International Journal of Latest Trends in Engineering and Technology
Vol.(7)Issue(4), pp.234-239
DOI: http://dx.doi.org/10.21172/1.74.032
e-ISSN:2278-621X

A SURVEY ON WIRELESS SENSOR NETWORK (WSN) SECURITY USING AI METHODS

Harsimran Kaur[1] and Mani Sahore[2]

Abstract: Wireless sensor networks monitor dynamic environments that change rapidly over time. This dynamic behavior is either caused by external factors or initiated by the system designers themselves. To adapt to such conditions, sensor networks often adopt machine learning techniques to eliminate the need for unnecessary redesign. Machine learning also inspires many practical solutions that maximize resource utilization and prolong the lifespan of the network. In this paper, we provided a comparative guide to aid WSN designers in developing suitable machine learning solutions for their specific application challenges.

I. INTRODUCTION

A Wireless sensor network (WSN) is composed typically of multiple autonomous, tiny, low cost and low power sensor nodes. These nodes gather data about their environment and collaborate to forward sensed data to centralized backend units called base stations or sinks for further processing. The sensor nodes could be equipped with various types of sensors, such as thermal, acoustic, chemical, pressure, weather, and optical sensors. Because of this diversity, WSNs have tremendous potential for building powerful applications, each with its own individual characteristics and requirements. Developing efficient algorithms that are suitable for many different application scenarios is a challenging task. In particular, WSN designers have to address common issues related to data aggregation, data reliability, localization, node clustering, energy aware routing, events scheduling, fault detection and security. Machine learning (ML) was introduced in the late 1950's as a technique for artificial intelligence (AI) [1]. Over time, its focus evolved and shifted more to algorithms which are computationally viable and robust.

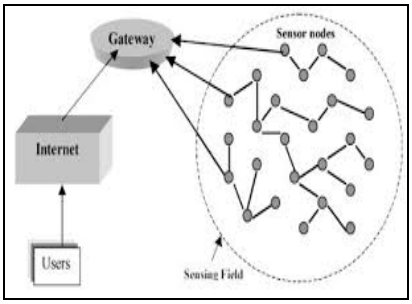

Figure 1: WSN Architecture

[1] *Chitkara university of Engineering and Technology Baddi(Himachal Pradesh)*
[2] *Chitkara university of Engineering and Technology Baddi(Himachal Pradesh)*

Machine learning is important in WSN applications for the following main reasons:

1) Sensor networks usually monitor dynamic environments that change rapidly over time. For example, a node's location may change due to soil erosion or sea turbulence. It is desirable to develop sensor networks that can adapt and operate efficiently in such environments [2, 3].
2) WSNs may be used for collecting new knowledge about unreachable, dangerous locations [4] (e.g., volcano eruption and waste water monitoring) in exploratory applications. Due to the unexpected behavior patterns that may arise in such scenarios, system designers may develop solutions that initially may not operate as expected. System designers would rather have robust machine learning algorithms that are able to calibrate itself to newly acquired knowledge.
3) WSNs are usually deployed in complicated environments where researchers cannot build accurate mathematical models to describe the system behavior. Meanwhile, some tasks in WSNs can be prescribed using simple mathematical models but may still need complex algorithms to solve them (e.g., the routing problem [5], [6]). Under similar circumstances, machine learning provides low-complexity estimates for the system model.

II. LITERATURE REVIEW

The nodes moves from one position with x-y coordinate to another position with different x-y coordinates. The mobility factor makes the node attack prone as it is easy to attack and disappear if they can move. Attacks are even smart and act differently in different situations. So, above table describes the traditional methods for solvation of problems in WSN [12- 20].

