Journal of Scientific & Industrial Research
Vol. 80, December 2021, pp. 1091-1095

Data Congestion Prediction in Sensors Based IoT Network

Rajesh K Yadav[1], Aastha Maheshwari[1]* and Prem Nath

[1]Department of computer Science and Engineering, Delhi Technological University Delhi, 110 042, India

[2]Department of computer Science and Engineering, H N B Garhwal University, Uttrakhand, 246 174 India

Received 20 July 2021; revised 02 December 2021; accepted 02 December 2021

Internet of Things (IoT) becoming the major part of human life and make the life simpler. IoT uses sensor nodes to monitors certain phenomena and transmitting the collected information to the IoT gateway. The size of the network is increase rapidly and causing congestion in the network and results in network delay, loss of data packets, a decrease in throughput, and poor energy efficiency. It is important to predict the congestion and mitigate the data accordingly. To resolve the problem of congestion, our focus is on predicting the congested node effectively. We propose an optimized deep neural network - Restricted Boltzmann machine (DNN-RBM) based data congestion prediction approach which is used for analyzing and predicting the congested node in the sensors based IoT environment. To enhance the performance of DNN, the weight parameters of DNN are optimized using the Restricted Boltzmann Machine (RBM)-algorithm. The dataset is used to train the model and enable the prediction to find the congested nodes in the network with more accuracy to enhance the performance of the network. The performance factors congestion window, throughput, propagation delay, RTT, number of packets sent, and packet loss are given as input by using DNN-RBM. Predicted results show that the proposed DNN-RBM model predicts congestion with more than 95% accuracy as compared with other models like ANN, DNN-GA.

Keywords: Congestion, Deep neural network (DNN), Internet of Things (IOT), Restricted Boltzmann machine (RBM), Wireless sensor networks (WSNs)

Introduction

The Internet of Things (IOT), because of the need for different network integration types, receives great attention in research. The goal of IOT is to connect a broad range of data and information sharing devices and services. All the people now use various types of communication devices, all connected to the internet. In the last two decades, the concept of IOT[1] has come to light and numerous scientists and industries work on it. IOT is committed to making our everyday lives and our society easier. The Smart Grid[2], smart city, environmental surveillance[3], and healthcare monitoring systems, etc., are the main scenarios of IOT. CompTIA's predicted in 2020 more than 50 billion devices present in the network.[4] The whole network communication and computer scenario can thus be modified. Most linked devices however have a high degree of processing power, restricted storage space, and energy constraints.[5] A high level of wireless restricted devices connected to the Internet is envisaged as the future of the Internet of Things (IOT). Wireless Sensor Networks (WSNs) built into the Internet allow the virtual world to be connected autonomously and intelligently to the physical world.

Therefore, WSNs are introducing several new, real-world IOT app services. For over 10 years, efforts have been made to integrate WSN with the web, in particular with the development of many important WSN applications. WSN is also a significant technology for IOT, with its integration in the cloud[6] and the internet.

Network Congestion is a fundamental issue found in the computer network by numerous researchers. Now, IOT's all linked to the Internet. The increase in the number of connected devices via the Internet would also increase network congestion.[7]

In this paper we are firstly predicting the congested nodes in the network, for this a Deep Neural Network (DNN) is provided. The performance of the DNN is enhanced by optimizing the weight parameters. To optimize the weight parameters of the DNN, the Restricted Boltzmann Machine (RBM) algorithm is presented. And the exhibition of the proposed approach analyzed the throughput, congestion window, propagation delay, and accuracy. The results of the proposed model can be used for offloading of data packets from the congested node to other IOT devices.

*Author for Correspondence
E-mail: aasthamaheshwari.am@gmail.com

Method

Approach

To solve the problem of congestion detection in the IOT network, an efficient technique is presented in our research work. To identify the data congestion in the WSN based IOT organization, and DNN-RBM is proposed right now. The node parameters are given as input to the machine learning algorithm. Based on the input parameters, the proposed machine learning algorithm decides whether a node is congested or not. As the network is using wireless communication medium, hence there is a high probability of data corruption and the presence of noise in the network.

