INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2022, VOL. 68, NO. 4, PP. 799–805
Manuscript received December 21, 2021; revised November, 2022. DOI: 10.24425/ijet.2022.143888

Performance Analysis of LEACH with Deep Learning in Wireless Sensor Networks

Hardik K Prajapati, and Rutvij Joshi

***Abstract*—Thousands of low-power micro sensors make up Wireless Sensor Networks, and its principal role is to detect and report specified events to a base station. Due to bounded battery power these nodes are having very limited memory and processing capacity. Since battery replacement or recharge in sensor nodes is nearly impossible, power consumption becomes one of the most important design considerations in WSN. So one of the most important requirements in WSN is to increase battery life and network life time. Seeing as data transmission and reception consume the most energy, it's critical to develop a routing protocol that addresses the WSN's major problem. When it comes to sending aggregated data to the sink, hierarchical routing is critical. This research concentrates on a cluster head election system that rotates the cluster head role among nodes with greater energy levels than the others. We used a combination of LEACH and deep learning to extend the network life of the WSN in this study. In this proposed method, cluster head selection has been performed by Convolutional Neural Network (CNN). The comparison has been done between the proposed solution and LEACH, which shows the proposed solution increases the network lifetime and throughput.**

***Keywords*—Machine learning; Deep learning; Convolutional Neural Network (CNN); LEACH**

I. Introduction

WIRELESS sensor networks are the most powerful and significant technology of the twenty-first century. It requires neither any specific infrastructure nor any monitoring, thus it provides a fresh route for its implementation in diverse applications [1]. Sensor networks are simple to set up, but they need that data get to its destination in a timely way. The most significant element in a wireless sensor network's life is its energy consumption, which is restricted due to the sensor node's tiny battery size. The three major roles of a sensor node are sending, receiving, and sensing. The function that consumes the most energy is data transmission. As a result, if we want to extend the network's life span, we'll need energy-efficient routing techniques. The routing protocol specifies the fastest and most energy-efficient route for data to be sent to the sink [2]. Data can be transmitted across intermediary nodes if there isn't a direct link between source and base station, which is known as multi-hop communication. Because computing consumes less energy than transmission, it is always preferable to calculate data rather than communicate it. There is no data protection in WSN, and the bandwidth is limited.

Hardik K Prajapati and is with Gujarat Technological University, Ahmedabad, Gujarat, India (e-mail: hardikprajapati@gmail.com).
Rutvij Joshi is with Parul University, Vadodara, Gujarat, India (e-mail: rutvij.joshi19624@paruluniversity.ac.in).

Clustering-based protocols adjust energy usage by balancing all nodes to become the cluster head. Choosing the right cluster head can help to balance the load in the sensor network, lowering energy consumption and increasing lifespan. The nodes are divided into tiny regions, each of which is overseen by a Cluster Head, which is selected by a set of rules. Every member of the cluster transmits data to the head node. The cluster head processes data that comes in from multiple nodes and forwards it to the sink either directly or indirectly through another cluster head. Every routing protocol has its own clustering method, cluster head selection, and data transmission strategy. The application has a big impact on how sensor networks are deployed. As a result, the most difficult challenge in WSN energy optimization is routing. Cluster heads are used to reduce inter-node interference. The allocation of resources becomes more efficient, and communication expenses are reduced [3].

The formation of clusters and the selection of cluster heads are crucial processes. The cluster head and the nodes should be within a reasonable distance of each other. The clusters should be small, to meet this criterion, which will increase the number of clusters. The drawback of this strategy is that adding additional CHs would increase network traffic and soon drain network energy. Increased cluster size, but at the other hand, lowers the number of cluster heads. However, non-CH nodes will have to use a lot of energy in order to reach the CH, which will cause them to lose energy and die. As a result, clusters should be designed so that sensor nodes are close to the cluster head and power efficiency is significant. When the cluster is formed and the cluster head is designated, the cluster head uses the TDMA technique to create the scheduling table. According to the slots granted by the CH, the sensor nodes will communicate with the head node [4].

