IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 31, 2019, accepted April 28, 2019, date of publication May 14, 2019, date of current version May 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2915958

Predicting Delay in IoT Using Deep Learning: A Multiparametric Approach

MUHAMMAD ATEEQ[1], FARRUH ISHMANOV[2], MUHAMMAD KHALIL AFZAL[1], (Senior Member, IEEE), AND MUHAMMAD NAEEM[3], (Senior Member, IEEE)

[1]Department of Computer Science, COMSATS University Islamabad at Wah Cantonment, Wah Cantonment 47040, Pakistan
[2]Department of Electronics and Communication Engineering, Kwangwoon University, Seoul 01897, South Korea
[3]Department of Electrical and Computer Engineering, COMSATS University Islamabad at Wah Cantonment, Wah Cantonment 47040, Pakistan

Corresponding author: Farruh Ishmanov (farruh.uzb@gmail.com)

This research was funded by the Research Grant of Kwangwoon University in 2019.

ABSTRACT The proliferation of the Internet of Things (IoT) requires to accommodate diverse applications with stringent performance requirements. Delay is one of the key metrics in the IoT, particularly, for domains, such as health care, where critical cases requiring an emergency response frequently occur. In this paper, we analyze the performance data generated using the IEEE 802.15.4 standard to derive an accurate predictive model for delay-sensitive applications. A deep neural network (DNN) is adopted to model the relationship between diverse communication parameters (e.g., queue size, application traffic rate, and transmission power) and delay. Evaluation reveals that the DNN model achieves a prediction accuracy of over 98% and outperforms other popular regression models. In addition, a fine-grained analysis of the size of training data, depth (number of layers), width (number of neurons per layer), and epochs (number of iterations) is carried out in an attempt to achieve best possible prediction results with minimally complex DNN. The statistics show that the derived model achieves a comparable accuracy even when trained with a small fraction ($\geq$10%) of data. The proposed model recommends the values for different controllable communication parameters to the transmitter that can be fine-tuned considering the desired delay bounds.

INDEX TERMS Delay prediction, deep learning, e-health, internet of things, multi-layer neural networks, wireless sensor networks.

I. INTRODUCTION

Growth and proliferation in communication and sensing technologies has been enormous in recent years. This has accelerated the realization of the Internet of Things (IoT), connecting ubiquitous physical objects in massive numbers. Wireless Sensor Networks (WSNs) have been at the heart of IoT [1], facilitating diverse application scenarios. IEEE 802.15.4 is one of the most popular communication standards used in WSN deployments. According to a recent survey, WSN deployments for which communication standard is known, more than 50% are based on IEEE 802.15.4 [2]. Application domains served by WSNs include but not limited to: area and health-care monitoring; environmental, earth and industrial sensing; smart grids, and many others [3]–[5]. Depending on the domain of deployment, WSNs have different Quality of Service (QoS) requirements to meet; including energy, reliability, delay, and throughput [6]–[8]. Accordingly, there has been a tremendous research effort in the past couple of decades to improve QoS in WSNs with an aim to serve diverse application requirements [8]–[10]. With considerations of energy at the center of WSN design, applications need to meet multiple, often conflicting QoS metrics, simultaneously. Optimizing multiple and conflicting metrics are frequently modeled as optimization problems. These optimization problems, requiring certain constraints to be met, are often intractable and are identified as NP-hard. To cope with the complexity of these problems and devise practical solutions, a compromise on the accuracy of the solutions has to be afforded. Mathematical programming based serialization methods and nature-inspired meta-heuristic algorithms are popular choices in this context [10]. Besides, most of the solutions are proposed in the form of some protocol at any layer. Realizing the potential benefit(s) of the proposed solution, if any, requires standardization and arduous effort to implement it in the hardware. More so, these solutions

The associate editor coordinating the review of this manuscript and approving it for publication was Byung-Seo Kim.

lack in self-learning and do not adapt as the situations evolve. In summary, the limitations in conventional solutions include:

- Computational complexity of the intractable problems to optimize QoS metrics
- Strenuous effort of implementations to realizing the potential of solutions proposed in the form of protocols at different layers
- Lack of adaptivity and self-learning in legacy solutions

The limitations found in conventional approaches to facilitate QoS in WSN require devising alternative solutions that can potentially eliminate these shortcomings or minimize their effect. Data-driven approaches, using performance data and sophisticated learning algorithms, have emerged as a prime candidate to enable intelligence and adaptivity in modern communication systems with sufficient accuracy. Deep learning, in particular, with its rich armory of customizable neural network structures has been a lucrative playground for researchers from diverse fields including the communication domain. In addition to the increasing computation power to run these sophisticated algorithms comfortably, the availability of quality data is of paramount importance. With the spread of IoT, more and more performance data is becoming available from real-world deployments and testbeds. Considering the growing QoS concerns and demands, availability of state-of-the-art learning algorithms and quality data coupled with rich computation and communication infrastructure, we are motivated to devise an adaptive solution for the delay-sensitive application, based on Deep Neural Networks (DNN) using a real-world dataset collected for IEEE 802.15.4 standard.