Table 1: Literature review

Author	Routing propotocl	method	Problem formulation	metrics
Kannan et.al	**secure AODV routing protocol**	**MANET**	**Security problem**	**Accuracy**
Nakayma et.al	**trusted secure AODV**	**MANET**	**Attack prevention**	**PDF, AED, AT, NRL.**
Boukerche et.al	**AODV**	**MANET**	**Routing problem**	**Accuracy**
W. Wu et al	**AODV**	**MANET**	**Routing protocol efficiency**	**accuracy**
Parma nand et al	**DSR**	**Qualnet**	**routing**	**number of hops per route.**
Samir et.al	**DSR**	**MANET**	**Mobile pattern analysis**	**accuracy**
Chetna et al	**AODV**	**NS2**	**Black hole detection**	**efficiency**

Sagar et.al	**AODV**	**MANET**	**Black hole detection**
Mariappan Kadar	**Hybrid**	**MANET**	**efficiency**

III. MACHINE LEARNING IN WIRELESS SENSOR NETWORKS

Usually, sensor network designers characterize machine learning as a collection of tools and algorithms that are used to create prediction models. However, machine learning experts recognize it as a rich field with very large themes and patterns. Understanding such themes are beneficial to those who wish to apply machine learning to WSNs. Applied to numerous WSNs applications, machine learning algorithms provide tremendous flexibility benefits. This section provides some of the theoretical concepts and strategies of adopting machine learning in the context of WSNs. Existing machine learning algorithms can be categorized by the intended structure of the model. Most machine learning algorithms fall into the categories of supervised, unsupervised and reinforcement learning.

A. Supervised Learning

In supervised learning, a labeled training set (i.e., predefined inputs and known outputs) is used to build the system model. Neural networks are made up of simple components functioning in parallel. These components are stimulated through biological nervous systems. As per in their nature, the connections amongst numerous components mostly define the specific network function [7, 8]. An individual could easily train a NN to accomplish a particular function by means of amending the values of the weights (connections) amongst several components. Normally, neural networks are trained, or adjusted, so, particularly, input directs to a precise target output. The subsequent figure demonstrates such circumstance. At this point, the network is agreed, dependent on a comparison of the O/P in addition to the target, unless the network O/P matches the actual target. Typically, such types of input/target pairs are required to train a network. With this rule, as through other sorts of backpropagation techniques, 'learning' is a supervised procedure which take place with every single cycle or 'epoch' which is demonstrated with a novel input pattern through a forward activation flow of O/Ps, in addition to the backwards error propagation of weight amendments.

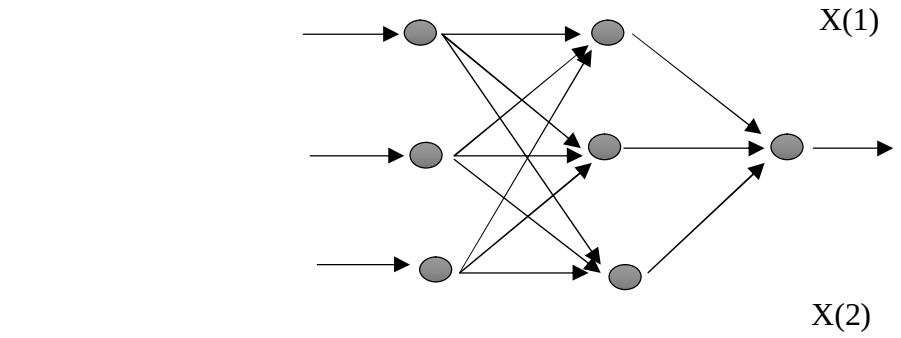

Figure 2: Neural network architecture

Neural Network, principally, demonstrated with a specific scheme, this makes an arbitrary 'guess' as in the direction of what it might be. These networks have been trained in the direction of performing complex functions in numerous areas, which also encompasses speech, pattern recognition, vision, control systems, identification, and classification. Neural networks could also be trained towards resolving issues which are challenging for conventional computers or human beings. Even though, there are numerous different types of learning rules utilized via neural networks, this specific demonstration is concerned simply with one and only the delta rule. This specific delta rule is so often used by the utmost common class of Artificial Neural Networks entitled as feed forward neural networks. The training criteria of NN can be summarized below:

i. Input is given to input neurons.
ii. Obtained output response is compared to input data.
iii. Error data is utilised to manage the weights attached to neurons.
iv. Hidden units find outs its error during back signal.
v. Then weights gets updated in the end.