II. Literature Survey

Wireless communications and mobile computers are evolving rapidly, and mobile components are becoming increasingly prevalent. WSNs, like many other technologies, were originally used for military objectives, including as battlefield monitoring. Following that, they were put to commercial usage. WSNs have unquestionably become an important technology for a variety of smart settings nowadays. Wireless sensor networks (WSNs) are made up of small autonomous devices called sensor nodes that communicate through radio connections. The WSN has sparked a lot of interest in scientific study, especially because of unique routing challenges posed by network lifespan limitations and limited node processing capacity [5]. Due to numerous restrictions, wireless sensor networks (WSNs) allow novel applications and necessitate non-traditional protocol design paradigms.

A good balance between communication and signal/data processing capabilities must be discovered due to the necessity for minimal device complexity along with low energy consumption (i.e. extended network lifespan). Since the last decade, this has motivated a significant amount of effort in research, standardization, and industrial investments in this field. Due to their versatility in solving issues in a range of application areas, wireless sensor networks have grown in popularity and have the potential to transform our lives in a variety of ways. WSNs have been successfully applied in various application domains such as military applications, area monitoring, transportation, health application, environment sensing, structure testing, agriculture sector etc. Sensor nodes are obligated to execute all duties, including sensing phenomena, data processing, transmission, packet forwarding from other nodes, and data reception. However, because they may be deployed at the bottom of the ocean or in the mountains, they are inaccessible and the battery cannot be replaced. As a result, they have very limited battery power, which must be used as effectively as possible to ensure the network's lifespan [6].

With the development of IoT and 5G, the need for bandwidth and data rate has skyrocketed. As a result, it is a top priority for network operators to create communication protocols that meet all QoS criteria. Traditional routing protocols were created decades ago, and the fundamental issue with them is that they are not self-adaptive. As a result, it is critical now to build self-learning routing protocols, which may be accomplished through the application of deep learning [7]. Throughout history, artificial intelligence (AI) has made tremendous advances, and it now has a wide range of applications, including e-health, intelligent control, pattern recognition, and so on. In the case of big data, AI has played a critical role in achieving high efficiency and flexibility. Traffic prediction and classification are two of the first applications of machine learning in the networking field. Because of the well-crafted question descriptions and requests from diverse networking subfields, research on the two issues has always maintained a certain level of interest [8]. Deep learning has emerged as a feasible way for network operators to build and operate their networks in a more intelligent and autonomous manner, owing to recent achievements in the field of artificial intelligence. Deep learning has piqued interest in robotics, self-driving cars, forecasting, and plenty of other fields. Deep learning may also be used to tackle a variety of problems in the field of wireless sensor network [9].

Clustering is often regarded as the most difficult unsupervised learning task. Clustering's main goal is to discover a pattern in a set of unlabelled data [10]. The practice of arranging items into groups whose members are related in some way is how clustering is defined. A cluster is defined as a group of data items that are similar to one another but different from data points in other clusters. The data is split into clusters based on some criterion, most often a distance: two or more data points belong to the same cluster if their distances are near. Distance-based clustering is the name for this form of grouping. Another form of clustering is conceptual clustering, in which two or more entities are assigned to the same cluster if they have a common idea. There is no definitive best criterion that is independent of the clustering's ultimate goal. As a result, it is the user's role to choose this criterion in such a way that the clustering result meets the requirements.

