, December
Accepted: 24, January

ABSTRACT

This study discusses fully distributed fault detection via a wireless sensor network. Initially, we suggested using the Convex hull approach to determine a range of extreme points including nearby nodes. As the number of nodes rises, the message's duration is constrained. Secondly, in order to enhance convergence performance and identify node errors, we suggested using a convolution neural network (CNN) and a Naïve Bayes classifier. Lastly, we use real-world datasets to examine CNN, convex hull, and Naïve bayes algorithms to find and classify the defects. Based on performance measures, the results of simulations and experiments demonstrate that the CNN algorithm has better-identified defects than the convex hull technique while maintaining feasibility and economy.

DOI: https://doi.org/10.59890/ijaamr.v2i1.664
https://journal.multitechpublisher.com/index.php/ijaamr

INTRODUCTION

Recent developments in embedded computing and wireless transmission have made wireless sensor networks capable of delivering a wide range of applications. Numerous monitoring and control applications, including industrial sensing, traffic checking, and environmental surveillance, have been made possible by the widespread use of wireless sensor networks. Massive numbers of tiny, low-power wireless devices for traffic, industrial sensing, and environmental monitoring are part of WSNs. WSNs are made up of massive numbers of tiny, low-power wireless devices that are routinely transmitted to far-off, ill-organized locations. Numerous mobile and inevitable applications are constantly collecting and processing data from the real world and providing very detailed information on the identified state or events. In particular, sparsity of data, global optimization, and extensive applicability characterize SVM as a classification technique. Based on the cardinality of the preparation set, a quadratic dimensionality optimization problem must be solved in order to prepare an SVM. Support vectors are a subset of the preparation set that convey the next discriminating guidelineDue to strict energy constraints, limited data transfer capacity, and other demands on the communication capabilities of wireless sensor systems, distributed SVM preparation was examined in recent works. A parallel structure of centralized SVM is one approach. Partial SVMs are created utilizing small chunks of training data and merged in a combination focus when training data collection is massive. Large amounts of data can be handled by this technology, but it can only be used if a central processor is available to connect the partial, incomplete support vectors. The concentrated SVM does not always ensure the arbitrary division of the data set. However, there are fully suitable approaches for the entire SVM that make use of communicated improvement techniques. SVM can be solved using an existing convex optimization technique because it is a quadratic advancement problem. In order to get the multiplier exchange direction strategy, a distributed SVM was introduced.

This approach, which is based on neighborhood message trading, has been connected to centralized SVM. The gradient-based iteration may result in significant intercommunication costs because it is intended to keep the nodes connected until they combine. Furthermore, in a nonlinear scenario, the duration of the transferred message could end up being extraordinarily long. Applications for wireless sensor networks are inappropriate because of these issues. Instead of using the gradient approach, another distributed SVM class makes use of distributed support vectors that are gathered from nearby information sources. When the marked classes are linearly divisible, intermingling is ensured by these distributed SVM techniques based on gossips.

These approaches can be rough, though not assured, intermingling with the SVM arrangement when they are not directly identifiable. This article uses the notion of gossip-based gradual SVM with a geometric illustration. The nearest geometric point computations and the idea of the convex hull are essential to the geometric understanding of SVM. Not at all like the gossip-based incremental support vectors. At that point, we identify the flaw in the system we suggested and examined using techniques like CNN, convex hull, and Naïve Bayes. Comparing the Convolution Neural Network to the other techniques, it provides superior fault detection. The format of this document is as follows. The proposed framework model is highlighted in Section 2, the experiment results are explained in Section 3, and the article is concluded in Chapter 4.

LITERATURE REVIEW

Fault detection in wireless sensor networks (WSNs) is a critical aspect of ensuring the reliability and performance of these networks, which plays a crucial role in diverse applications. The integration of deep learning algorithms into fault detection mechanisms has gained significant attention due to their ability to handle complex data patterns. This literature review aims to provide an overview of the progress made in using deep learning techniques for fault detection in WSNs. One paragraph may focus on the methodologies and algorithms employed. Researchers have explored various deep learning algorithms, including Convolutional Neural Networks (CNNs) for image data from sensor nodes, Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks for sequential sensor data, and Autoencoders for anomaly detection. These algorithms have shown promise in detecting different types of faults, including sensor node failures, data anomalies, and communication disruptions, by learning patterns and anomalies from the sensor data.

