International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 07 | July - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

AN OVERVIEW OF MACHINE LEARNING ALGORITHMS FOR WIRELESS SENSOR NETWORKS

Pritam Nanda

Dept of Computer Science and Engineering

E-mail – pritamn2022@gift.edu.in

Sasmita Tripathy

Dept of Computer Science and Engineering

E-mail – sasmita.tripathy@gift.edu.in

ABSTRACT

Wireless sensor networks (WSNs) are particularly desirable for real-time applications because of their small size, low cost, and simplicity of installation. Nevertheless, WSNs may need to be modified or redesigned due to a variety of internal or external circumstances, which is difficult for conventional, explicitly planned WSN systems to manage. Machine learning (ML) approaches can be used to solve this problem. ML makes it possible for networks to learn from their experiences and adapt without requiring reprogramming or human intervention.A prior investigation [1] examined machine learning methods for WSNs between 2002 and 2013. We review ML-based algorithms for WSNs from 2014 to March 2018 in this revised study, stressing their advantages, drawbacks, and effects on network lifetime. We also discuss machine learning techniques for energy harvesting, congestion control, mobile sink scheduling, and synchronization. The survey discusses why certain ML approaches are selected for particular WSN difficulties and offers a statistical analysis of the data obtained. We also talk about some outstanding issues in the sector.

Keywords: Wireless sensor networks, Machine learning, Energy efficiency, Network lifetime, Data aggregation

1. INTRODUCTION

The wireless sensor network (WSN), due to its small size, low cost, and ease of deployment, is one of the most promising technologies for certain real-time applications [2]. Monitoring a field of interest, gathering specific data, and transmitting it to the base station for post-data analysis are the responsibilities of WSN [3, 4]. A significant number of sensor nodes are used in some WSN apps. Therefore, scalable and effective methods are needed to manage such a vast number of nodes. Additionally, the WSNs may alter dynamically as a result of outside factors or as planned by the system designers. Consequently, it might have an impact on cross-layer design, localization, latency, network routing methods, QoS, connection quality, fault detection, and coverage [5, 6].Due to its extremely dynamic nature, the network may need to be redesigned with depreciating dispensable components. However, standard WSN techniques need explicit programming, which prevents the network from functioning effectively in a dynamic context. Machine Learning (ML) is the process that occurs without explicit programming and automatically learns or improves from research or experience [7-9]. Our computer operations were becoming more dependable, economical, and efficient thanks to ML. By automatically, swiftly, and reliably processing increasingly more complicated data, machine learning creates models. It is primarily divided into four categories: semi-supervised learning,

reinforcement learning, unsupervised learning, and supervised learning.

The power of machine learning (ML) rests in its capacity to deliver broad fixes via an architecture that can be trained to perform better. Its multidisciplinary nature makes it essential to many areas, including as computers, medicine, and engineering. Numerous problems with WSNs have been addressed by recent ML advancements [1]. Using ML reduces the need for human intervention or reprogramming while also enhancing WSN performance. It is difficult to access the massive amounts of data that sensors generate and to extract the relevant information from the data without machine learning (ML). Additionally, it is used to integrate machine-to-machine (M2M), cyber-physical systems (CPS), and the Internet of Things (IoT) [1]. The following are a few uses of ML in WSNs:

- Using machine learning techniques, it is simple to determine the ideal number of sensor nodes to cover the desired region.
- Energy harvesting gives WSNs operating in hostile environments a sustained, self-sufficient maintenance source.
- The ability of WSNs to predict how much energy will be gathered during a specific time window is enhanced by ML algorithms.
- A variety of internal and external events may cause sensor nodes to move. With the aid of ML algorithms, accurate localization is quick and easy.
- ML is used to separate malfunctioning sensor nodes from healthy sensor nodes and increase network efficiency.
- An important factor in extending the network lifetime is data routing. Dynamic routing algorithms are necessary due to the dynamic nature of sensor networks.