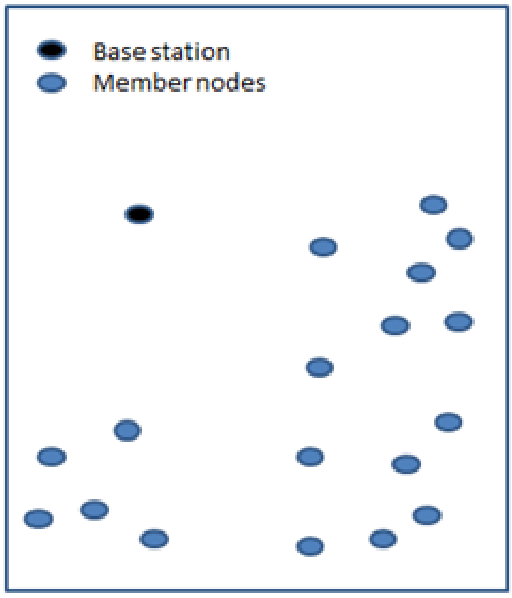

Fig. 1. Wireless sensor network.

III. LEACH Protocol

LEACH is a hierarchy based routing protocol, which divides the area in small clusters. A small number of sensor nodes are chosen at random to serve as cluster heads (CHs), and their roles are switched to equally divide the energy burden across the network.

The goal is to use local cluster heads as sink routers and organize sensor nodes into clusters depending on received signal strength. Cluster Heads compress data from member nodes and transmit an aggregated data to the base station to effectively optimize the amount of data that must be sent to the BS. To reduce inter- and intra-cluster interference, LEACH uses a time division multiple access / code-division multiple access as medium access protocol. Fig. 1 shows deployment of wireless sensor nodes before clustering. Fig. 2 shows the whole network has been divided in 3 clusters and each cluster has elected one cluster head which is connected to base station. The setup phase and the steady state phase are the two phases of LEACH operation. The nodes are grouped into clusters and CHs are chosen during the setup process. The heart of the LEACH protocol is the cluster head election. First, set a threshold (Tn), and then assign different random values to each node in each cycle. In the current round, if the sensor

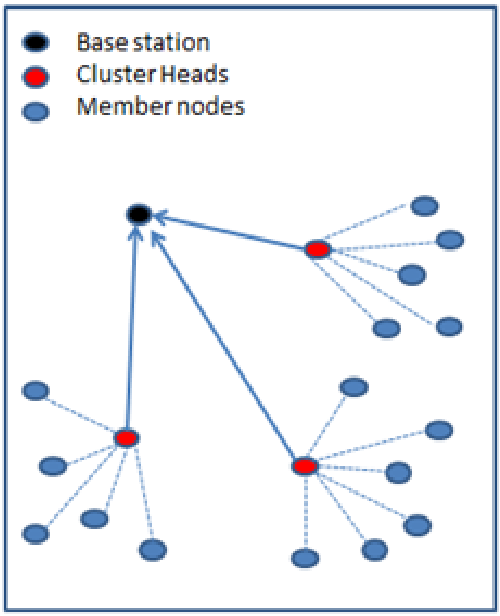

Fig. 2. WSN after cluster formation and CH selection

node's random value is less than Tn, the node functions as a CH. The Tn is given as:

$$T(n) = \begin{cases} \dfrac{P}{1 - P * \left(r \mod \dfrac{1}{P} \right)} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where p signifies the probability of choosing a node as CH in r rounds, and G defines a collection of nodes that have not been chosen as CHs in 1/ p rounds. As a result, the CH will be required to seek for re-election, and the number of cluster heads will fluctuate greatly. On the one hand, having too many CHs will overload the network since the CH must perform data fusion on the received data before transferring it to the base station. However, with fewer cluster heads, each cluster's coverage area will be too large, resulting in increased data transmission energy usage. The single-hop transmission method is used by all nodes in this strategy. If the transmission distance is too wide, the CH will spend a lot of energy to transmit the data, which may lead the CH to expire prematurely due to energy depletion [6].