Another paragraph may emphasize the challenges and future directions in this field. While deep learning shows great potential for enhancing fault detection in WSNs, it also faces challenges such as the resource constraints of sensor nodes and the need for energy-efficient models. Future research should focus on developing lightweight deep learning models and addressing real-time detection, potentially incorporating edge computing into WSNs. Moreover, the availability of larger and more diverse datasets for training deep learning models is essential for improving the robustness of fault detection in WSNs.

In summary, the integration of deep learning algorithms into fault detection mechanisms in wireless sensor networks is a growing field with promising results. As technology advances, overcoming challenges related to computational resources, energy efficiency, and the development of real-time solutions will be crucial to the continued growth and effectiveness of deep learning-based fault detection in WSNs.

Causes:

1. Growing Importance of WSNs: Wireless sensor networks have gained increasing significance in various domains, including environmental monitoring, industrial automation, smart cities, and

healthcare. The reliability and efficiency of these networks are vital for collecting and transmitting data accurately. Failures or anomalies in these networks can lead to significant consequences, such as data loss, erroneous decision-making, or system malfunctions.

2. Complexity and Volume of Data: WSNs generate a vast amount of data from numerous sensor nodes. Analyzing this data manually or using traditional methods can be time-consuming and impractical. Deep learning algorithms, with their capacity to automatically learn and detect complex patterns, provide an efficient means to process and detect faults in these data streams.

3. Advancements in Deep Learning: The rapid advances in deep learning techniques, facilitated by increased computing power and large datasets, have made them more accessible and effective. Deep learning algorithms, such as neural networks and convolutional networks, have shown exceptional capabilities in recognizing patterns and anomalies, making them an attractive solution for fault detection.

4. Efficiency and Real-Time Detection: Deep learning-based fault detection in WSNs can offer real-time or near-real-time capabilities, crucial for applications that require immediate responses to anomalies. Traditional fault detection methods may not always provide timely alerts or may be limited in terms of accuracy.

5. Research and Innovation: The scientific community's continuous drive to advance technology and improve the performance of WSNs has prompted researchers and engineers to explore innovative methods such as deep learning to address challenges related to fault detection. This project serves as a response to the need for research and innovation in this domain.

6. Resource and Energy Efficiency: Addressing the challenges associated with the computational and energy constraints of sensor nodes is another motivation. Developing lightweight deep learning models and optimization techniques can help minimize the resource demands, making deep learning-based fault detection more practical for resource-constrained environments.

Effects:

1. Improved Reliability and Network Performance: One of the primary effects of such a project is the enhancement of WSN reliability. By leveraging deep learning algorithms, the project can lead to more accurate and timely fault detection. This, in turn, can help in maintaining a higher level of network uptime, reducing the

risk of data loss or erroneous information, and ultimately improving the overall performance and dependability of WSNs in applications such as environmental monitoring, industrial automation, and healthcare.

2. Efficient Resource Management: Deep learning algorithms can be optimized for resource-constrained environments, enabling efficient resource management within the WSN. This includes the development of lightweight models and energy-efficient techniques for fault detection. The result is a more sustainable and cost-effective deployment of sensor networks, with longer node lifetimes and reduced operational costs.

3. Reduced Downtime and Maintenance Costs: The timely detection of faults through deep learning algorithms can help reduce downtime and minimize the need for costly manual inspections and maintenance. Early detection of sensor node failures, communication disruptions, or data anomalies allows for quicker response and remediation actions, thereby reducing operational disruptions and maintenance expenses.