B. Unsupervised Learning

Unsupervised learners are not provided with labels (i.e., there is no output vector).
Principal component analysis is a classic method used to compress higher dimensional data sets to lower dimensional for data analysis, apparition, feature extraction, or data compression. PCA involves the calculation of the Eigen value decomposition of a data covariance medium or singular value decay of a data matrix, usually after mean centering the data for each attribute [9, 10].
Following are the steps that are basically used for the execution of PCA:
Step 1: Get normalizes data from the iris regions. 2-D iris image is represent as 1-D Vector by concatenating each row (or Column) into a long vector
Step 2: Take away the mean image from each image vector. Mean should be row wise.
Step 3: For calculating the eigen vectors and eigen values, Compute the covariance matrix.
Step 4: Analyze the eigenvectors and Eigen values of the covariance matrix.

Step 5: The eigenvectors are sorted from high to low according to their corresponding Eigen values. Choose components and forming a feature vector.
Step 6: Derive the new data set once we have chosen the components, we simply take the transpose of the vector and increase it on the left of the original data set, transposed.

$$Final\ Dataset = RowFeatureVector \times Row\ Mean\ Adjust$$

Where RowFeatureVector is the matrix with the eigenvectors in the columns transposed so that the eigenvectors are now in the rows, with the most major eigenvector at the top, and RowMeanAdjust is the mean used to data transposed. The data items are in each editorial, with each row holding a split dimension. Principal components analysis is basically useful for dropping the number of variables that consists a dataset while retaining the contradiction in the data and to identify unknown patterns in the data and to classify them according to how much of the information, stored in the data, they report for. [10]

C. Reinforcement Learning

Reinforcement learning enables an agent (e.g., a sensor node) to learn by interacting with its environment.The Self-Organizing Map is one of the commonly used network model. It belongs to the learning networks. The Self-Organizing Map is un-supervised learning method. If Self-Organizing Map is used for feature extraction then it is called Self-Organizing Feature Map [11]. Below figure shows that there are 5 cluster units Yi and 7 input units Xi. Clusters are arranged in linear array.

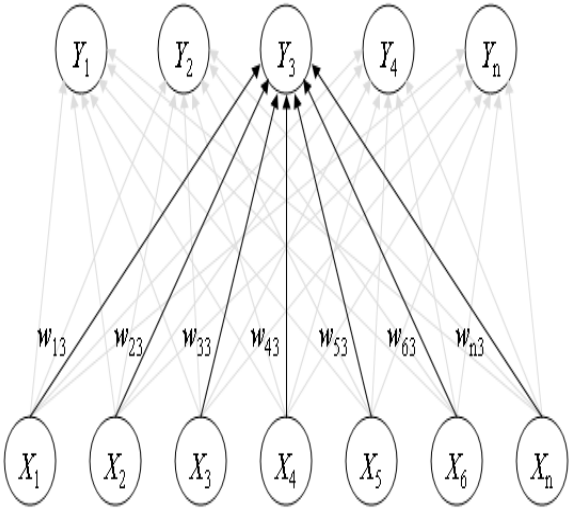

Figure 3: SOM Example

Self-Organizing Map was designed by Kohonen. The SOM has been useful in many applications. It maps the high dimensional space to map units for preserve mapping. Neuron units commonly made lattice onto a plane. Preserving property means reserving the distance between points. In addition to that Self-Organizing Map has the capability of generalizing. It means recognizing the patterns that never met before. The Self-Organizing Map I 2-D can be represented as following:

$$Y = \{x \ldots\ldots\ldots\ldots, x_{acw}\}$$

The neurons are connected to adjacent neurons by a relation. Commonly, the neurons are connected

IV. CONCLUSION

Wireless sensor networks are different from traditional network in number of aspects, thereby, necessitating protocols and tools that address unique challenges and limitations. As a consequence, wireless sensor networks require innovative solutions for energy aware and real-time routing, security, scheduling, localization, node clustering, data aggregation, fault detection and data integrity. Machine learning provides a collection of techniques to enhance the ability of wireless sensor network to adapt to the dynamic behavior of its surrounding environment.
place. But the very notion of field marks the guidelines of how such relationship is to be carried out.