The algorithm is consisting of major 2 steps; the first step is to create the dataset with effective parameters to predict the congestion in the node. The second step is to detect data congestion using DNN-RBM.

The dataset is created by simulating the IoT network in MATLAB by plotting 250 sensor nodes randomly in 1000 × 1000 area. Initial energy of the node set to 10.3 J and initial transmission power set to 0.66 W. Routing protocol used for communication is AODV and packet size is 512 bytes. Each node consists set of parameters as congestion window, throughput, propagation delay, RTT, number of packets sent, and packet loss for each sample. This dataset is used as an input to the machine learning model that helps in the prediction of congestion.

Deep Neural Network (DNN) for Congestion Prediction

An Artificial Neural Network (ANN) is a DNN consisting of several hidden layers between the output and input layers as shown in Fig. 1. Deep learning methods are extremely efficient when there are large numbers of samples in the course of the training. This characterizes the suggested congestion forecast using the technique based on DNN. With a deep neural network preparation, a large number of neurons are regenerated in each phase till the error between yield and information is not within tolerance. This is time-expending. In the proposed approach we are using the DNN to predict the congestion in network. For this prediction, the performance factors congestion window, throughput, propagation delay, RTT, number of packets sent, packet loss is given as input, and RBM is used to optimize weights to the proposed DNN-RBM.

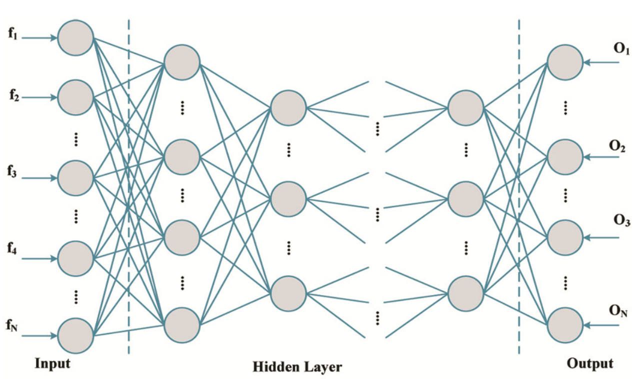

Fig. 1 — Structure of DNN

The present study involves three steps:

1) Pre-training
2) Fine-tuning stages
3) Optimization

Pre-Training

The purpose of the DNN is to get acquainted with a jumbled and conceptual representation of information in a hierarchical manner by passing the information via several change layers. The DNN's architecture has usually comprised three parts, including the input, hidden, and output layer, each of which includes multiple linked preparation units. The average design of the DNN is shown in Fig. 1. Each layer in the DNN employs a nonlinear modification on its input and produces a representation in its output. The vectorization is the DNNs' input, while the system's output is the choice vector. In this research work, the DNN is examined using various features like congestion window, throughput, propagation delay, RTT, number of packets sent and packet loss, etc.

Fine-Tuning Phase

There are two steps in this process discussed as follow:

Initialization: The weight parameters of the DNN are to be optimized. So, these weight parameters or solutions are initialized as follows:

$$S = \{s_1, s_2, \ldots.., s_N\} \qquad \ldots (1)$$

$$s_N = \{w_{xm}, w_{qx}, w_{pq}\}_N \qquad \ldots (2)$$

where, w_{xm} denotes the interconnection weight among the first hidden layer input feature.

w_{qx} denotes the interconnection weight among the $(n)^{th}$ hidden layer and the $(n-1)^{th}$ hidden layer and w_{pq} denotes the interconnection weight at the n^{th}

output layer and hidden layer having q^{th} and p^{th} nodes separately.

Fitness calculation: After the initialization of solutions, the fitness function is estimated to evaluate each solution.