IV. CONVOLUTIONAL NEURAL NETWORK

Convolutional neural networks (CNNs), a type of ANN that has been popular in computer vision, are gaining popularity in various fields like network traffic control, traffic classification & prediction, video analysis and so on [11]. The following are some of the significant reasons why CNN is preferred above other traditional methods. First, the main motivation for utilizing CNN is the concept of weight sharing, which reduces the number of parameters that need to be trained, resulting in better generalization. CNN can be trained smoothly and without over fitting because of the fewer parameters. Second, the classification and feature extraction stages are merged, both of which are based on learning. Third, building large networks with generic models of artificial neural networks (ANN) is far more challenging than doing so with CNN [12]. There are three sorts of layers in CNNs. Convolutional layers, pooling layers, and fully-connected layers are the three types. CNN architecture is produced when these layers are layered. The flow of CNN to process an input picture and classify objects based on values is depicted in the Fig. 3 below [13].

The core functionality of the CNN may be divided into four distinct sections.

1. The input layer, like in other types of ANN, will store the image's pixel values.

2. The convolutional layer will calculate the scalar product between their weights and the region related to the input volume to identify the output of neurons connected to particular parts of the input. The rectified linear unit (often abbreviated as ReLu) is designed to apply 'element wise' activation function such as sigmoid to the output of the previous layer's activation.

3. The pooling layer will next conduct down sampling along the input's spatial dimensions, decreasing the number of parameters inside that activation even further.

4. The fully-connected layers will then attempt to create class scores from the activations, which will be utilized for classification, in the same way that regular ANNs do. It's also possible that ReLu may be employed between these layers to boost performance [14], [15] . CNNs may modify the original input layer by layer utilizing convolutional and down sampling techniques to create class scores for classification and regression using this basic method of transformation.

A. *Convolution Layer:*

The first layer to extract features from an input picture is convolution. By learning picture characteristics with tiny squares of input data, convolution retains the connection between pixels. It's a mathematical procedure with two inputs: an image matrix and a filter or kernel.

B. *Pooling Layer:*

When the pictures are too huge, the pooling layers portion would minimize the number of parameters. Spatial pooling, also known as subsampling or down sampling, lowers the dimensionality of each map while preserving critical data. There are several forms of spatial pooling: Maximum Pooling, Average Pooling, Sum Pooling, shown as below in Fig. 6.

The biggest element from the corrected feature map is used in max pooling. The average pooling might be obtained by taking average of the elements. Sum pooling refers to the sum of all elements in a feature map.

C. *Fully Connected Layers:*

The fully-connected layer is made up of neurons that are directly linked to the neurons in the two neighboring layers, but not to any of the layers inside them. This is similar to how neurons are organized in classic ANN models. We flattened our matrix into vector and sent it into a fully connected layer,