4. Enhanced Data Quality and Decision-Making: Reliable fault detection can contribute to higher data quality by reducing the impact of erroneous or missing data points. This improved data quality can lead to more accurate decision-making processes in applications that rely on WSN data. For example, in environmental monitoring, accurate data is critical for assessing and responding to environmental changes effectively.

5. Real-Time and Proactive Responses: The integration of deep learning in fault detection enables real-time or near-real-time responses to anomalies. This can be crucial in applications where immediate actions are necessary, such as in critical infrastructure monitoring, where early detection of faults can prevent accidents or disasters.

6. Advancements in Research and Development: The project can contribute to the advancement of knowledge and research in the field of WSNs and deep learning. New methodologies, techniques, and best practices developed as part of the project can serve as a foundation for further research and innovation, benefiting the broader scientific community.

7. Competitive Advantage: Organizations and industries that adopt deep learning-based fault detection in WSNs can gain a competitive advantage by having more reliable and efficient networks. This can lead to improved customer satisfaction, reduced operational risks, and enhanced market positioning.

METHODOLOGY

In this suggested study, we use many methods, including the CNN algorithm, Naïve Bayes, and convex hull algorithm. The convex hull algorithm only exchanges messages with nodes that are next to one other. Even in the worst-case situation, one-hop communication allows for fundamental control over the amount of information exchanged, which in turn determines the topology of the network link. In order to enhance the power and convergence performance of a wireless sensor network, the CNN algorithms and the naïve Bayes classifier detect system flaws. When contrasted with other methodologies, the CNN technique performs better. In general, a physical layer device may experience a malfunction, as seen beneath Figure 1. A deficiency occurs when one of the system's trademark properties or parameters deviates from standard operation at any point. Deficiencies may arise, for instance, when a network node malfunctions because of poor battery life, low correspondence impedance, physical harm, or environmental hindrance.

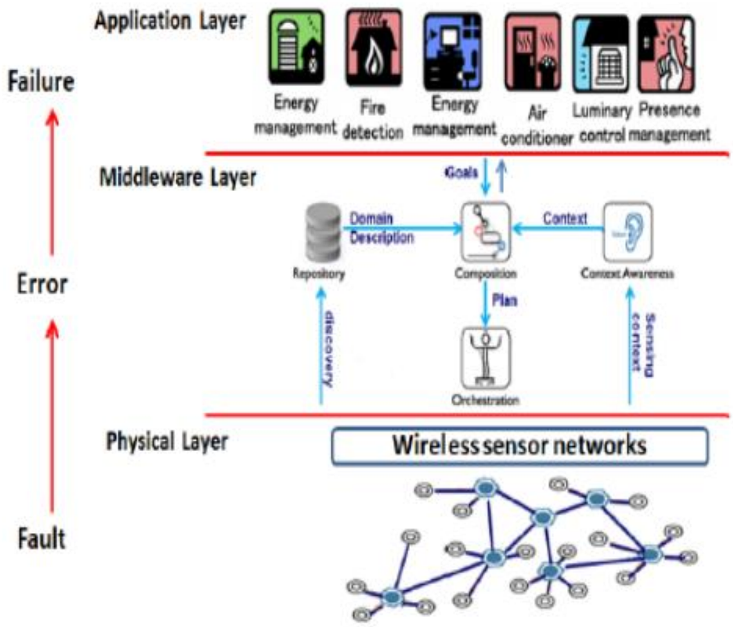

Figure 1. Architecture of Sensor Network Application

An error occurs when the middleware layer fails to correctly identify a condition or event in the allocated area because of a weakness. An error is an outcome for which there is a deficiency. This takes into account an absurdly logical problem with the devices because they are erroneously identified as living things despite this. As a result, middleware errors or physical layer issues can cause services to fail at the application level. Therefore, user annoyance may arise from malfunctions in such dynamic and interactive systems. In a smart home, for example, a user's identity is determined by a variety of sensor.

The system is configured based on the user's choices when they return home. An unexpected system configuration may arise from incorrect user identification. Any device's flaws can result in improper control and use of the equipment in the real world. For instance, a room or physical area may overcool or overheat if temperature sensor values are not correctly sensed.

