Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 583 – 597

doi:10.3906/elk-2001-145

Deep learning techniques of losses in data transmitted in wireless sensor networks

Mevlut ERSOY[1,*], **Bekir AKSOY**[2]
[1]Department of Computer Engineering, Faculty of Engineering Süleyman Demirel University, Isparta, Türkiye
[2]Department of Mechatronics Engineering, Faculty of Technology, Isparta University of Applied Sciences, Isparta, Türkiye

Received: 29.01.2020 • **Accepted/Published Online:** 10.08.2020 • **Final Version:** 30.03.2021

Abstract: Wireless sensor network (WSN) systems are frequently used today as a result of rapid technological developments. Wireless sensor networks, which form the basis of the Internet of Things (IoT), have a wide range of use in the world from education to health, and from military applications to home applications. It enables the data obtained from the sensors to be transferred between nodes with the help of end-to-end wireless protocols. In parallel with the increasing number of nodes in WSN, data traffic density also increases. Due to the limitations of the WSN network, lost packet rates also increase with increasing data traffic. In this study, a data set was created by examining the data transfers of different amounts of WSN nodes placed in different places. The effects of the number of sensors and the distance between them were evaluated from the data set. In this study, a data set was created by collecting the data from the sensor nodes placed at 1500m x 1500m intervals in the ns-3 discrete event emulator program. Today, with the rapid development of technology, deep learning methods which are one of the artificial intelligence methods, are also used in WSN. In this study, the loss rate in the transferred data packets was tried to be estimated with the highest accuracy by using deep belief network (DBN), recurrent neural network (RNN), and deep neural network (DNN) over the obtained dataset. Of these three deep learning methods, DNN deep learning method was found to accurately estimate the loss rate in the transferred data packets with an accuracy rate of 88.50%.

Key words: Wireless sensor networks (WSNs), deep learning, deep belief networks (DBNs), deep neural networks (DNNs), recurrent neural networks (RNNs)

1. Introduction

Wireless sensor networks (WSN), which form the basis of the Internet of Things (IoT) concept, have a wide range of work areas worldwide [1–5]. The interests of WSNs consist of indoor and outdoor environments. Many new applications are being developed in these areas [6–8]. These applications have enabled the data to be collected in real time from different environments to the created WSNs. So, people can remotely control their homes, get real-time information about the cities they visit, or follow the living creatures in their environment [9]. Typically, the WSN can consist of a processor, limited memory space, wireless radio transmitter/receiver, and several sensor structures. In WSN terminology, these structures are called as nodes. Nodes can be a few or a thousand in an environment. Nodes that have the ability to collect information from environments use gateways to transfer data to a particular server or cloud system. These gateways serve as the base station called

*Correspondence: mevlutersoy@sdu.edu.tr

sink node in the WSN infrastructure [7]. Excessive number of nodes in the environment can cause blockages in sink nodes. The problems that occur in transferring data to sink nodes cause the efficiency of battery usage to decrease and cause congestion and connection status checks to be done inefficiently. When these situations arise, the data must be predicted in advance to be sent to the central node, and real-time precautions must be taken according to these predicted results.

The heterogeneous structure of WSN nodes can cause data transmission times to change momentarily. Data density and variable data rates that occur due to the increase in the number of nodes may cause packets to be dropped in the sink nodes. This situation may cause obstruction in the operation of the system for a certain period of time [10, 11]. This can lead to the drop or retransmission of packets, leading to inefficient use of limited energies in WSNs. In addition, network performance decreases and sink nodes are unable to perform their required tasks. There are many variables that affect the performance of the network in WSNs[12]. Here are a few examples:

- Number of nodes in the network
- Distance between nodes
- Number of packets to be transferred
- Size of the packets
- Packet loss rate
- Transmission times of packets

These are the performance criteria that vary according to the request packets coming to WSN and are determined during the process.

In the literature, different studies have been conducted on deep learning methods used for WSNs [13]. Deep learning is a method of machine learning, which is a subbranch of artificial intelligence that began to spread in the 2000s [14]. Deep learning allows to estimate the outcome to be obtained with the given dataset and to train artificial intelligence according to these outcomes[1]. Multilayered neural networks were introduced by Hinton in 2006 to process data such as images, sound, text and to generate output. After its introduction bu Hinton, te networks were started to be used frequently in deep learning applications [15]. The reasons for the deep learning to become widespread nowadays are that the data required for training are at a sufficient level and there is an advanced infrastructure that can process this data [16]. For example, the DNN technique performs well in classification and segmentation processes [17]. However, in some cases, classic classification methods are inadequate. The accuracy rate was increased by using the quantum-inspired classifier method developed in 2018–2019 [18–21].

Deep learning generally consists of the input layer, three or more hidden layers, and the output layer [22]. As it consists of multiple layers, the learning process takes place much more successfully[2]. In deep learning, the learning process can take place as supervised, unsupervised, and semi-supervised [23]. In its architecture, the features learned in each layer are transferred to the next layer as input [24]. The use of deep learning is quite

[1]MEDIUM (2019). Derin Öğrenme (Deep Learning) Nedir ve Nasıl Çalışır? [online]. Website https://medium.com/@nyilmazsimsek/derin-C3B6C49Frenme-deep-learning-nedir-ve-nasC4B1l-C3A7alC4B1C59FC4B1r-2d7f5850782 [accessed 26 January 2020].

[2]BEYAZ (2019). Derin Öğrenme (Deep Learning) Nedir? [online]. Website https://www.beyaz.net/tr/yazilim/makaleler/derin-ogrenme-deep-learning-nedir.html [accessed 26 January 2020].

wide and with the help of artificial neural networks, it is used in fields such as object identification [25], speech identification [26] and natural language processing [27].

When the academic literature is examined it was seen that, Mendoza et al. performed an audio detection operation with an accuracy rate of 83.79% using WSN based on the convolutional neural network (CNN) deep learning algorithm [28]. In another study, DNN, CNN, and convolution preprocessing neural network(CPNN) based deep learning methods were used to increase the industrial WSN security, and CPNN algorithm was found to perform best in the authentication process in the sensor nodes [29]. Lee et al. improved the query process by processing data collected from four WSNs in real time using deep learning methods [30]. Wang et al. proposed a data fusion algorithm based on deep learning models to reduce energy consumption and prolong the life of WSNs [31]. In another study, the detection of rust on coffee leaves was carried out using WSN, remote sensing, and deep learning methods [32]. Turabieh et al. reported that Layered Recurrent Neural Network (L-RNN) deep learning algorithm works correctly with few or more examples to predict indoor user localization using fingerprints in WIFI networks [33]. In another study, experimental data collected from a WSN structure created for the fire detection system was verified using the Bayesian classification model [34]. In their study, Panda et al. conducted fault diagnosis by using deep learning techniques with the WSN [35].