2. MACHINELEARNING TECHNIQUES

In order to better comprehend the following parts, we will examine a variety of machine learning approaches and their learning processes in this section. Additionally, we provided a succinct overview of evolutionary computing methods for WSNs. ML approaches have been divided into four categories: semi-supervised learning, reinforcement learning, unsupervised learning, and supervised learning, depending on the learning styles. ML method taxonomy is displayed in Fig. 1.

2.1 SUPERVISED LEARNING

In machine learning, supervised learning is one of the most crucial methods for handling data. When a system is trained using supervised learning, we give it a set of input and outputs (datasets with labels), and it determines the relationship between them. Once the training process is complete, we may identify a function (f: x → y) from an input x that best estimates an output y. One of the main tasks of supervised learning algorithms is to create a model that illustrates the connections and interdependencies between input data and the anticipated outputs. Many problems in WSNs are resolved by supervised learning, including localization [10–25], coverage issues [26–31], anomaly and fault detection [32–45], routing [46–53], MAC [54], data aggregation [55–67], synchronization [68–71], congestion control [72–74], target tracking [75–78], and event detection [79–81].

Regression and classification are the two categories for supervised learning. There are four types of classification algorithms: instance-based (k-NN), statistical learning (Bayesian and SVM), perceptron-based (ANN and deep learning), and logic-based (decision tree and random forest).

2.1.1 REGRESSION

As a supervised learning technique, regression makes predictions about a given collection of characteristics (X) and a corresponding value (Y). The regression model uses continuous or quantitative variables. Regression is a relatively basic machine learning technique that produces precise predictions with few mistakes. Equation (1) displays the linear regression mathematical notation [84].

2.1.2 DECISION TREE

A type of supervised machine learning technique called decision trees (DT) uses a collection of if-then rules to classify data and improve readability. Decision nodes (choice between options) and leaf nodes (ultimate outcomes) are the two types of nodes

found in a decision tree [87]. By building a training model using decision rules deduced from training data, decision trees are used to forecast a class or target. Figure 3 displays an example of a decision tree's graphical form. Transparency, a reduction in decision-making uncertainty, and the ability to conduct a thorough study are the decision tree's main benefits.

2.1.3 RANDOM FOREST

A supervised machine learning approach called the random forest (RF) algorithm uses a collection of trees and assigns a classification to each tree in the forest. The random forest classifier is created first, and then the outcomes are predicted using the RF method [89]. For larger datasets and diverse data, RF performs well. The missing values are correctly predicted by this method. A huge number of decision trees will be produced by isolating variables at each node of the tree and randomly picking a subset of training samples. Because of the high quality of training data and too strong decision trees, the RF classifier's sensitivity level is lower when compared to other streamline machine learning classifiers. The curse of dimensionality and highly variable data provide substantial hurdles for current classification methods.

2.1.4 ARTIFICIAL NEURAL NETWORK (ANN)

An artificial neural network (ANN) is a supervised machine learning approach for data classification that is based on the model of a human neuron [91,92]. An enormous number of neurons, or processing units, coupled to an ANN process data and generate precise outputs. An artificial neural network (ANN) usually functions on layers, each of which is connected to nodes, each of which has an active role. An ANN's fundamental layer structure is seen in Fig. 4. Three layers make up each ANN: the input layer, the output layer, and one or more hidden layers.

2.1.5 DEEP LEARNING

Deep learning is a subclass of ANN and is a supervised machine learning technique used for categorization. The data learning representation techniques with multi-layer representations (between the input and output layers) are known as deep learning approaches. In order to arrive at the optimal solution, it is composed of basic nonlinear modules that translate the representation from a lower layer to a higher layer [93]. It draws inspiration from the information processing and communication patterns found in human nervous systems [94]. The ability to extract high-level characteristics from data, work with or without labels, and be taught to accomplish numerous goals are the main advantages of deep learning. Many fields, including bioinformatics, social network analysis, business intelligence, speech recognition, handwriting identification, and medical image processing, can benefit from it.