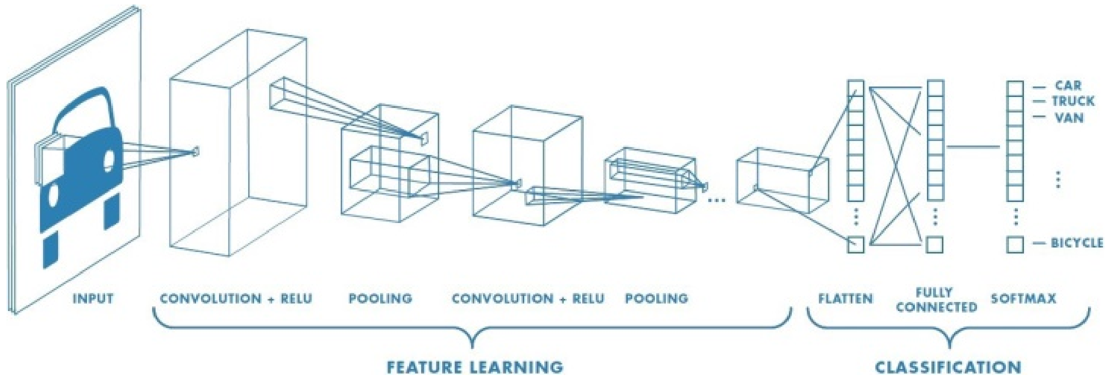

Fig. 3. CNN Architecture

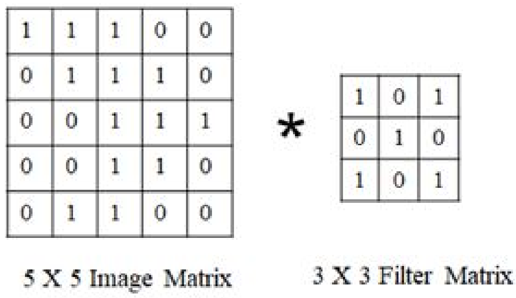

Fig. 4. Multiplication of Image matrix and Filter matrix

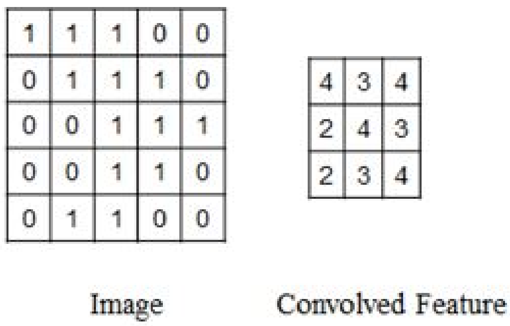

Fig. 5. Feature Map

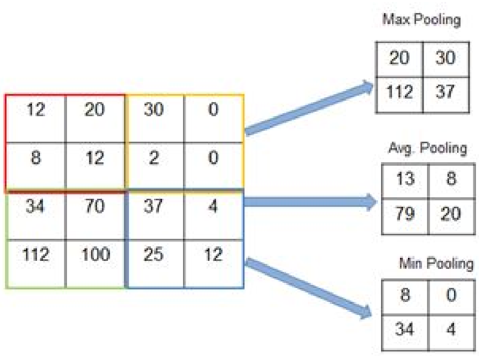

Fig. 6. Types of Pooling

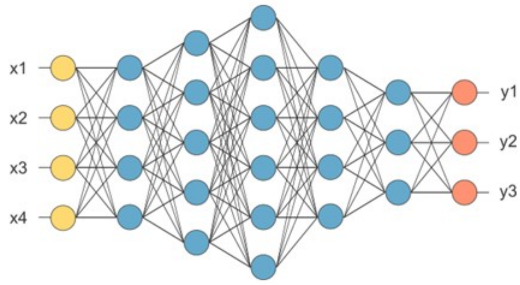

Fig. 7. Fully Connected Layers

similar to a neural network, in the FC layer. The feature map matrix will be transformed to a vector (x1, x2, x3...) as shown in Fig. 7 We put these attributes together to make a model using the fully linked layers. Finally, we have a softmax or sigmoid activation function to categorize the outputs as cat, dog, automobile, truck, and so on.

V. Proposed Methodology

In a wireless sensor network, the monitoring region is split into numerous high-density sensor nodes. This region is split into tiny clusters in LEACH, with one cluster head for each cluster. The choice of CH is important for the network's lifespan, and here is where the LEACH protocol falls well short. In LEACH, clusters are generated in real time, and cluster heads are chosen at random. Each node in the cluster has an equal chance of becoming CH. The CH selection in this technique is not based on probability, but rather on the use of a Convolutional Neural Network. The network is taught by feeding it leftover energy from nodes. Each node's energy is evaluated after each round, and the node with the most energy is chosen as CH. Training has been done in two phases:

1. A forward phase in which the input is sent through the network, distance and energy.

2. Gradients are backpropagated and weights are modified in this step.

To train our CNN, we have followed this method. We've additionally employed two key implementation-specific concepts:

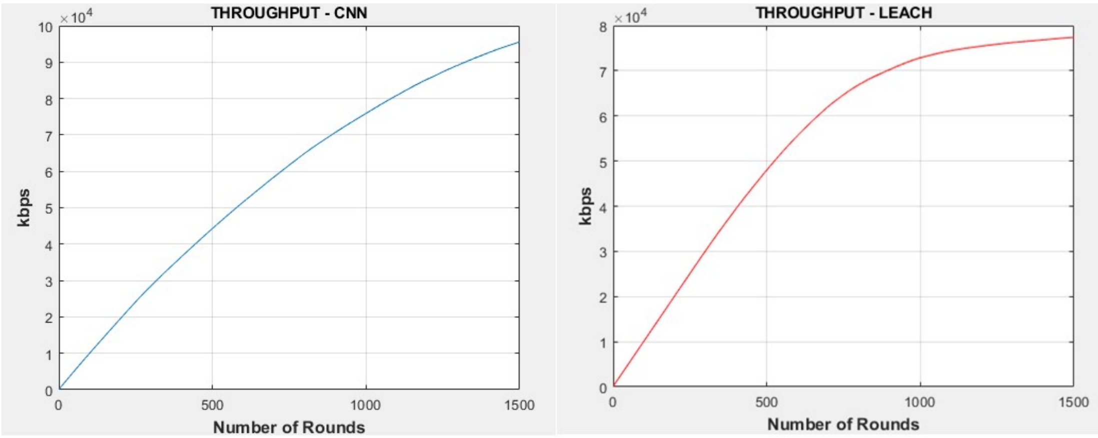

Fig. 8. Comparison of Throughput between traditional LEACH and CNN-LEACH

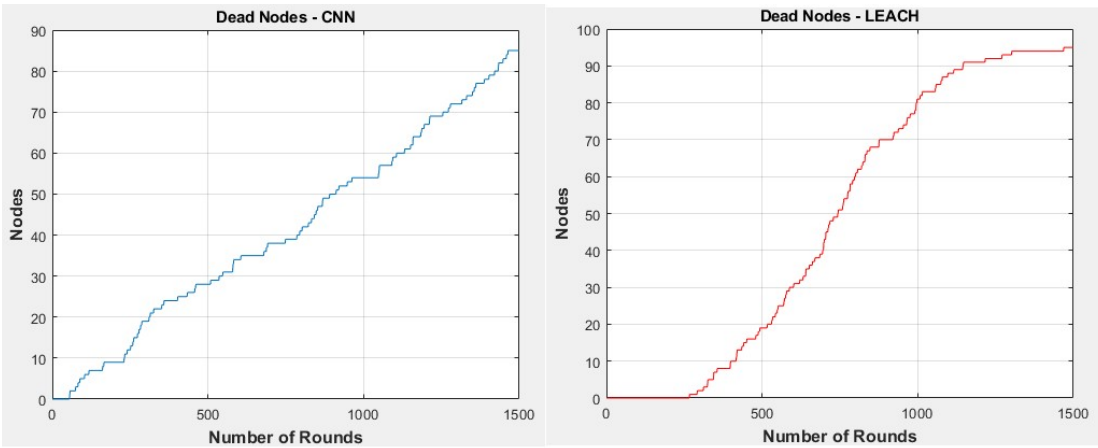

Fig. 9. Comparison of dead nodes between traditional LEACH and CNN-LEACH

• During the forward phase, each layer will store energy value, it will require for the backward phase. This implies that each backward phase must be followed by a forward phase.

• Each layer will receive a gradient and will also return a gradient during the backward phase. It will receive the loss gradient with regard to its inputs and will return the loss gradient with respect to its inputs.

It will gather data from non-CH nodes and transfer it to the base station after selecting CH. The same routing will be used as in LEACH.