In this study, the performance criteria of WSNs with variable nodes were estimated using deep learning techniques. In the simulation environment created for WSN, data related to certain performance criteria were collected. Thanks to the algorithm that has been tested, the WSN performance criteria have been determined in advance and the necessary information has been created to take measures in the operation of the network. The collected data were processed using deep belief network (DBN), deep neural network (DNN) and recurrent neural network (RNN) deep learning techniques, and the loss rate in the transmitted data packets was estimated.

2. Materials and methods

In this study, the number of nodes, the distance between nodes, the amount of transmitted packets, minimum, maximum and average delay times, and standard deviation values of these times received from the ns-3 discrete emulator program were determined as input parameters for DBN, DNN, and RNN deep learning techniques. Similarly, the loss rate in the transmitted data packets received from the ns-3 discrete event emulator program was determined as the output parameter. Coefficient of determination (R^2), mean absolute error (MAE), mean squared error (MSE), root mean squared error (RMSE) were used as performance evaluation criteria.

2.1. Materials

WSN are temporary (adhoc) networks consisting of a large number of sensor nodes to detect and control physical events [36]. Nodes can perform data collection via a mobile or fixed system [37, 38]. These nodes can transfer data to each other via wireless media. WSN architecture (Figure 1) consists of sensor nodes that control or detect the environment and sink nodes that are used to transfer data from sensor nodes to central points [39]. According to the figure, the sensing data by the sensor nodes are transferred to the sink nodes thanks to the neighboring nodes. Sink nodes, which are called base stations, are the nodes that are directly connected to the storage unit where the sensed data is stored. In this way, the sensed data in the end nodes are saved to the storage unit via the sink node. Data loss in neighboring nodes may occur during this data transfer due to the location of the sensors, limited battery conditions, rules of routing algorithms, etc.

Data transfer process in WSN consists of 4 stages [40]. These stages are data collection, data processing,

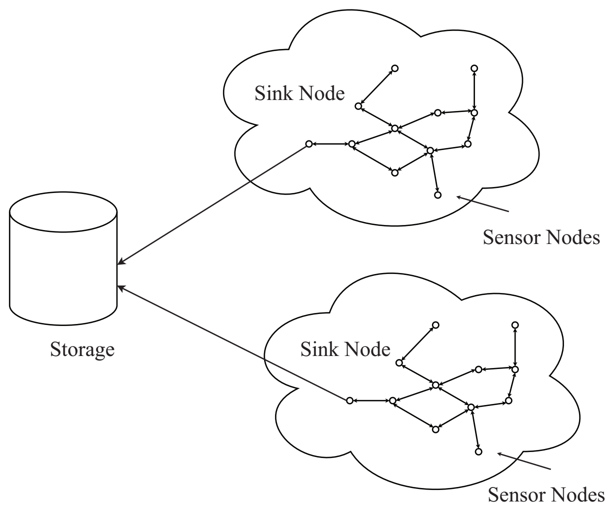

Figure 1. Transfer of sensed data from sensor nodes to storage unit via sink node in WSN.

data packaging, and data transfer. It is important to detect blockages in the data transfer process of the WSN networks to determine the current path of the routing algorithm and to transfer the data without loss in energy efficient data transfer [41]. In the proposed study, it is aimed to give preliminary information to the routing algorithms by predicting packet losses with deep learning techniques.

2.1.1. Deep belief networks (DBNs)

Deep belief networks (DBN) are one of the deep learning methods based on the restricted boltzmann machine (RBM) algorithm proposed by Geoffrey Hinton[3] [42]. RBMs consist of two layers: input (visible) and hidden layer. RBM is a model in which there is a full connection between its layers, but there is no inner connection inside its layers between the nodes [43, 44]. Each hidden layer in the network is connected to the input layer of the next RBM layer [45]. Thus, the hidden layer of the previous network acts as the visible (input) layer of the next network. These two layers have non-directional connections, which allows DBN to be used in supervised and unsupervised learning practices [46, 47]. In the DBN, the greedy layer-by-layer training method is used to learn the most efficient weights from top to bottom [48]. The working logic of these networks is based on making the closest selection to the result[4]. Using the input (visible) layer, which is the lower layer of the DBN as the training set, individual activation possibilities for the hidden layer are calculated. Afterwards, hidden layer and input layers are updated as shown in Figure 2 by using the positive and negative phases [49]. The first layers (V_1, V_2 and V_3) seen in Figure 2 (a) and (b), are represented input layers of DBN networks. Hidden units (h) represent features that capture the correlations present in the data. In n-1 layered DBN network, each two layers are connected by a matrix of symmetrical weights W^n. A neighboring hidden units are connected to each other by weight value (W_{11}, W_{12}, etc). Weights for the second RBM is the transpose of the weights for the first RBM (ex: W_1=W_0^T). All the hidden units of the each hidden layers are updated in parallel. This is

[3]PATHMIND (n.d.). Deep-Belief Networks [online]. Website https://pathmind.com/wiki/deep-belief-network [accessed 25 January 2020].
[4]BILGISAYAR KAVRAMLARI (2008). Açgözlü Yaklaşımı (Greedy Approach) [online]. Website http://bilgisayarkavramlari.sadievrenseker.com/2008/03/24/acgozlu-yaklasimi-greedy-approach/ [accessed 26 January 2020].

called as the positive phase. To reconstruct the visible units DBN network use negative phase which is similar technique to positive phase. The process of calculation the positive phase, negative phase, and update all the associated weights will be repeated till we get required threshold values[5].