2.1.6 SUPPORT VECTOR MACHIE (SVM)

A supervised machine learning classifier called the support vector machine (SVM) selects the best hyperplane to use for data classification. SVM uses a hyperplane to coordinate individual observations and delivers the best classification [99]. Once a boundary is defined and a collection of points aids in boundary identification, the majority of the training data becomes superfluous. Support vectors are the points that are utilized to determine the border. The optimal classification from a given set of data is produced using SVM. As a result, the quantity of features in the training set has no bearing on the model complexity of an SVM. Because of this, SVMs work effectively in learning problems where the number of features is high in comparison to the number of training examples. Using SVM with WSNs.

2.1.7 BAYESIAN

A supervised machine learning method called Bayesian is based on statistical learning techniques [102]. Using a variety of statistical techniques, such as the Chi-square test, a Bayesian learning algorithm determines the links between the datasets by learning conditional independence. The probability ($\theta|X1$, $X2$, $X3$ …., Xn) to be maximized is returned as a label θ for a collection of inputs $X1$, $X2$, $X3$ …., Xn}. Different probability functions for various class node variables are allowed by Bayesian learning. Recently, a number of WSN issues have been resolved using Bayesian learning techniques in an effort to increase

network efficiency. The problems include target tracking [75–78,103], event detection [104], routing [51–53], localization [21–25], coverage [31], anomaly & fault detection [37,38,41–43], synchronization [71], and mobile sink path selection [105].N

2.1.7 K-NEAREST NEIGHBOR(K-NN)

The simplest lazy instance-based learning technique for regression and classification is K-Nearest Neighbor (k-NN). The input from the feature space that is considered closest is the training set. K-NN typically uses the distance between predefined training samples and the test sample to determine classification. Numerous distance functions, including the Euclidean, Hamming, Canberra, Manhattan, Minkowski, and Chebychev distance functions, are used in the K-NN approach. The size of the input dataset and optimal performance at the same scale of the data determine the k-NN algorithm's complexity. This method lowers the dimensionality and extracts any missing values from the feature space [106–108].

2.2 UNSUPERVISED LEARNING

Without a result (unlabelled) linked to the inputs, unsupervised learning involves even the model attempting to infer associations from the data. Unsupervised learning techniques include grouping a collection of related patterns into clusters, reducing the dimensionality of the data, and identifying anomalies. Unsupervised learning has made significant contributions to WSNs by addressing a number of problems, including data aggregation [116–125], anomaly detection [111], routing [112–115], and connection problems [110]. Unsupervised learning is further divided into dimensionality reduction (PCA, ICA, and SVD) and clustering (k-means, hierarchical, and fuzzy-c-means).

2.2.1 K-MEANS CLUSTERING

From a given dataset, the k-means algorithm constructs a particular number of clusters with ease [126]. The first k random sites are taken into consideration, and all the remaining points are connected to the closest centers. A fresh centroid from every cluster is recalculated once the clusters are created by encircling every point in the dataset. Until there are no more changes to the centroid of any cluster, repeat the method with the cluster's centroid changing with each iteration. The k-means technique has an O(n∗k∗i∗d) time complexity, where n denotes the number of points, k, the number of centroids, i, the number of iterations, and d, the number of attributes.

2.2.2 HIERARCHICAL CLUSTERING

Similar items are grouped into clusters using the hierarchical clustering approach, which determines the top-down or bottom-up order of the clusters. Divisive clustering, also known as top-down hierarchical clustering, involves splitting a big single division iteratively until there is one cluster for each observation. Agglomerative clustering, also known as bottom-up hierarchical clustering, allocates each observation to its cluster based on density functions [129, 130]. The hierarchical clustering technique is simple to use and does not require any prior knowledge of the number of groups. This clustering algorithm has an O(n3) worst-case time complexity and an O(n2) worst-case space complexity.