There have been a number of studies focusing QoS predictions in wireless and IoT [11]–[18]. Delay prediction is focused in wireless multi-hop routing environments in [11]–[13]. Data from real-world experiment along with large-scale simulation is used to predict delay using regression in industrial WSN [11] with acceptable prediction accuracy. However, the limitations of this study include: the load of running machine learning algorithms is placed on the sensor nodes, and the intervals for which data collection is performed are measured as small as a few seconds. Thus, both limited capacity of sensor devices and lack of sufficient amount of data makes it very difficult to achieve even moderately generalizable results. The studies [12]–[14] focus mobile ad-hoc environments, where mobility and path characteristics are treated as primary concerns in determining delay. Another important factor is that the data used in these research efforts is generated using simulations. On the other hand, the focus in [15] is QoS prediction in IoT environments. The predictions include service response time and throughput. However, the approach used is based on matrix factorization technique and limited to missing value predictions in a data matrix containing values for both response time and throughput. The metrics focused in [16]–[18] include reliability, lifetime, transmission power, distance and energy in WSN. These studies use neural networks to predict the metrics in focus. However, the major limitation is the number of parameters available for making adequately informed predictions. In addition in order to predict one metric at the sender-side (e.g., packet loss ratio), the receiver-side quantities (e.g., number of erroneous packets and signal strength) are used as input to train the model in [17]. Whereas in [16], to predict any one of the many metrics, the other ones are supplied as input to the learning model (e.g., in order to predict any one of lifetime, distance and transmission power, the other two are used as input). These kinds of designs render the effectiveness of learning and applicability of results as impractical. Moreover, there is no literature that investigates the prediction potential of QoS attributes in relation to comprehensive communication parameters to achieve a more accurate, practical and adaptive model. We have already demonstrated the potential of deep learning in predicting packet delivery ratio and energy consumption, compared to legacy regression models [19]. Based on the gaps found in the literature and our previous work [19], the motivations behind this study include:

- To avoid using receiver-side parameter configurations to predict receiver-side QoS metrics,
- Lack in understanding of delay in relation to diverse communication parameters settings
- Lack in fine-grained analysis of deep learning for achieving an adequate prediction accuracy
- To remove the barrier of using rich computation as well as data for by putting the load on remote server instead of resource-constrained sensor devices

Keeping in view the limitations of existing solutions, the focus of this paper is predicting delay in IEEE 802.14.5 network under diverse communication parameters configurations. A number of application areas are identified to be delay-sensitive and mission-critical (e.g., health-care, safety and emergency response). In such applications e.g., in case of remote intense patient care, a critical event must be reported to a monitoring point observing a time limit, to decide a proper action. In addition, this may also involve varying volumes of data depending on the number and types of sensing and observations being done. To carry out predictive modeling of delay using DNN, we contribute the following:

- Analyze the delay in relation to diverse communication parameters in order to identify their respective contribution in determining delay
- Train deep learning models against each individual parameter to figure out their prediction accuracy
- Identify the critical set of parameters that contribute categorically in predicting delay
- Carry out a fine-grained analysis of different hyper-parameters of deep neural networks (e.g., number of epochs, layers and neurons), and the fraction of data used to train the learning model, to achieve precise predictive configurations

To the best of authors' knowledge, this is the first work that uses deep learning for data-driven predictions of delay in IEEE 802.15.4.

TABLE 1. Communication parameters details.