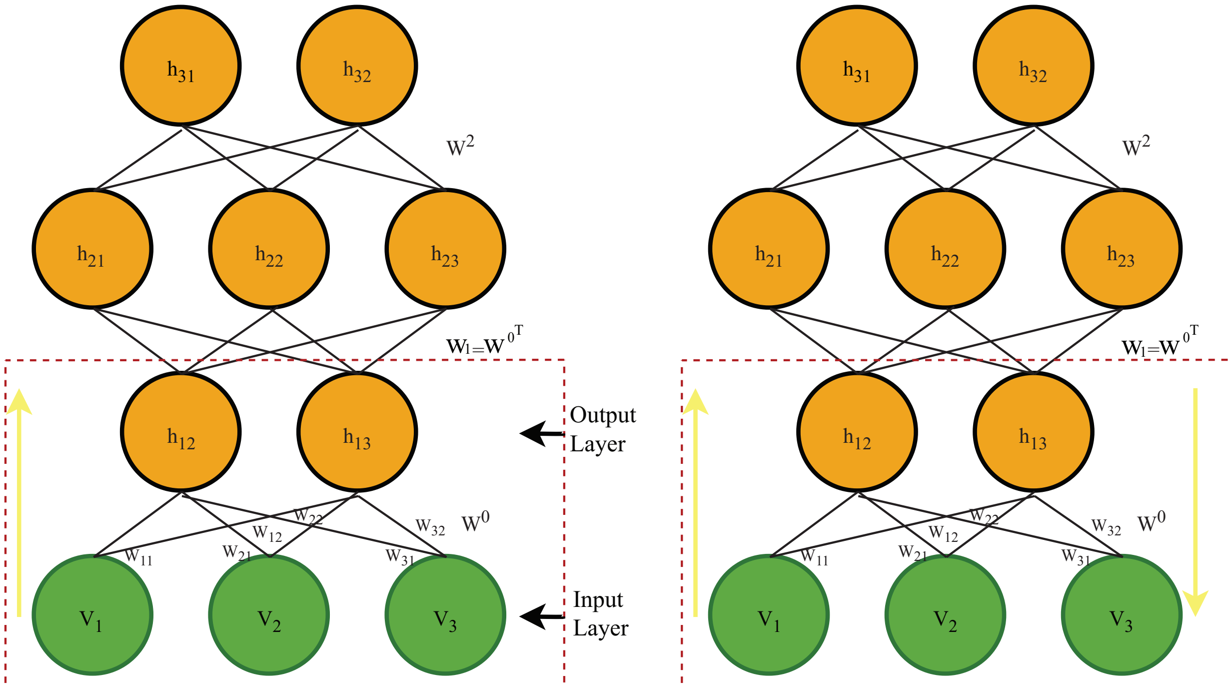

Figure 2. Architecture of DBN (a) positive (b) negative phase calculation.

2.1.2. Recurrent neural networks (RNNs)

Feed forward neural networks are a network model that consists of three layers: input layer, hidden layer and output layer, where the data flow is from the input layer to the output layer[6]. In these traditional neural networks, all inputs and outputs are independent of each other. Recurrent Neural Networks (RNN) are different from feed forward neural networks in terms of using the samples that were previously used as inputs in the system as input again [43, 50]. RNN has connected input and output as well as a memory to collect information about calculated data[7]. Accordingly, an interpretation can be made by using the information used in the past for a newly learned situation[8]. In other words, RNN has feedback links so that the outputs can be reused in the system [51]. A simple RNN structure and unfolded version of the structure can be seen in Figure 3. In the figure, "h" represents cells (nodes) in the neural network, "X" represents the input value of the network, and

[5]MEDIUM (2018). Deep Learning — Deep Belief Network (DBN) [online]. Website https://medium.com/datadriveninvestor/deep-learning-deep-belief-network-dbn-ab715b5b8afc (accessed 25 January 2020).

[6]BILGISAYAR KAVRAMLARI (2008). İleri Beslemeli Ağlar (Feedforward Neural Networks) [online]. Website http://bilgisayarkavramlari.sadievrenseker.com/2008/11/02/ileri-beslemeli-aglar-feedforward-neural-networks/ [accessed 25 November 2020].

[7]DEVHUNTER (2018). Tekrarlayan Sinir Ağları Eğitimi, Bölüm 1 – RNN'lere [online]. Website https://devhunteryz.wordpress.com/2018/07/09/tekrarlayan-sinir-aglari-egitimi-bolum-1-rnnlere-giris/ [accessed 25 November 2019].

[8]ELITCENKALP (2018). Recurrent Neural Networks [online]. Website http://elitcenkalp.blogspot.com/2018/04/recurrent-neural-network.html (accessed 25 November 2019).

"o" represents the output value of the network. If we unfold this loop, the standard RNN can be considered to copy the same structure multiple times, and the state h of each copy is taken as an input to its successor. The input layer, hidden layer, and output layer at time t is shown as $X^{(t)}$, $h^{(t)}$ and $o^{(t)}$, respectively. Where U,V, and W are the weighting matrices of the input-to-hidden connection, hidden-to-output connection, and hidden-to-hidden connection, respectively [52]. In RNNs, the connections between the nodes are performed with a redirected loop and sequential events can be interpreted according to each other [44, 53].

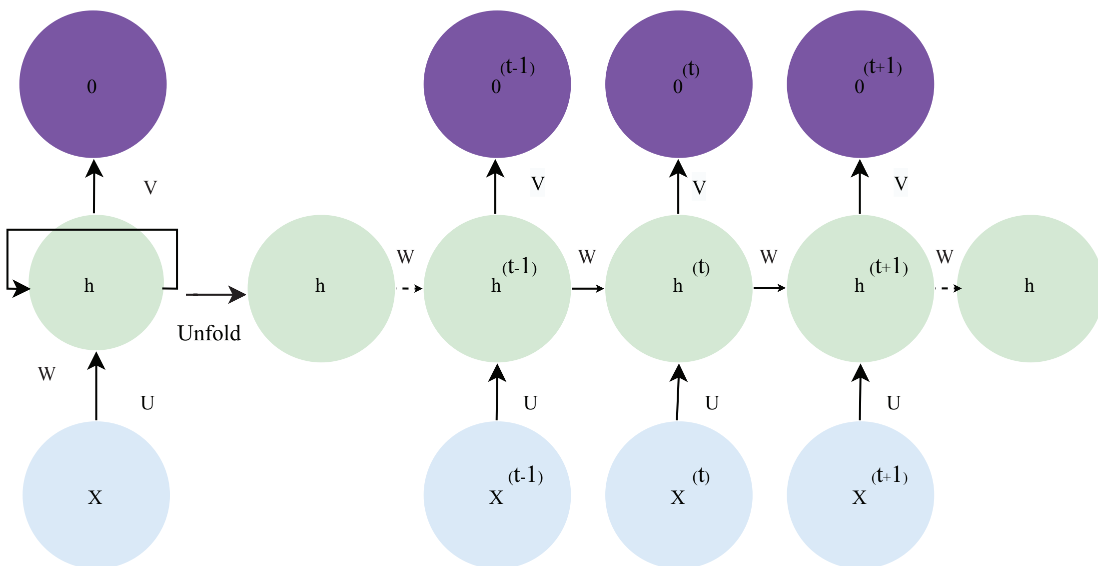

Figure 3. Architecture of standart RNN and unfolded RNN[52].

2.1.3. Deep neural networks (DNNs)

DNN is an artificial intelligence algorithm, also called as convolutional network [49, 54]. DNNs are the most known representatives of deep learning and have more than one hidden layer in the network [55]. DNNs are defined as a more extended type of artificial neural networks with two or more hidden layer numbers [56]. DNNs are networks that can understand and learn the properties of an object and make classifications using assumptions made in accordance with these properties [55]. In short, DNSs are used to find the relationship between input and output[9].