REFERENCES

[1] T. O. Ayodele, "Introduction to machine learning," in New Advances in Machine Learning. InTech, 2010.
[2] A. H. Duffy, "The "what" and "how" of learning in design," IEEE Expert, vol. 12, no. 3, pp. 71–76, 1997.
[3] P. Langley and H. A. Simon, "Applications of machine learning and rule induction," Communications of the ACM, vol. 38, no. 11, pp. 54– 64, 1995.
[4] L. Paradis and Q. Han, "A survey of fault management in wireless sensor networks," Journal of Network and Systems Management, vol. 15, no. 2, pp. 171–190, 2007.
[5] B. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," in 22nd International Conference on Distributed Computing Systems Workshops, 2002, pp. 575– 578.
[6] J. Al-Karaki and A. Kamal, "Routing techniques in wireless sensor networks: A survey," IEEE Wireless Communications, vol. 11, no. 6, pp. 6–28, 2004.
[7] S. Das, A. Abraham, and B. K. Panigrahi, "Computational intelligence: Foundations, perspectives, and recent trends," John Wiley & Sons, Inc., 2010, pp. 1–37.
[8] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin, "Learning from data," AMLBook, 2012.
[9] S. Soro and W. Heinzelman, "Prolonging the lifetime of wireless sensor networks via unequal clustering," in 19th IEEE International Parallel and Distributed Processing Symposium, 2005, pp. 4–8.
[10] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks".
[11] S. Yun, J. Lee, W. Chung, E. Kim, and S. Kim, "A soft computing approach to localization in wireless sensor networks," Expert Systems with Applications, vol. 36, no. 4, pp. 7552–7561, 2009.
[12]Sangeetha Kannan et.al, "Secure Data Transmission in MANETs using AODV", IJCCER, Vol. 2, 2014.
[13] H. Nakayama, S. Kurosawa, A. Jamalipour, Y. Nemoto and N. Kato, "A Dynamic Anomaly Detection Scheme for AODV-Based Mobile Ad Hoc Networks," in *IEEE Transactions on Vehicular Technology,* vol. 58, no. 5, pp. 2471-2481, Jun 2009.
[14]Azzedine Boukerchea, 1, , Begumhan Turgutb, 2, , Nevin Aydin, "Routing protocols in ad hoc networks: A survey", Computer Networks, Vol. 55, Issue 13, 15, pp. 3032–3080, 2011.
[15]C. W. Wu, T. C. Chiang and L. C. Fu, "An ant colony optimization algorithm for multi-objective clustering in mobile ad hoc networks," 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, 2014, pp. 2963-2968.
[16]Parma Nand, S. C. Sharma, "Comparison of Routing Protocols for MANET and Performance Analysis of DSR Protocol", Advances in Computing, Communication and Control, vol. 125, 2011.
[17]Samir R. DasRobert CastañedaJiangtao Yan, "Simulation-based performance evaluation of routing protocols for mobile ad hoc networks", Mobile Networks and Applications, vol. 5, 2000.
[18]Chetana Khetmal1, Prof.Shailendra Kelkar2, Mr.Nilesh Bhosale, "MANET: Black Hole Node Detection in AODV," International Journal of Computational Engineering Research, Vol, 03, 2013.
[19]S. R. Deshmukh and P. N. Chatur, "Secure routing to avoid black hole affected routes in MANET," 2016 Symposium on Colossal Data Analysis and Networking (CDAN), Indore, 2016, pp. 1-4.
[20]Marichelvam, Mariappan Kadarkarainadar, Thirumoorthy Prabaharan, and Xin She Yang, "A discrete firefly algorithm for the multi-objective hybrid flowshop scheduling problems," IEEE transactions on evolutionary computation, vol.18, pp. 301-305, 2014.