Type	Layer	Parameter	Values
Pre-configured	Application	Inter-Arrival Time: IAT (ms) Packet Size: PS (bytes)	10, 15, 20, 25, 30, 35, 40, 50 20, 35, 50, 65, 80, 95, 110
	MAC	max Queue Size: QS Max Transmissions: MT Retry Delay: RD (ms)	1, 30, 60 1, 3, 5 30, 60
	PHY	Transmission Power: TP Distance: DT (m)	3, 7, 11, 15, 19, 23, 27, 31 10, 15, 20, 25, 30, 35
Per-packet	MAC	Actual Queue Size: AQS Actual Transmissions: AT	actual value 0 to 60 actual value 0 to 5

Rest of this paper is organized as follows: Section II explains the data used and insights in terms of relationships among different parameters and delay. Section III details the adapted DNN. The prediction results and evaluation of hyper-parameters is presented in section IV. Section V concluded the paper and highlights shortcoming in the current study and suggests the future work.

II. THE DATASET AND INSIGHTS

A. THE DATA

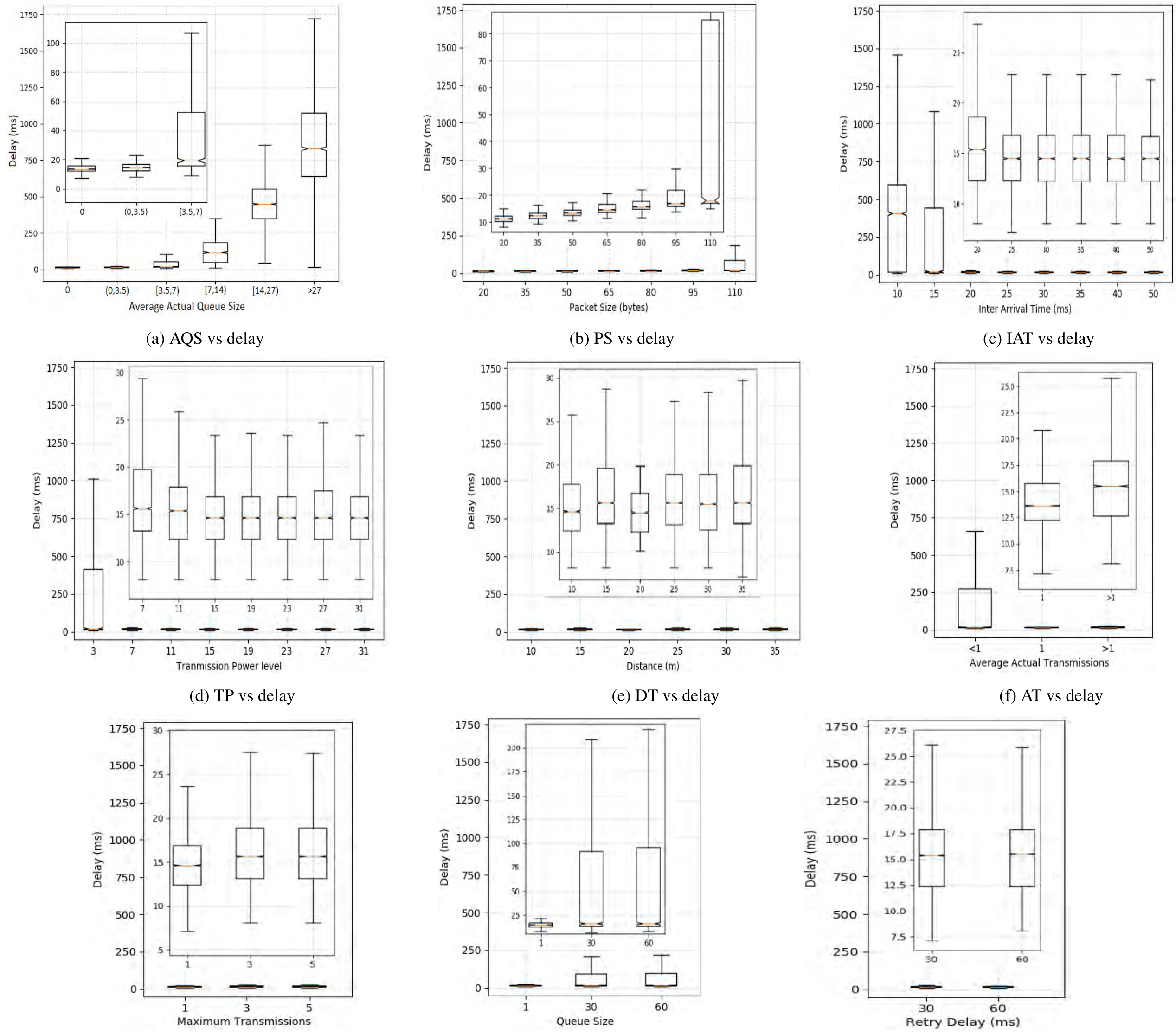
For the purpose of analysis, predictions, and evaluations a publicly available dataset of delay measurement in IEEE 802.15.4 network [20] is used. In the experiments, delay measurements were taken for more than 48 thousands configurations of seven communication stack parameters from different layers. Table 1 summarizes both types of parameters (pre-configured and per-packet) along with their explanation and the range of values tried in the experiments. The application layer parameters include packet Inter-Arrival Time (IAT), and Payload Size (PS). Medium Access Control (MAC) layer parameters consist of maximum Queue Size (QS), Maximum Transmissions (MT) and Retry Delay (RD). At the physical layer, the parameters are Transmission Power level (TP) and Distance (DT) between the nodes. All these seven parameters were pre-configured with certain values detailed in Table 1. For each configuration of these parameters, 300 packet transmissions were done. For each packet transmission, there is rich meta-information consisting of average Actual Queue Size (AQS) and average Actual Transmission number (AT).