3. MACHINE LEARNING ALGORITHMS FOR WIRELESS SENSOR NETWORKS (WSNS)

Wireless sensor networks (WSNs) are a prominent technology due to their compact size, cost-effectiveness, and ease of deployment, making them ideal for real-time applications. However, WSNs often encounter dynamic changes due to external or internal factors, necessitating network redesigns. Traditional WSNs are explicitly programmed, which limits their ability to respond dynamically to these changes. Machine learning (ML) techniques can address this limitation by enabling the network to learn and adapt from experiences without human intervention.

Supervised Learning

Supervised learning involves training systems with labeled data to establish input-output relationships. This technique is used to address various challenges in WSNs, such as localization, coverage, anomaly detection, and more. Common algorithms include:

- **Regression:** Predicts values based on features, useful for localization, connectivity, and energy harvesting.
- **Decision Trees:** Use if-then rules for classification, aiding in connectivity and data aggregation.
- Random Forest: An ensemble of decision trees, improving coverage and MAC protocols.
- **Artificial Neural Networks (ANN):** Mimic human neuron networks for complex data classification, applied in localization, fault detection, and routing.
- **Support Vector Machine (SVM):** Optimal hyperplane-based classification, used in localization and congestion control.

Unsupervised Learning

Unsupervised learning classifies patterns and reduces data dimensionality without labeled outputs. It addresses issues like connectivity and data aggregation. Key algorithms include:

- **k-Means Clustering**: Forms clusters based on distance metrics, aiding in optimal cluster head selection.
- **Hierarchical Clustering:** Groups data with a hierarchical structure, useful for data aggregation and synchronization.
- **Fuzzy c-Means Clustering**: Soft clustering assigns data to multiple clusters, improving localization and connectivity.
- **Principal Component Analysis (PCA):** Extracts significant features, reducing dimensionality for fault detection and target tracking.

Semi-Supervised Learning

Combines labeled and unlabeled data to address real-world applications like localization and fault detection.

Reinforcement Learning

Interacts with environments to optimize actions, enhancing network performance through adaptive learning.

Evolutionary Computation

Nature-inspired algorithms iteratively optimize solutions, applied in localization, coverage, routing, and target tracking.

Applications of ML in WSNs

- **Localization**: ML techniques improve accuracy and reduce error rates in identifying sensor node positions.
- **Coverage & Connectivity:** Ensures efficient monitoring and connectivity, employing ML algorithms for optimal sensor node deployment and dynamic routing.
- **Anomaly Detection:** Detects deviations in sensor data, safeguarding against attacks using ML approaches like k-Means, SVM, and Bayesian networks.
- **Fault Detection**: Identifies faults in sensor nodes and networks, improving accuracy and reducing false positives through ML techniques.
- **Routing:** Optimizes data transmission paths, reducing energy consumption and enhancing network lifetime with ML-based routing protocols.

ML techniques have demonstrated significant potential in enhancing the performance, efficiency, and reliability of WSNs by enabling adaptive responses to dynamic environments and optimizing various network operations.

4. Statistical Analysis and Limitations

Statistical Analysis

Statistical analysis in the context of wireless sensor networks (WSNs) involves examining data collected from various sources to derive meaningful insights and support decision-making. This process includes

the use of various statistical tools and methods to understand patterns, trends, and relationships within the data. In WSNs, statistical analysis can help to:

Evaluate Network Performance:

- **Throughput:** Measure the amount of data successfully delivered over the network.
- **Latency:** Determine the delay experienced in data transmission.
- **Packet Delivery Ratio:** Calculate the ratio of packets successfully delivered to the total number of packets sent.

Assess Energy Efficiency:

- Analyze the energy consumption patterns of sensor nodes.
- Identify energy-saving opportunities and strategies.

2. **Detect Anomalies:**

- Use statistical methods to identify outliers or unusual patterns in the data, which may indicate faults or security breaches.

Optimize Network Configuration:

- Utilize statistical models to predict optimal configurations for improved performance and longevity.

Support Predictive Maintenance:

- Apply statistical techniques to forecast potential failures or maintenance needs based on historical data.