1. Convex Hull Algorithm

Convex hull for a set Scan likewise be characterized as the arrangement of focuses that can be communicated as arched mixes of the points in that set S. Convex hull have their image processing applications, design acknowledgement, etc. medicinal recreations. A convex hull is diverse for various items since it depends on the feature point of each object. The convex hull can be characterized for the object of any sort with any number of measurements. The convex hull's complexity and extreme points are focused on the feature space's dimensionality.

2. Naïve Bayes

The algorithm of the SVM naïve Bayers classifier is described as follows:

- Algorithms

  Read information, create a cv partition purpose that defines folds, create a guidance set, create an analysis set, compute the class likelihood, normal training set distribution percentage
- Parameters

  Test set probability, kernel supply, test set estimate likelihood, re-structure, get an anticipated test set output, compare expected output with the actual set

Naive Bayes requires better fault detection as it is necessary to calculate the probabilistic models from a continuous distribution. This means the training phase is pretty fast. The computing time, the best algorithm for classification error, appears to be the Naïve Bayes algorithm. The Naïve Bayes algorithm is better than the convex hull algorithm.

3. Convolution Neural Network

Algorithm

- Step 1: Convolution Operation

  The first building block in our plan of attack is convolution operation

- Step 1b: ReLU Layer
  The second part of this step will involve the Rectified Linear Unit or ReLU

- Step 2: Pooling
  In this part, we'll cover pooling and will get to understand exactly how it generally works.

- Step 3: Flattening
  This will be a brief breakdown of the flattening process and how we move from pooled to flattened layers when working with Convolutional Neural Networks.

- Step 4: Full Connection
  In this part, everything that we covered throughout the section will be merged. By learning this, you'll get to envision a fuller picture of how Convolutional Neural Networks

RESEARCH RESULT AND DISCUSSION

The experimental outcomes of the proposed research are defined as follows, which will be implemented in the MATLAB software platform.

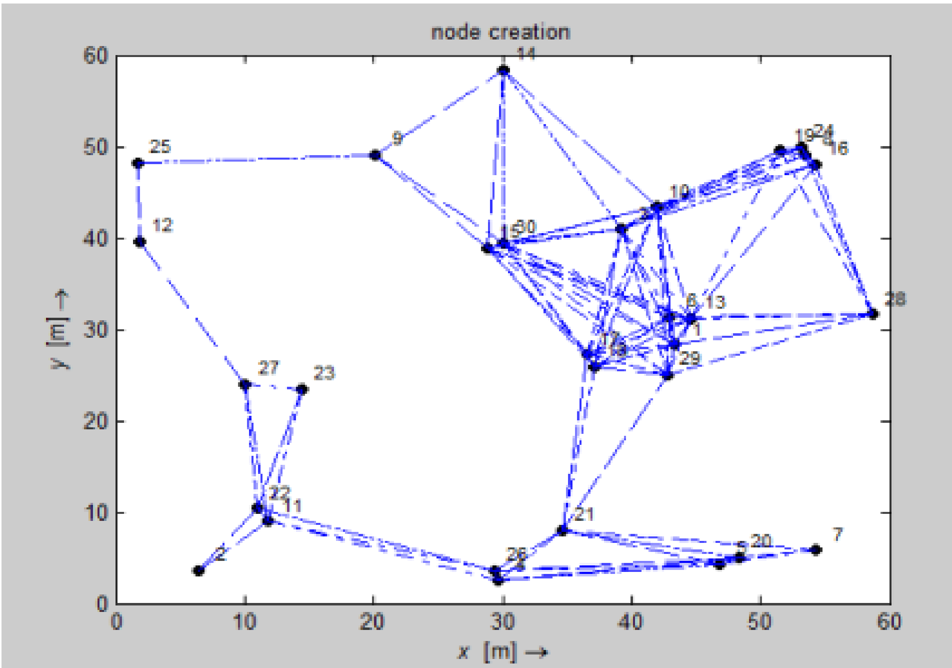

Figure 2

That's picture demonstrates a node creation representing the information in a single data structure. These nodes may contain exchanged information, condition, or data from another node.