2.1.4. Performance evaluation criteria

In the study, R^2 [57], RMSE [58], MSE [59], and MAE [60] performance evaluation criteria were used to evaluate the prediction model created using DBN. Mathematical expressions of R^2, MAE, MSE, and RMSE performance criteria are given between Eq (1) and Eq (4).

$$\mathbb{R}^2 = \left(1 - \frac{\sum_{t=1}^{N} (A_t - F_t)^2}{\sum_{t=1}^{N} (A_t - M)^2}\right) x100 \tag{1}$$

[9]PYTHON DUNYASI (2019). Derin Öğrenme (Deep Learning)Nedir? [online]. Website https://pythondunyasi.com/derin-ogrenme-deep-learningnedir/ [accessed 27 January 2020].

$$MAE = \frac{1}{n}\sum_{t=1}^{N}|A_t - F_t| \tag{2}$$

$$MSE = \frac{1}{n}\sum_{t=1}^{N}(A_t - F_t)^2 \tag{3}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{N}(A_t - F_t)^2} \tag{4}$$

In the equations, N is the number of samples, A_t is the real value, and F_t is the estimated value [61, 62].

2.2. Methods

The data in this study were obtained from the ns-3 discrete-event emulator program. The sensor nodes are placed randomly in an area of 1500m × 1500m. Figure 4 shows an example network structure created with ns-3. In this network structure, the nodes are distributed to the test bed using uniform random distribution. By distributing the sensors randomly to the test bed, a heterogeneous data set is obtained. In experiments, 10% of total nodes are determined as sink nodes.

As seen in Table 1, the amount of sensors used in the study has been increased from 10 to 100 by tens. The distance between the sensor nodes ranges from 10m to 100m. The last formed node among the sensor nodes is considered to be the sink node. According to the data requests transmitted to the nodes, the sensor nodes at the end provided data transfer to the sink nodes. The number of packets transmitted to the nodes has changed over time. The number of packets transmitted to the sink node, the lost packet percentages, the minimum, maximum, and average delay times were obtained in the WSN environment created according to these changes. In addition, standard deviations of the delay times of 100 packets transmitted during each simulation period were examined. A data set was created using these analyzed data. Simulation parameters can be seen in Table 1.

The sizes of the data obtained from the sensor nodes are determined between 56 bytes and 64 bytes. The ad hoc on-demand distance vector routing (AODV) algorithm has been selected to be used for data transfer. Transfer of the data to the sink node was carried out using neighboring nodes. The data transmitted to the sink nodes are transferred to a storage unit. The simulation software was run for 10 s of simulation time according to different node numbers. At the end of the simulation, the package sizes obtained from the sensor nodes, the package dimensions transferred to the storage unit, the average latency time, the maximum latency time, and throughput values were obtained. According to the 42765 data obtained, the number of lost packets was calculated.

In this study, the rate of loss in the transmitted data packets was estimated by applying DBN, RNN, and DNN deep learning techniques on the data collected from WSNs. The work flow diagram of this study can be seen in Figure 5.

As can be seen in Figure 5, the collected data was first subjected to a data preprocessing processes such as scaling and cleaning of recurring data. After the data preprocessing phase, the data were randomly allocated for 80% training and 20% testing. Regression operation was performed with DBN, DNN, and RNN using numpy, pandas, random, matplot, scikit-learn and tensorflow libraries in Python programming language. First of all,

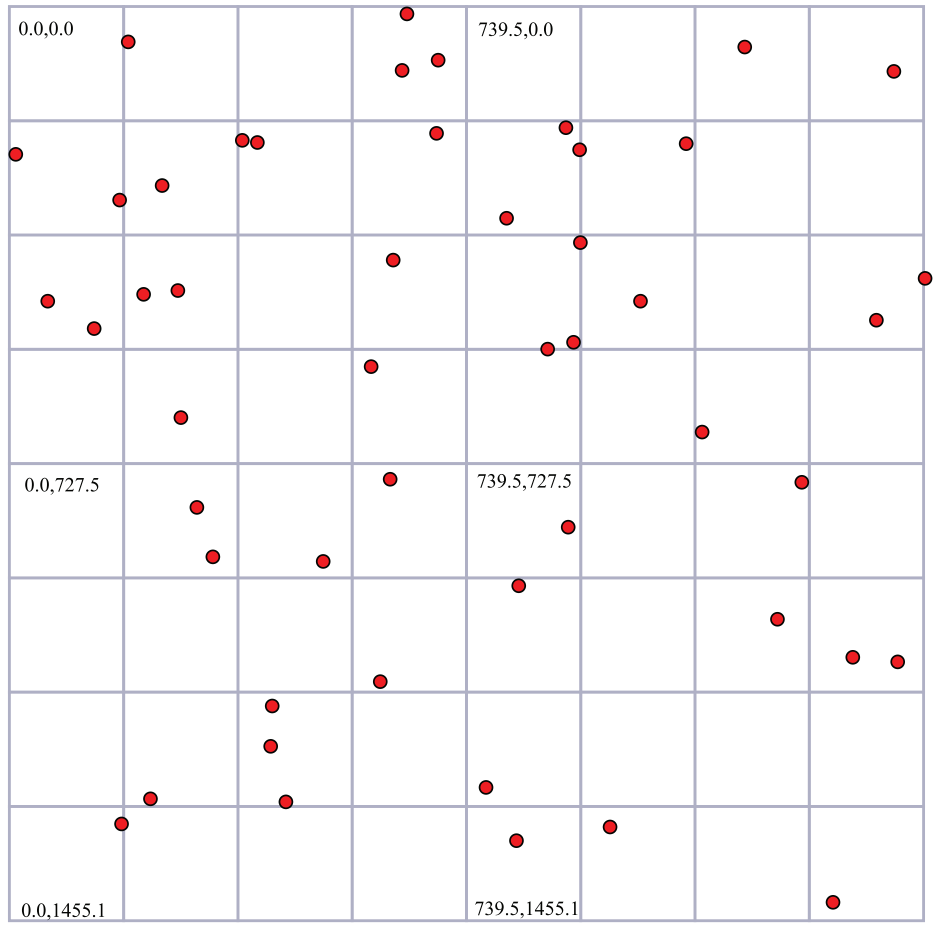

Figure 4. Distribution of sensor nodes consisting of 50 nodes in the testbed.

Table 1. The values of the tested WSN parameters used in simulation.