Limitations

Despite the benefits of statistical analysis in WSNs, there are several limitations to consider:

Data Quality:

- **Incomplete Data:** Sensor nodes may fail, leading to gaps in the data collected.
- **Noise:** Environmental factors can introduce noise, complicating the analysis and interpretation of data.

Computational Constraints:

- **Resource Limitations:** Sensor nodes typically have limited processing power, memory, and energy, which restricts the complexity of statistical computations that can be performed locally.
- **Scalability:** Large-scale WSNs generate vast amounts of data, posing challenges for real-time processing and analysis.

Dynamic Environments:

- **Changing Conditions:** WSNs often operate in environments where conditions can change rapidly, affecting the reliability and relevance of statistical models.
- **Mobility:** Movement of sensor nodes or monitored objects can introduce variability that is difficult to model statistically.

Model Assumptions:

- **Simplified Assumptions:** Statistical models often rely on assumptions (e.g., normality, independence) that may not hold true in all scenarios, leading to inaccuracies.
- **Generalization:** Models developed for specific conditions may not generalize well to different environments or applications.

Implementation Challenges:

- **Synchronization Issues:** Ensuring synchronized data collection across distributed sensor nodes can be difficult, impacting the accuracy of time-dependent analyses.
- **Communication Overhead:** Transmitting large amounts of data for centralized processing can lead to increased communication overhead and energy consumption.

In summary, while statistical analysis provides valuable insights and aids in the optimization of

WSNs, it is essential to be aware of its limitations. Addressing these challenges requires careful consideration of data quality, computational resources, and the dynamic nature of WSN environments.

5 OPEN ISSUES

Energy Efficiency:

- **Battery Life:** Sensor nodes have limited battery life, and energy-efficient protocols are needed to extend the network lifetime.
- **Energy Harvesting:** Developing reliable energy harvesting techniques to sustain WSNs in remote or hostile environments.

Scalability:

- **Network Expansion:** Ensuring that WSNs can scale to accommodate a large number of sensor nodes without degrading performance.
- **Data Management:** Efficiently managing and processing the vast amounts of data generated by large-scale WSNs.

Security:

- **Data Privacy:** Protecting sensitive data transmitted across the network from unauthorized access and breaches.
- **Attack Detection:** Developing robust methods to detect and mitigate various types of attacks, such as denial-of-service (DoS), spoofing, and eavesdropping.

Localization:

- **Accurate Positioning:** Improving the accuracy of localization techniques, especially in environments where GPS signals are weak or unavailable.
    - **Dynamic Environments:** Adapting localization methods to account for the mobility of nodes and changes in the environment.

Fault Tolerance:

- **Node Failures:** Enhancing the network's ability to detect and recover from node failures without significant impact on overall performance.
- **Data Reliability:** Ensuring the reliability and integrity of data in the presence of faulty or compromised nodes.

Quality of Service (QoS):

- **Latency:** Reducing data transmission delays to meet the requirements of real-time applications.
- **Bandwidth Management:** Efficiently managing limited bandwidth to prioritize critical data and avoid congestion.

Interoperability:

- **Standardization:** Developing standardized protocols and interfaces to ensure interoperability among different types of sensor nodes and networks.
- **Integration with IoT:** Facilitating seamless integration of WSNs with the broader Internet of Things (IoT) ecosystem.

Deployment and Maintenance:

- **Deployment Strategies:** Optimizing sensor node deployment strategies to ensure maximum coverage and connectivity.
- **Maintenance Costs:** Reducing the maintenance and operational costs associated with large-scale WSN deployments.

2. **Data Aggregation and Processing:**

- **Efficient Aggregation:** Designing algorithms for efficient data aggregation to reduce redundancy and communication overhead.
- **Real-time Processing:** Developing methods for real-time data processing and analysis at the edge to reduce latency and bandwidth usage.