The fitness function of this work is defined as follows:

$$Fit_N = Min\left(Error\right) \quad \ldots (3)$$

Using this fitness, each solution is evaluated. The solution with minimum fitness is the selected optimal solution.

Optimization with RBM

After two steps, the author approaches for RBM technique for optimization of given problem. As one of the profound algorithms RBM is used for classification, extraction, regression, and identification features.

It operates in two aspects:

1) The first is the hidden layer bias that enables the Restricted Boltzmann machine to do its forward bias activation function.

2) The second is the visible layer bias that helps the Restricted Boltzmann machine rebuild in the backward bias.

Restricted Boltzmann machine is a visual model with an undirected energy function and transforms it in a distribution of probability by taking the negative energy exponent and normalizing it. RBM creates S (visible layer) distribution which includes a certain amount of latent (hidden layer) variables, such that the distribution initially defines as an energy function.

The DNN has mainly trained two layers simultaneously and treated these two layers like RBM. The secret layer of an RBM is the input layer of the neighboring one in the network. The first RBM is trained and its output is then used as inputs for the next RBM. It is achieved before the production layer has been reached. The DNN can identify the underlying trends of the data after this training phase.

Termination criteria: This calculation stops its activity just on the off chance that it accomplishes the greatest number of emphasis and chooses the arrangement that has better fitness esteem, and is alluded to as the best weight parameters. When the better exercise is accomplished by the RBM calculation, they chose arrangement is applied to the DNN. The overall process of RBM based weight improvement is shown in Fig. 2.

Results and Discussion

The proposed approach DNN-RBM for congestion control is executed in the foundation of Python. The best way to determine the best method for accurately predicting data congestion through three different machine-learning techniques is given in Table 1.

From the experimental results shown in Table 1, we see that all other machine learning techniques were supervised by the Linear Regression. The explanation is that time-dependent data and methods of regression can be used to forecast time-dependent data. Precision, recall and accuracy are the metrics used for measuring prediction results. The summation of true positives and false positives is known as accuracy. The recall is defined by the sum of True Positive and False Negatives as True Positive. And precision is determined by summing all the true and false values as all true values.

Performance Metrics

The exhibition of our proposed method is assessed utilizing accompanying metrics. Execution metrics of