Parameters	Values
Node number	10, 20,30,40,50,60,70,80,90,100
Distance between nodes	10, 20,30,40,50,60,70,80,90,100
Area of the network	1500mx1500m
Simulation time	10 sn
Number of packets delivered	10, 20,30,40,50,60,70,80,90,100
Delivered packet size	56-64 byte
Transmit power	7,5dB
Routing Algorithms Used	Ad-Hoc On–Demand Distance Vector Routing (AODV)
Data Rate	11 Mbps
Access method	OFDMA
Process frequency	2.4 GHz

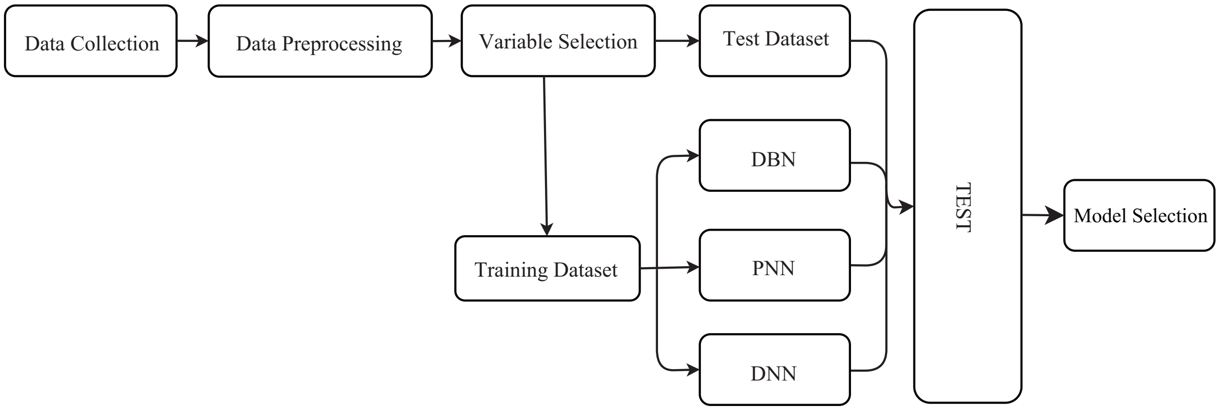

Figure 5. General structure of the study.

in supervised DBN regression model, we used 100 variable at Hidden layers structure. In the trained model, the batch size value is 16, Relu is selected as the activation function and the epoch number is taken as 50. For the DNN model, which is used as the second model, we used 5 fully connected hidden layers each of which consisting of 16 neurons. In the trained model, the batch size value is 16, rectified linear nit (ReLU) is selected as the activation function and the epoch number is taken as 50. Finally, in RNN model the first layer is 10 Long Short-Term Memory (LSTM), and the last two layers are 6 and 1 is fully connected layer. In the trained model, the batch size value is 16, ReLU is selected as the activation function and the epoch number is taken as 50. The results obtained were evaluated according to R^2, MAE, MSE, and RMSE performance evaluation criteria.

3. Research findings

Firstly, the loss rate in the data packets, which was first transmitted using the DBN deep learning technique with the software prepared in the Python programming language, was estimated using the regression technique. For the DBN model, the supervised DBN regression class in the DBN algorithm in the tensorflow library in the Python programming language was used. Hidden layers structure with 100 variables, learning rate value of 0.01, and ReLU activation function were used for supervised DBN regression class. The RNN model used in the study consists of three layers, the first layer of which is 10 long short-term memory (LSTM), and the last two layers fully connected layers. Finally, the DNN regressor of the tensorflow library in the Python programming language was used for the DBN method used in the study. 5 fully connected hidden layer parameters consisting of 16 neurons were used in DNN regressor.The estimation model obtained was evaluated according to R^2, MAE, MSE, and RMSE performance evaluation criteria and the results are presented in Table 2.

Table 2. Results obtained from DBN method according to performance evaluation criteria.

Deep learning method	R^2	MAE	MSE	RMSE
DBN	61.72	0.197	0.087	0.295

As seen in Table 2, the accuracy of the results obtained by using DBN to estimate the loss rate in the transmitted data packets was found to be low. The RNN model used in the study consists of three layers, the first layer of which is 10 long short-term memory (LSTM), and the last two layers are fully connected layers with 6 and 1 LSTM. RNN technique was used as an alternative solution since the obtained result was not considered sufficient. The loss rate in the data packets transmitted using RNN technique was estimated using the regression technique. The estimation model obtained was evaluated according to R^2, MAE, MSE, and RMSE performance evaluation criteria and the results are presented in Table 3.

Table 3. Results obtained from RNN method according to performance evaluation criteria.

Deep learning method	R^2	MAE	MSE	RMSE
RNN	66.31	0.195	0.077	0.277

As seen in Table 3, the accuracy of the results obtained by using RNN to estimate the loss rate in the transmitted data packets was found to be low. Finally, the DNN regressor of the tensorflow library in the Python programming language was used for the DBN method used in the study. 5 fully connected hidden layer parameters consisting of 16 neurons were used in DNN regressor. DNN technique was used as an alternative solution since the obtained result was not considered sufficient. The loss rate in the data packets transmitted using DNN technique was estimated using the regression technique. The estimation model obtained was evaluated according to R^2, MAE, MSE, and RMSE performance evaluation criteria and the results are presented in Table 4.

Table 4. Results obtained from DNN method according to performance evaluation criteria.

Deep learning method	R^2	MAE	MSE	RMSE
DNN	88.50	0.11	0.035	0.167

The performance evaluation criteria result of the models obtained with three different deep learning techniques used in the study are given in Table 5.

Table 5. Results obtained from DBN, RNN, and DNN methods according to performance evaluation criteria.

Deep learning method	R^2	MAE	MSE	RMSE
DBN	61.72	0.197	0.087	0.295
RNN	66.31	0.195	0.077	0.277
DNN	88.50	0.11	0.035	0.167

When the graphics in Figure 6 are analyzed, it is seen that there are differences between the real values and the values predicted by the models using DBN and RNN deep learning methods. The graphics show the performance of the regression task. In DNN model the data points (red) lie along the blue curve (y = x) which shows that the real true and predicted values are close. The more points are aligned along the y= x line, the better the prediction. The R^2 score (0.885) is close to the best possible score of 1.0. The RMSE is also small (0.167) which also tells us that the predicted targets are close to the true targets. Therefore, it is seen that DBN and RNN deep learning methods are not suitable for the data set obtained from WSN in this study. However, when the graphic belonging to the DNN model is examined, it is seen that the real values and the values predicted by the model are close to each other. For this reason, it is seen that the DNN method, which is one of the three different deep learning methods used in this study, is suitable for the data set obtained from WSN.