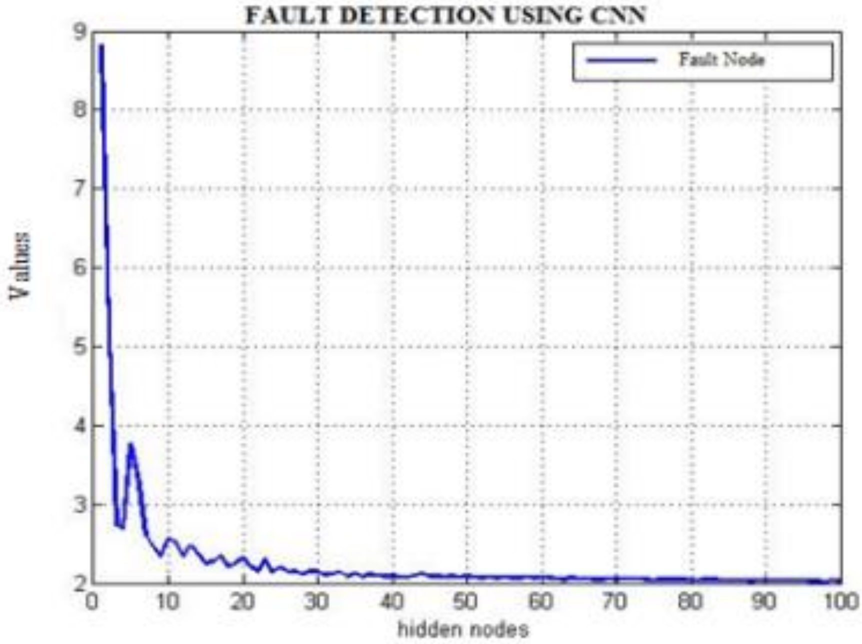

Figure 3

That picture shows that the performance of Node faults detection by using a convolution neural network. The CNN technique detects fault easily from all the hidden nodes.

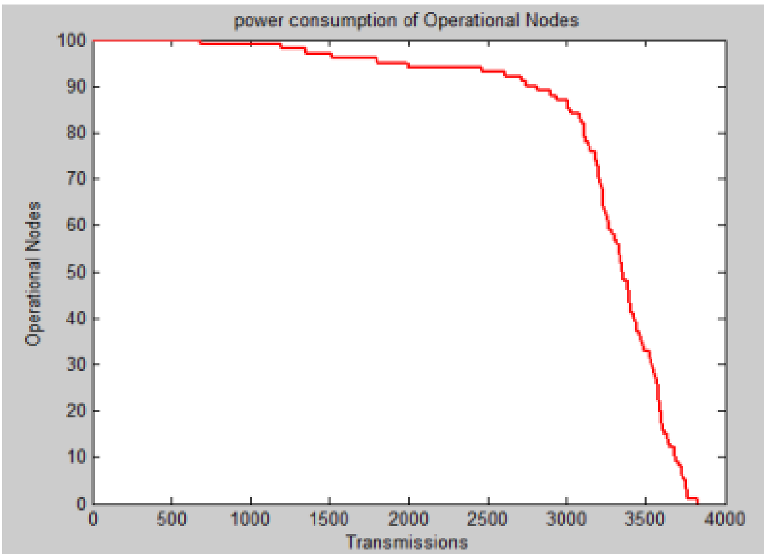

Figure 4

That picture shows a region where both the average length of the path and the estimated power consumption is small. In this region, with only a negligible rise in energy consumption, we can significantly enhance our suggested CNN algorithm's efficiency in terms of convergence velocity.