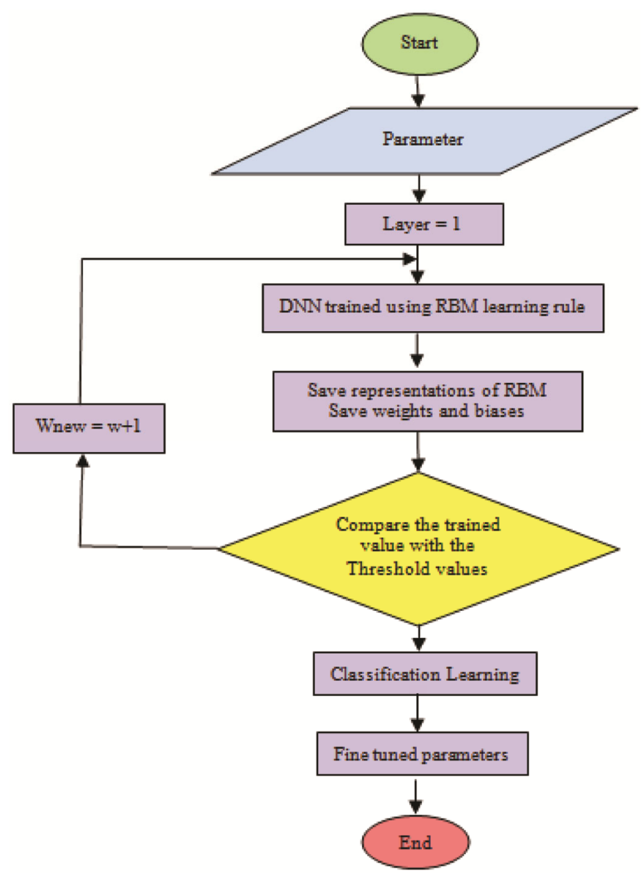

Fig. 2 — Flowchart of the RBM for the training of the DNN

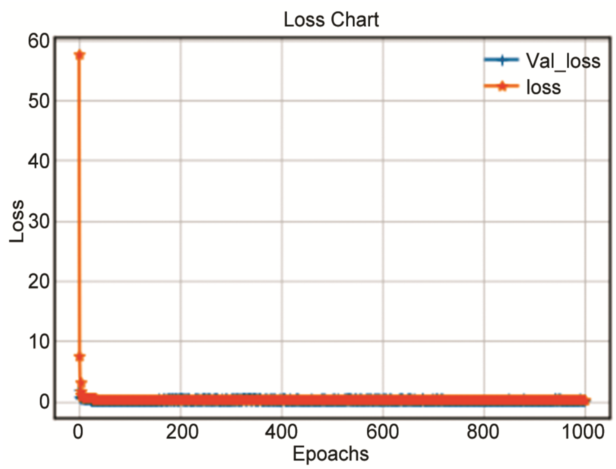

Fig. 3 — Loss chart for epoch=100 (DNN-RBM)

Table 1 — Results of Machine learning Techniques used

Techniques	Precision	Recall	Accuracy
Decision Tree	100	88.03	88.05
Random forest	100	79.80	79.81
Linear Regression	98.9	80.81	80.82
DNN-RBM	99.9	90.45	90.55

our proposed method DNN-RBM are contrasted and that of DNN and Logistic regression.

Packet Loss: The use of ACKs (accountability) on a sender can be measured; the protocol guarantees confidence.[8]

Also measurable is the sequence number on the receiver. As in[9], CTS (Clear to Send) packet loss may also be used as a congestive indicator.

Propagation Delay: When there is a very high transmission delay than transfer delay, the chances of losing the packet are high. That is also the time taken for a packet to get there.

Throughput: It is the amount of data that can be sent from the source to the target per second. The unit of this parameter is kb/s.

$$Throughput = \frac{Amountoftransmitteddata(kb)}{Transmittedtime(s)} \quad \dots (4)$$

Round-trip time (RTT): RTT is the total amount of time required to send the first packet, plus the time required to get the response packet.

Buffer Occupancy: Every node of the sensor has a buffer used to store packets before wireless transmission. A congestion warning is triggered if the buffer occupancy exceeds a threshold. A strong and easy indicator of congestion is the buffer threshold test.

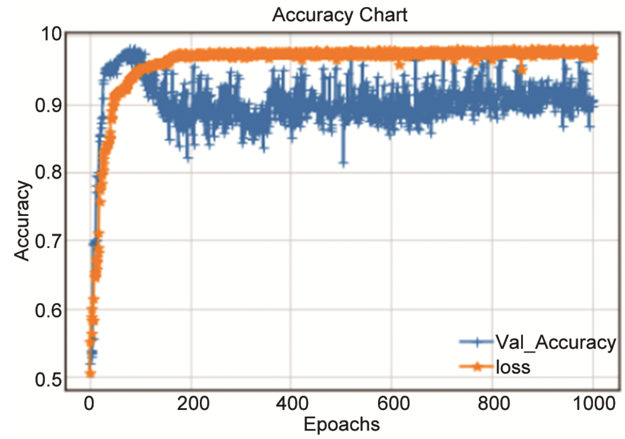

Fig. 4 — Accuracy chart for epoch = 100 (DNN-RBM)