Addressing these open issues is crucial for the advancement and widespread adoption of WSNs in

various applications, including environmental monitoring, healthcare, smart cities, and industrial automation. Continuous research and innovation are required to overcome these challenges and unlock the full potential of WSNs.

6 CONCLUSIONS

Wireless sensor networks (WSNs) have proven to be a transformative technology for a wide range of real-time applications due to their cost-effectiveness, compact size, and ease of deployment. However, their dynamic nature necessitates advanced approaches to manage and optimize their performance. Machine learning (ML) techniques offer promising solutions by enabling adaptive and intelligent responses to changing conditions without the need for human intervention.

This survey has reviewed various ML algorithms applied to WSNs, highlighting their advantages, limitations, and impact on network lifetime from 2014 to March 2018. Supervised, unsupervised, semi-supervised, and reinforcement learning techniques have been explored for addressing key issues such as localization, coverage, connectivity, anomaly detection, fault detection, and routing.

Despite significant progress, several open issues remain, including energy efficiency, scalability, security, accurate localization, fault tolerance, quality of service, interoperability, deployment strategies, maintenance, and efficient data aggregation and processing. Addressing these challenges will require ongoing research and innovation.

REFERENCE

1. M.A. Alsheikh, S. Lin, D. Niyato, H.-P. Tan, Machine learning in wireless sensornetworks: algorithms, strategies, and applications, IEEE Commun. Surv. Tutorials16 (4) (2014) 1996–2018.
2. P. Rawat, K.D. Singh, H. Chaouchi, J.M. Bonnin, Wireless sensor networks: a surveyon recent developments and potential synergies, J. Supercomput 68 (1) (2014)1–48.
3. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Comput. Networks 38 (4) (2002) 393–422.
4. J. Yick, B. Mukherjee, D. Ghosal, Wireless sensor network survey, Comput. Networks 52 (12) (2008) 2292–2330.
5. D.K. Sah, T. Amgoth, Parametric survey on cross-layer designs for wireless sensornetworks, Comput Sci. Rev. 27 (2018) 112–134.
6. J.B. Predd, S.B. Kulkarni, H.V. Poor, Distributed learning in wireless sensor networks, IEEE Signal Process. Mag. 23 (4) (2006) 56–69.
7. T.M. Mitchell, Machine Learning, first ed., McGraw-Hill, Inc., New York, NY, USA,1997.
8. T.O. Ayodele, Introduction to Machine Learning, first ed, InTech, 2010.
9. P. Langley, H.A. Simon, Applications of machine learning and rule induction, Commun. ACM 38 (11) (1995) 54–64.
10. S.S. Banihashemian, F. Adibnia, M.A. Sarram, A new range-free and storage-efficient localization algorithm using neural networks in wireless sensor networks,Wireless Personal Commun. 98 (1) (2018) 1547–1568.
11. A. El Assaf, S. Zaidi, S. Affes, N. Kandil, Robust ANNs-based WSN localization inthe presence of anisotropic signal attenuation, IEEE Wireless Commun. Lett. 5 (5) (2016) 504–507.
12. S.K. Gharghan, R. Nordin, M. Ismail, J.A. Ali, Accurate wireless sensor localization technique based on hybrid PSO-ANN algorithm for indoor and outdoor trackcycling, IEEE Sens. J. 16 (2) (2016) 529–541.
13. S. Phoemphon, C. So-In, D.T. Niyato, A hybrid model using fuzzy logic and anextreme learning machine with vector particle swarm optimization for wirelesssensor network localization, Appl. Soft Comput. 65 (2018) 101–120.
14. N. Baccar, R. Bouallegue, Interval type 2 fuzzy localization for wireless sensor networks, EURASIP J. Adv. Signal Process. 2016 (1) (2016) 1–13.
15. J. Kang, Y.J. Park, J. Lee, S.H. Wang, D.S. Eom, Novel leakage detection by ensemble CNN-SVM and graph-based localization in water distribution systems, IEEETrans. Ind. Electron. 65 (5) (2018) 4279–4289.