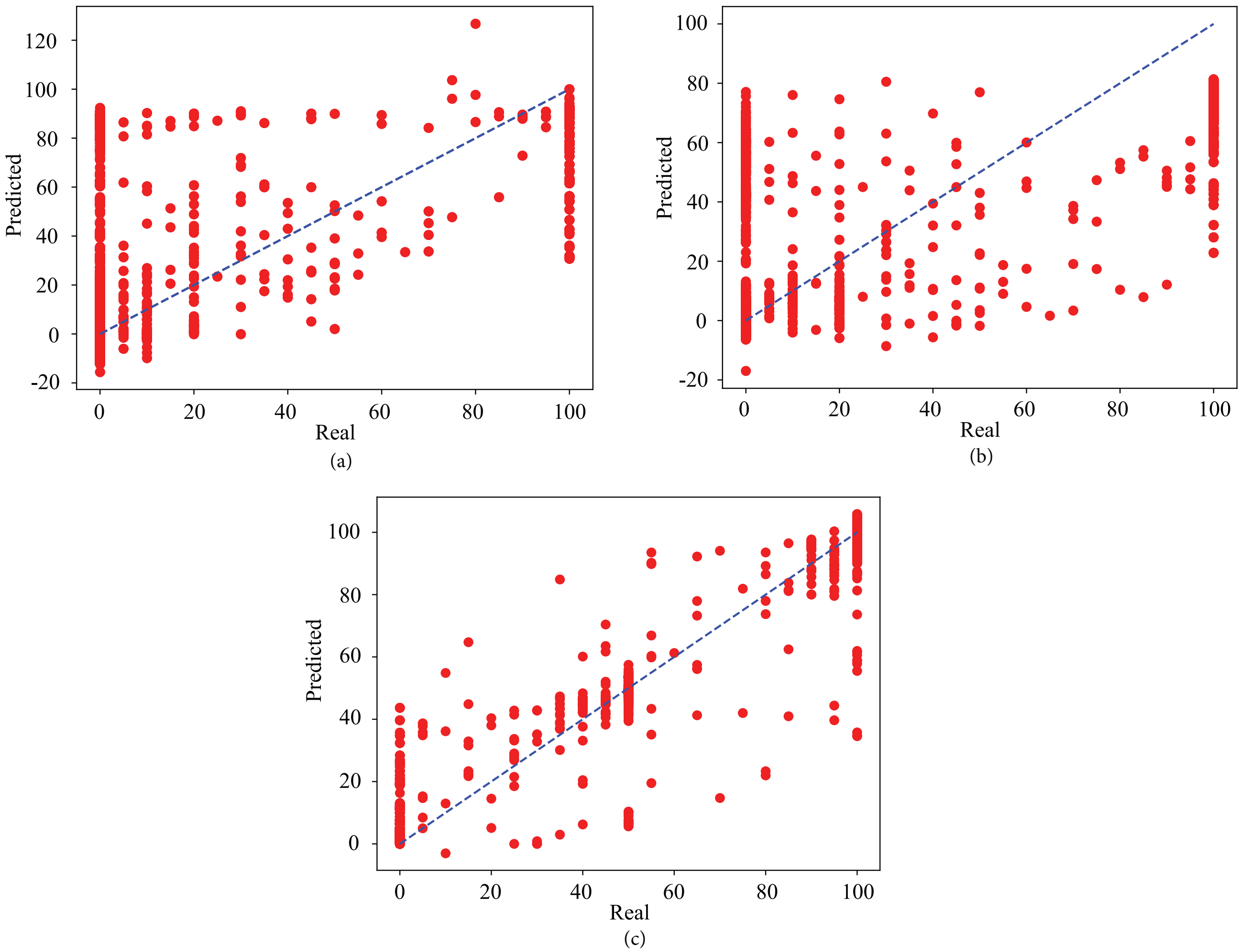

Figure 6. Regressions curve obtained for the deep learning network at the testing stages a) DBN (b) RNN (c) DNN.

When the performance of three different deep learning methods used in the study were examined, it was seen that the DNN model was more successful. The reason of DNN model is more successful model on our data; in training phase data flows from the input layer to the output layer without going backward and the links between the layers are one way which is in the forward direction. Also, the outputs are obtained by supervised learning with datasets of some information based on 'what we want' through back propagation. In other hand DBNs have no intra-layer or between unit connections among each layer and depend heavily on initialization and can be computationally intractable without effective pre-training. They are a class of artificial neural network where connections between nodes form a directed graph along a sequence like features links from a layer to previous layers, allowing information to flow back into the previous parts of the network thus each model in the layers depends on past events, allowing information to persist.

4. Results

The AODV routing algorithm system, which was created with the Ns-3 discrete emulator program, has a 2.4 GHz operating frequency and 11 Mbps data rate and the nodes are randomly placed in an area of 1500m x

1500m. For the prepared system, the number of nodes collected, the distance between nodes, the number of the packet transmitted, the minimum, maximum, and average delay times, and the standard deviation of the time were taken as input values. The loss rate in the transmitted data packet was taken as the output value. DBN, RNN, and DNN deep learning methods were used on the obtained data set and the following results were found.

- Firstly, DBN deep learning method was applied on the data set and the loss rate in the transmitted data packets was determined with 61.72% accuracy according to R^2 performance evaluation criteria.
- Secondly, RNN deep learning method was used on the data set and the loss rate in the transmitted data packets was determined with 66.31% accuracy according to R^2 performance evaluation criteria.
- Finally, DNN deep learning method was used on the data set and the loss rate in the transmitted data packets was determined with 88.50% accuracy according to R^2 performance evaluation criteria.

When the results obtained from deep learning methods were examined, it was determined that the best result was obtained with the DNN algorithm. The low accuracy rate obtained from DBN and RNN methods means that this data set is not suitable for DBN and RNN methods. Accordingly, the workload in the nodes is reduced in order to determine the number of sensors and the distance between the sensors required for the successful data transmission. It has been shown that the workload required for these calculations can be achieved using deep learning algorithms. Thus, less energy consumption of WSNs with less work load has been provided to increase the data transmission success rates. In future studies, higher accuracy rates are hoped to be obtained by using different artificial algorithms or hybrid algorithms.

References

[1] Vikram N, Harish KS, Nihaal MS, Umesh R, Shetty A et al. A low cost home automation system using wi-fi based wireless sensor network incorporating internet of things (IoT), In: 7th IEEE International Advanced Computing Conference; Hyderabad, Telangana, India; 2017. pp. 174-178.

[2] Kocakulak M, Butun I. An overview of wireless sensor networks towards internet of things, In: 2017 IEEE 7th Annual Computing and Communication Workshop and Conference; Las Vegas, NV, USA; 2017. pp. 1-6.

[3] Bera S, Misra S, Roy SK, Obaidat MS. Soft-WSN: Software-defined WSN management system for IoT applications. IEEE Systems Journal 2016; 12 (3): 2074-2081. doi: 10.1109/JSYST.2016.2615761.

[4] Wu F, Xu L, Kumari S, Li X, Shen J et al. An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment. Journal of Network and Computer Applications 2017; 89: 72-85. doi: 10.1016/j.jnca.2016.12.008.

[5] Li X, Niu J, Kumari S, Wu F, Sangaiah AK et al. A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments. Journal of Network and Computer Applications 2018; 103; 194-204. doi: 10.1016/j.jnca.2017.07.001.