VI. Simulation Results

A topology of 100 static nodes is constructed in MATLAB to compare classical LEACH with CNN-based Leach. Throughput, live nodes, and network residual energy are the metrics that are compared. The network area is 300x300, and the performance was tested for 1500 rounds in total. The key difference between these two protocols is that in Normal LEACH, the cluster head is chosen by probability, whereas in DL, the cluster head is chosen using CNN. When compared to Traditional LEACH, the output reveals that employing CNN for cluster head selection improves performance significantly. The selected parameters are shown in Table I

TABLE I
Parameters of the Network

Parameters	Value
Network Area	300x300
Initial Energy	0.5 J/Node
Number of Nodes	100
Number of Rounds	1500
Energy required to run the transmitter	5 nJ
Energy Spent per bit	10 pJ
Energy for data aggregation	5 nJ
Message size	4000 bits
Probability node will become CH	0.05
Software used	MATLAB

Fig. 8 shows the throughput in CNN-LEACH and traditional LEACH respectively. We can see that, after 1500 rounds the throughput achieved in LEACH is 78 kbps and in LEACH with CNN it is 95 kbps. Which is an overall improvement of approximately 17 kbps. This is because CH is not selected based on the probability value only. The CNN has been trained properly and then it selects the CH based on the residual energy of the node. It increases the packet delivery ratio.

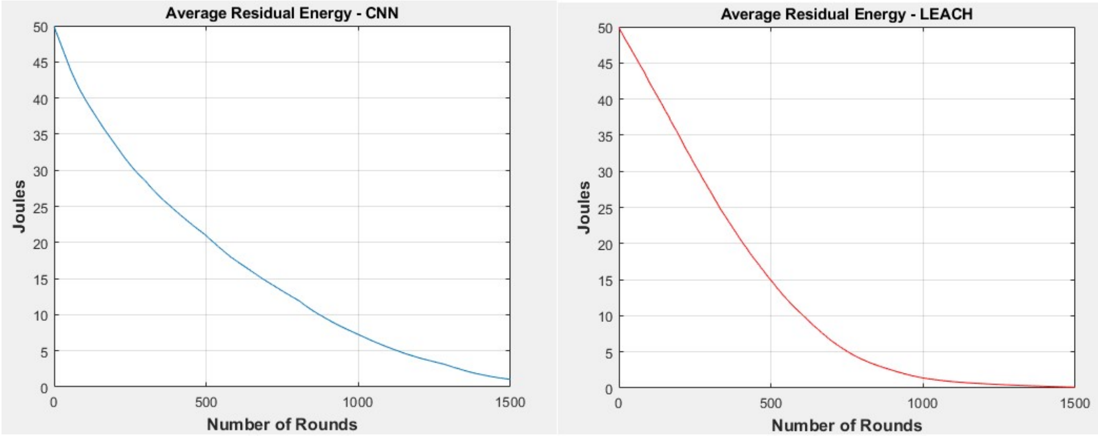

Fig. 10. Comparison of residual energy between traditional LEACH and CNN-LEACH

TABLE II
PERFORMANCE COMPARISONS OF TRADITIONAL LEACH AND LEACH WITH CNN

Sr. No.	Parameters	Traditional LEACH	LEACH using CNN
1	Throughput(kbps)	75	92
2	Dead Nodes	95	85
3	Residual Energy (Joules)	0	5

In Fig. 9, we can see that the traditional LEACH having more dead nodes than the LEACH with CNN, which are 95 & 85 respectively. This is also because of the better solution of CH selection process. If the CH is selected properly the non CH nodes need to spend less energy for communication. It is apparent from Fig. 10 that LEACH CNN performs better than standard LEACH in terms of energy dissipation. After 1200 rounds the traditional LEACH network consumes all of its energy, whereas the CNN LEACH network has 5 J of energy left. It shows that the deep learning approach to select the cluster head is more efficient in terms of energy than traditional LEACH. Table II shows the performance comparison between traditional Leach and LEACH with CNN.