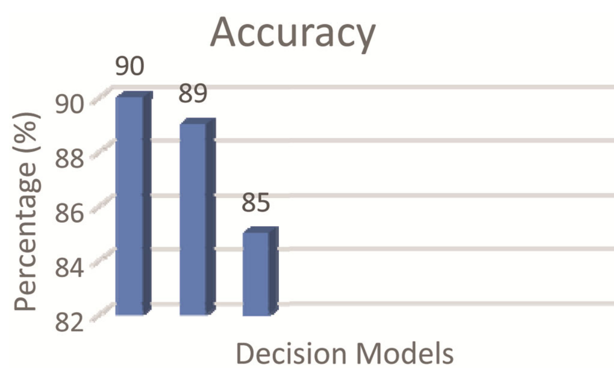

Fig. 5 — Accuracy of different decision models

Performance Analysis

The above Table 1 Shows the comparative results with different machine learning techniques. Due to the weight optimization of DNN using RBM, the prediction of congested packets is improved accurately. The results, therefore, demonstrate that the DNN-RBM is performed well, providing an overall accuracy of 99.9%, 90.4%, and 90.5%.

In Fig. 3 Loss chart for epoch count, 100 is displayed for the DNN-RBM training, and Fig. 4 Displays the accuracy achieved for 100 runs for the DNN-RBM model.

The Fig. 5 represents the Accuracy of Different models for which DNN-RBM model is with 90% accuracy, DNN-GA model is with 89% accuracy and ANN model is with 85% accuracy.

Conclusions

This paper have introduced the dependence of Restricted Boltzmann machine algorithm on deep neural network (DNN-RBM) for predicting congested packets in the WSN based IOT. The performance of the DNN has improved by optimizing the weight values using the RBM algorithm. In this proposed DNN-RBM, congestion window, throughput, propagation delay, RTT, number of packets sent, and packet loss are given as input. By utilizing these input factors, congestion in the nodes is predicted. The exhibition of the proposed DNN-RBM has been assessed regarding throughput, packet loss, RTT, and buffer occupancy. The Accuracy of Different models is predicted for which the DNN-RBM model is 90%, the DNN-GA model is 89% and the ANN model is 85%. Likewise, the presentation of the proposed DNN-RBM has contrasted that of the DNN-GA and DNN. As of now a robust algorithm is also required to automatically change the threshold values depending on the network type for future work congestion management schemes.

Conflict of Interest

The authors declare that there are no conflicts of interest.

References

1 Vermesan O, Friess P, Guillemin P, Gusmeroli S, Sundmaeker H, Bassi A, Jubert IS, Mazura M, Harrison M, Eisenhauer M & Doody P, Internet of things strategic research roadmap, *Internet Things-Global Technol Societ Trends*, **1** (2011) 9–52.

2 Dalipi F & Yayilgan S Y, Security and privacy considerations for iot application on smart grids: Survey and research challenges, in *IEEE 4th Int Conf Future Internet Things Cloud Workshops (FiCloudW)* 2016, 63–68.

3 Zhang D, Chen Z, Awad M K, Zhang N, Zhou H & Shen X S, Utility-optimal resource management and allocation algorithm for energy harvesting cognitive radio sensor networks, *IEEE J Select Areas Commun*, **34(12)** (2016) 3552–3565.

4 https://www.comptia.org/resources/sizing-up-the-internet-of-things (August 2015).

5 Ren J, Zhang Y, Zhang N, Zhang D & Shen X, Dynamic channel access to improve energy efficiency in cognitive radio sensor networks, *IEEE Trans Wirel Commun*, **15(5)** (2016) 3143–3156.

6 Deore R K, Sonawane V R & Satpute P H, Internet of Thing based home appliances control, in *Int Conf Comput Intel Commun Netw (CICN),* 2015, 898–902.

7 Maheshwari A & Yadav R K, Analysis of congestion control mechanism for IoT, in *10th Int Conf Cloud Comput*, *Data Sci & Eng (Confluence)*, 2020, 288–293.

8 Tezcan N & Wang W, ART: an asymmetric and reliable transport mechanism for wireless sensor networks, *Int J Sens Netw*, **2(3-4) (**2007**)** 188–200.

9 Akyildiz I F, Vuran M C & Akan O B, A cross-layer protocol for wireless sensor networks, in *40th Ann Conf Inf Sci Syst*, 2006, 1102–1107.