[6] Ndiaye M, Hancke GP, Abu-Mahfouz AM. Software defined networking for improved wireless sensor network management: A survey. Sensors 2017; 17 (5): 1-32. doi: 10.3390/s17051031

[7] Sohrabi K, Gao J, Ailawadhi V, Pottie GJ. Protocols for self-organization of a wireless sensor network. IEEE Personal Communications 2000; 7 (5): 16-27. doi: 10.1017/CBO9781107415324.004.

[8] Yick J, Mukherjee B, Ghosal D. Wireless sensor network survey. Computer Networks 2008; 52 (12): 2292-2330. doi: 10.1016/j.comnet.2008.04.002.

[9] Qian, J, Tiwari P, Gochhayat SP, Pandey HM. A noble double dictionary based ECG compression technique for IoTH. IEEE Internet of Things Journal 2020; 4662 (c): 1–1. doi: 10.1109/jiot.2020.2974678

[10] Muñoz R, Vilalta R, Yoshikane, N, Casellas R, Martínez R et al. IoT-aware multi-layer transport SDN and cloud architecture for traffic congestion avoidance through dynamic distribution of IoT analytics. In: 2017 European Conference on Optical Communication (ECOC); Gothenburg, Sweden; 2017. pp. 1-3.

[11] Ancillotti E, Bruno R. BDP-CoAP: Leveraging bandwidth-delay product for congestion control in CoAP. In: 2019 IEEE 5th World Forum on Internet of Things (WF-IoT); Limerick, Ireland; 2019.pp. 656-661.

[12] Toprasert T, Lilakiataskun W. TCP congestion control with MDP algorithm for IoT over heterogeneous network. In: 2017 17th International Symposium on Communications and Information Technologies (ISCIT); Cairns, Queensland, Australia; 2017; pp. 1-5.

[13] Amanullah MA, Habeeb RAA, Nasaruddin FH, Gani A, Ahmed E et al. Deep learning and big data technologies for IoT security. Computer Communications 2020; 151: 495-517. doi: 10.1016/j.comcom.2020.01.016

[14] Seyitoğlu Z. Changing of consumer experience in digital public relations in Turkey: Chatbot applications. MSc, İstanbul Kültür University, Bakırköy, Istanbul, Turkey, 2019 (in Turkish).

[15] Koç E, Çalışkan S, Yazıcıoğlu SA, Demirci U, Kuş Z. Artificial neural networks, word vectors and deep learning applications. MSc, Fatih Sultan Mehmet Vakıf University, Fatih, İstanbul, Turkey, 2018 (in Turkish).

[16] Özkan İ, Ülker E. Deep learning and deep learning models used in image analysis. Gaziosmanpaşa Bilimsel Araştırma Dergisi 2017; 6 (3): 85-104 (in Turkish).

[17] Mallick PK, Ryu SH, Satapathy SK, Mishra, S, Nguyen GN et al. Brain MRI image classification for cancer detection using deep wavelet autoencoder-based deep neural network. IEEE Access 2019; 7: 46278-46287.

[18] Tiwari P, Melucci M. Towards a quantum-inspired binary classifier. IEEE Access 2109; 7: 42354-42372.

[19] Tiwari P, Melucci M. Towards a quantum-inspired framework for binary classification. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management; Torino, Italy; 2018. pp. 1815-1818.

[20] Tiwari P, Melucci M. Binary classifier inspired by quantum theory. In: Proceedings of the AAAI Conference on Artificial Intelligence; Honolulu, Hawaii, USA; 2019. pp. 10051-10052.

[21] Tiwari P, Melucci M. Multi-class classification model inspired by quantum detection theory. ArXiv preprint arXiv 2018; 1810.04491.

[22] Uyulan Ç, Ergüzel TT, Tarhan N. The use of deep learning algorithms on EEG based signal analysis. The Journal Of Neurobehavioral Sciences 2019; 6 (2): 108-124. doi: 10.5455/JNBS.1553607558

[23] Çelenli Hİ. Application of paragraph vectors to news and tweet data. In: 26th Signal Processing and Communications Applications Conference (SIU); Izmir, Turkey; 2018. pp. 1-4.

[24] Işık G, Artuner H. Recognition of radio signals with deep learning neural networks. In: 24th Signal Processing and Communication Application Conference (SIU); Zonguldak, Turkey; 2016. pp. 837-840.

[25] Daş R, Polat B, Tuna G. Recognition and tracking of objects in pictures and videos using deep learning. Fırat Üniversitesi Mühendislik Bilimleri Dergisi 2019; 31 (2): 571-581 (in Turkish). doi: 10.35234/fumbd.608778

[26] Işık G. (2019). Identification of Turkish dialects using deep learning techniques.PhD, Hacettepe University, Sıhhiye, Ankara, Turkey, 2019 (in Turkish).

[27] Nergiz G, Safali Y, Avaroğlu E, Erdoğan S. Classification of Turkish news content by deep learning based LSTM using Fasttext model. In: International Artificial Intelligence and Data Processing Symposium (IDAP); Malatya, Turkey; 2019. pp. 1-6 (in Turkish).

[28] Mendoza JM, Tan V, Fuentes V, Perez G, Tiglao NM. Audio event detection using wireless sensor networks based on deep learning. In: International Wireless Internet Conference; Taipei, Taiwan; 2018. pp. 105-115. doi: 10.1007/978-3-030-06158-6-11

[29] Liao RF, Wen H, Wu J, Pan F, Xu A et al. Deep-learning-based physical layer authentication for industrial wireless sensor networks. Sensors 2019; 19 (11): 1-17. doi: 10.3390/s19112440

[30] Lee KS, Lee SR, Kim Y, Lee CG. Deep learning–based real-time query processing for wireless sensor network. International Journal of Distributed Sensor Networks 2017; 13 (5): 1- 10. doi: 10.1177/1550147717707896

[31] Wang L, Xia K. Data fusion algorithms for wireless sensor networks based on deep learning model. In: 3rd International Conference on High Performance Compilation, Computing and Communications Conference; Xi'an, China; 2019. pp. 155-158. doi: 10.1145/3318265.3318297

[32] Velásquez D, Sánchez A, Sarmiento S, Toro M, Maiza M et al. A method for detecting coffee leaf rust through wireless sensor networks, remote sensing, and deep learning: Case study of the Caturra Variety in Colombia. Applied Sciences 2020; 10 (2): 697. doi: 10.3390/app10020697

[33] Turabieh H, Sheta A. Cascaded layered recurrent neural network for indoor localization in wireless sensor networks. In: IEEE 2019 2nd International Conference on new Trends in Computing Sciences; Amman, Jordan; 2019. pp. 1-6

[34] Khalid W, Sattar A, Qureshi MA, Amin A, Malik MA et al. A smart wireless sensor network node for fire detection. Turkish Journal of Electrical Engineering & Computer Sciences 2019; 27(4): 2541-2556. doi:10.3906/elk-1812-81

[35] Panda M, Gouda BS, Panigrahi T. Fault Diagnosis in Wireless Sensor Networks Using a Neural Network Constructed by Deep Learning Technique. In: De D, Mukherjee A, Kumar Das S, Dey N (editors). Nature Inspired Computing for Wireless Sensor Networks. Singapore: Springer Nature Singapore Pte Ltd, 2020, pp. 77-101.