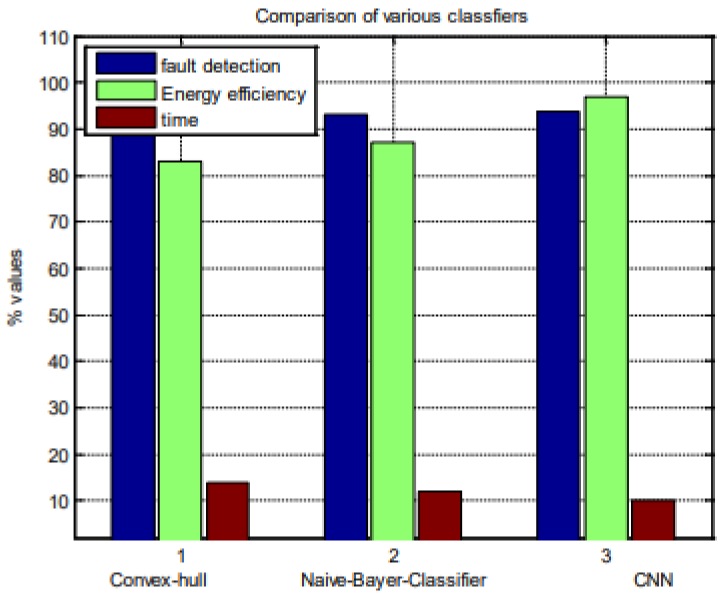

Figure 5

That picture shows the performance analysis of convex hull, Naïve Bayes and CNN techniques. The deep learning method provides better results than the Naïve-Bayes algorithm and the convex hull. This proposed work easily detects the faults and gives better energy efficiency, and Here, the time consumption is very low compared to both techniques.

CONCLUSION

This study suggested the use of several techniques, including convolution neural networks, convex hulls, and naïve bayes, for fault identification in wireless sensor networks. Our results demonstrate that CNN outperforms other methods in defect detection.

ADVANCED RESEARCH

In writing this article the researcher realizes that there are still many shortcomings in terms of language, writing, and form of presentation considering the limited knowledge and abilities of the researchers themselves. Therefore, for the perfection of the article, the researcher expects constructive criticism and suggestions from various parties.

REFERENCES

B.C. Lau *et al.* Probabilistic fault detector for wireless sensor network . Expert Syst. Appl. (2014)

C. Leys ***et al.*** Detecting outliersdo not use standard deviation around the mean, use absolute deviation around the median. J. Exp. Soc. Psychol. **(2013)**

M. Panda *et al.* Distributed self fault diagnosis algorithm for large scale wireless sensor networks using modified three sigma edit test. Ad Hoc Netw. Part A (2015)

Theodoridis, Sergios, and Michael Mavroforakis. "Reduced convex hulls: a geometric approach to support vector machines [lecture notes]." IEEE Signal Processing Magazine 24, no. 3 (2007): 119-122.

Vapnik, Vladimir Naumovich, and V. Vapnik. "Statistical Learning Theory, vol. 1. Hoboken." (1998).

Vimalachandran, Pasupathy, Hong Liu, Yongzheng Lin, Ke Ji, Hua Wang, and Yanchun Zhang. "Improving accessibility of the Australian My Health Records while preserving privacy and security of the system." Health Information Science and Systems 8, no. 1 (2020): 1-9.

Warriach, Ehsan Ullah, Kenji Tei, Tuan Anh Nguyen, and Marco Aiello. "Fault detection in wireless sensor networks: a hybrid approach." In Proceedings of the 11th international conference on Information Processing in Sensor Networks, pp. 87-88. ACM, 2012.

Yin, Jiao, MingJian Tang, Jinli Cao, Hua Wang, Mingshan You, and Yongzheng Lin. "Adaptive Online Learning for Vulnerability Exploitation Time Prediction." In International Conference on Web Information Systems Engineering, pp. 252-266. Springer, Cham, 2020.

Zhu, Kaihua, Hao Wang, Hongjie Bai, Jian Li, ZhihuanQiu, Hang Cui, and Edward Y. Chang. "Parallelizing support vector machines on distributed computers." In Advances in Neural Information Processing Systems, pp. 257-264. 2008.