VII. CONCLUSION

In a WSN, several sensor nodes are distributed throughout a broad region to sense various sorts of characteristics or identify specific phenomena and communicate with the network administrator or base station. The rapid depletion of node energy produces a significant number of black holes in the network core, resulting in data redundancy, data packet retransmission, end-to-end latency, and route update costs. Many studies have shown that LEACH is a relatively energy-efficient routing protocol, although it still has flaws and limitations, such as higher energy usage owing to an inadequate cluster head selection technique. We have coupled the benefits of a deep learning method with LEACH to increase network performance in this study. In comparison to the standard LEACH procedure, the simulation results clearly indicate that the suggested methodology provides higher performance. The most essential requirement for WSN, network life time, is enhanced by using deep learning to choose the CH. Different neural networks can be trained in the future work to choose the optimum CH and to develop routing protocols using deep learning. Cluster formation and CH selection may both be done using different optimization algorithms.

REFERENCES

[1] B. Rashid and M. H. Rehmani, "Applications of wireless sensor networks for urban areas: A survey," *Journal of Network and Computer Applications*, vol. 60, pp. 192–219, 2016. [Online]. Available: https://doi.org/10.1016/j.jnca.2015.09.008

[2] P. S. Mann and S. Singh, "Energy-efficient hierarchical routing for wireless sensor networks: A swarm intelligence approach," *Wireless Personal Communications*, vol. 92, pp. 785–805, 2017. [Online]. Available: https://doi.org/10.1007/s11277-016-3577-1

[3] P. Nayak and A. Devulapalli, "A fuzzy logic-based clustering algorithm for wsn to extend the network lifetime," *IEEE Sensors Journal*, vol. 16, no. 1, pp. 137–144, 2016. [Online]. Available: https://doi.org/10.1109/JSEN.2015.2472970

[4] W. Abidi and T. Ezzedine, "New approach for selecting cluster head based on leach protocol for wireless sensor networks," in *ENASE*, 2017.

[5] V. Rahmati, "Near optimum random routing of uniformly load balanced nodes in wireless sensor networks using connectivity matrix," *Wireless Personal Communications*, vol. 116, pp. 1–17, 02 2021. [Online]. Available: https://doi.org/10.1007/s11277-020-07829-7

[6] M. Ahmad Jan, P. Nanda, M. Usman, and X. He, "Pawn: a payload-based mutual authentication scheme for wireless sensor networks: Pawn," *Concurrency and Computation: Practice and Experience*, vol. 29, 10 2016. [Online]. Available: https://doi.org/10.1002/cpe.3986

[7] Q. Mao, F. Hu, and Q. Hao, "Deep learning for intelligent wireless networks: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 20, no. 4, pp. 2595–2621, 2018. [Online]. Available: https://doi.org/10.1109/COMST.2018.2846401

[8] Y. Liang and H. Mei, "Dynamic bandwidth allocation based on online traffic prediction for real-time mpeg-4 video streams," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, 01 2007. [Online]. Available: https://doi.org/10.1155/2007/87136

[9] Z. M. Fadlullah, F. Tang, B. Mao, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2432–2455, 2017. [Online]. Available: https://doi.org/10.1109/COMST.2017.2707140

[10] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005. [Online]. Available: https://doi.org/10.1109/TNN.2005.845141

[11] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, no. 4, pp. 611–629, 2018. [Online]. Available: https://doi.org/10.1007/s13244-018-0639-9

[12] S. Indolia, A. Goswami, S. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network- a deep learning approach," *Procedia Computer Science*, vol. 132, pp. 679–688, 01 2018. [Online]. Available: https://doi.org/10.1016/j.procs.2018.05.069

[13] T. M. Behera, S. K. Mohapatra, U. C. Samal, M. S. Khan, M. Daneshmand, and A. H. Gandomi, "Residual energy-based cluster-head selection in wsns for iot application," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5132–5139, 2019. [Online]. Available: https://doi.org/10.1109/JIOT.2019.2897119

[14] J. Wu, "Introduction to convolutional neural networks," 2017.

[15] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *ArXiv*, vol. abs/1511.08458, 2015. [Online]. Available: https://doi.org/10.48550/arXiv.1511.08458