[36] Sergiou C, Vassiliou V, Paphitis A. Congestion control in Wireless Sensor Networks through dynamic alternative path selection. Computer Networks 2014; 75: 226–238. doi: 10.1016/j.comnet.2014.10.007

[37] Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor networks: A survey. Computer Networks 2002; 38 (4): 393–422. doi: 10.1016/S1389-1286(01)00302-4

[38] Solaiman BF, Sheta A. Energy optimization in wireless sensor networks using a hybrid K-means PSO clustering algorithm. Turkish Journal of Electrical Engineering and Computer Sciences 2016; 24 (4): 2679–2695. doi: 10.3906/elk-1403-293

[39] Abbasi M, Rafiee M, Khosravi MR. Investigating the efficiency of multi-threading application programming interfaces for parallel packet classification in wireless sensor networks. Turkish Journal of Electrical Engineering & Computer Sciences 2020; 28 (3): 1699-1715 doi: 10.3906/elk-1910-168

[40] Gochhayat SP, Kaliyar P, Conti M, Tiwari P, Prasath VBS et al. LISA: Lightweight context-aware IoT service architecture. Journal of Cleaner Production 2019; 212: 1345–1356. doi: 10.1016/j.jclepro.2018.12.096

[41] Görkemli B, Al-Dulaimi Z. On the performance of quick artificial bee colony algorithm for dynamic deployment of wireless sensor networks. Turkish Journal of Electrical Engineering and Computer Sciences 2019; 27 (6): 4038–4054. doi: 10.3906/ELK-1902-189

[42] Diao W, Sun X, Zheng X, Dou F, Wang H, et al. Efficient saliency-based object detection in remote sensing images using deep belief networks. IEEE Geoscience and Remote Sensing Letters 2016; 13 (2): 137-141. doi: 10.1109/LGRS.2015.2498644

[43] Şeker A, Diri B, Balık HH. A review about deep learning methods and applications. Gazi Mühendislik Bilimleri Dergisi (GMBD) 2017; 3 (3): 47-64.

[44] Gündüz G, Cedimoğlu İH. Gender estimation with image by using deep learning algorithms. Sakarya University Journal of Computer and Information Sciences 2009; 2 (1): 9-17. doi: 10.35377/saucis.02.01.517930

[45] Küçük D, Arıcı N. A literature study on deep learning applications in natural language processing. Uluslararası Yönetim Bilişim Sistemleri ve Bilgisayar Bilimleri Dergisi 2018; 2 (2): 76-86.

[46] Kaya U, Yılmaz A, Dikmen Y. Deep learning methods used in the field of health. Avrupa Bilim ve Teknoloji Dergisi 2019; 16: 792-808. doi: 10.31590/ejosat.573248

[47] Kutlu Y, Altan G, İşçimen B, Doğdu SA, Turan C. Recognition of species of triglidae family using deep learning. Journal of the Black Sea/Mediterranean Environment 2017; 23 (1): 56-65.

[48] Ma M, Sun C, Chen X. Discriminative deep belief networks with ant colony optimization for health status assessment of machine. IEEE Transactions on Instrumentation and Measurement 2017; 66 (12): 3115-3125. doi: 10.1109/TIM.2017.2735661

[49] Karhunen J, Raiko T, Cho K. Unsupervised deep learning: A short review. In: Advances in Independent Component Analysis and Learning Machines. 1st ed. Finland: Academic Press, 2015, pp. 125-142. doi: 10.1016/B978-0-12-802806-3.00007-5

[50] Doğan F, Türkoğlu İ. Deep learning and application areas. DÜMF Mühendislik Dergisi 2019; 10 (2): 409-445. doi:10.24012/dumf.411130

[51] Ezel E. Image-based Turkish sign language recognition using deep learning method. PhD, Selçuk University, Konya, Turkey, 2018.

[52] Feng W, Guan N, Li Y, Zhang X, Luo Z. Audio visual speech recognition with multimodal recurrent neural networks. In: IEEE 2017 International Joint Conference on Neural Networks (IJCNN); Anchorage, Alaska, USA; 2017. pp. 681-688.

[53] Uysal F, Hardalaç F, Koç M. A Review about segmentation of magnetic resonance prostate images with deep learning. In: TIPTEKNO 2018 Bildirileri; Gazi Magosa, Turkish Republic of Northern Cyprus (TRNC); 2018. pp. 143-146.

[54] Raghavan VV, Gudivada VN, Govindaraju V, Rao CR. Cognitive computing: Theory and applications. North Holland: Elsevier, 2016.

[55] Zocca V, Spacagna G, Slater D, Roelants P. Python Deep Learning. Birmingham: Packt Publishing, 2017.

[56] Ser G, Bati CT. Determining the best model with deep neural networks:Keras application on mushroom data. Yüzüncü Yıl Üniversitesi Tarım Bilimleri Dergisi 2019; 29 (3): 406-417. doi: 10.29133/yyutbd.505086

[57] Rodríguez Sánchez A, Salmerón Gómez R, García C. The coefficient of determination in the ridge regression. Communications in Statistics-Simulation and Computation 2019; 26 (9): 1-19. doi: 10.1080/03610918.2019.1649421

[58] Clark TE, Ravazzolo F. Macroeconomic forecasting performance under alternative specifications of time-varying volatility. Journal of Applied Econometrics 2015; 30 (4): 551-575. doi: 10.1002/jae.2379

[59] Kaur T, Kumar S, Segal R. Application of artificial neural network for short term wind speed forecasting. In: IEEE 2016 Biennial international conference on power and energy systems: towards sustainable energy (PESTSE); Bangalore, Karnataka, India; 2016. pp. 1-5

[60] Ramos P, Santos N, Rebelo R. Performance of state space and ARIMA models for consumer retail sales forecasting. Robotics and Computer-Integrated Manufacturing 2015; 34: 151-163. doi: 10.1016/j.rcim.2014.12.015

[61] Kim S, Kim H. A new metric of absolute percentage error for intermittent demand forecasts. International Journal of Forecasting 2016; 32 (3): 669-679. doi: 10.1016/j.ijforecast.2015.12.003

[62] Fumo N, Biswas MR. Regression analysis for prediction of residential energy consumption. Renewable and Sustainable Energy Reviews 2015; 45: 332-343. doi: 10.1016/j.rser.2015.03.035