B. RELATIONSHIP BETWEEN PARAMETERS AND DELAY

In Fig. 1 and Fig. 2, the relationships among the communication parameters and delay are presented in 2 and 3-dimensions, respectively. The 3-dimensional plots are used to highlight the complex relationships where a combination of parameters seem to define the relationship better than a single parameter. Fig. 1(a) – Fig. 1(i) show relationships of delay with AQS, PS, IAT, TP, DT, AT, QS, MT and RD, respectively. The most prominent parameter that influences the delay values is AQS (Fig. 1(a)), thus highlighting queuing as the most defining factor for delay. With the increase in AQS, there is a significant hike in delay values.

With increase in PS (Fig. 1(b)) the delay increases monotonically. The combined effect of AQS and PS is shown in Fig. 2(a) with a consistent drop in delay as AQS and PS decrease. For IAT (Fig. 1(c)), the values of delay keep decreasing with the decrease in traffic rate at the application. However, the variation, when IAT is 20 or more, is very small. To better capture the trend IAT is also plotted with AQS (Fig. 2(b)) and the density of higher delay values is when IAT is 20 or less. In Fig. 2(c) delay is plotted against the combination of IAT and PS, confirming the same trend as witnessed previously. Thus, these three parameters (i.e., AQS, PS, and IAT) seem to have a fairly consistent relationship with the delay. In the case of TP (Fig. 1(d)), the delay tends to be higher when TP is minimum (i.e., 3). This is due to retransmissions as the TP is not adequate to guarantee successful transmission. However, as TP rises beyond 7, its relationship with delay does not change much. DT (Fig. 1(e)) does not seem to have any definitive relationship with delay because of the dominance of queuing factor.

The relationships of AT and MT with delay are shown in Fig. 1(f) and Fig. 1(g). MT is a pre-configured parameter with discrete values (i.e., 1, 3, and 5), whereas AT is the average of the actual number of transmissions made and represents per-packet values. Since the number of values tried for MT are limited, the relationship of both AT and MT is further studied alongside AQS. Fig. 2(d) presents the relationship of MT and AQS with delay and the trend highlights that both AQS as well as delay tend to grow with increasing value of MT. Following MT, AT is plotted against delay alongside AQS in Fig. 2(e). It is apparent that as long as the average AT remains under a certain level, both AQS and delay do not vary. However, as the average AT rises, there is a consistent rise in both AQS and delay. The relationship between QS and delay (Fig. 1(g)) also suggests a direct proportion. However, the availability of a fine-grained parameter AQS renders QS less effective. The relationship of RD with delay is shown in Fig. 1(i). Although RD directly influences the delay in case of retransmissions, however, the results (Fig. 1(i)) do not convey much information because the dataset contains only two variations of RD values (i.e., 30 and 60). Fig. 2(f) shows a confusion matrix highlighting the correlations between the communication parameters and actual delay values. The dominance of AQS over other parameters is clear with a



(a) AQS vs delay (b) PS vs delay (c) IAT vs delay (d) TP vs delay (e) DT vs delay (f) AT vs delay (g) MT vs delay (h) QS vs delay (i) RD vs delay

FIGURE 1. Relationship between communication parameters and delay. (a) AQS vs delay. (b) PS vs delay. (c) IAT vs delay. (d) TP vs delay. (e) DT vs delay. (f) AT vs delay. (g) MT vs delay. (h) QS vs delay. (i) RD vs delay.

correlation of 0.91. The confusion matrix also confirms the positive (AQS, QS, PS, and MT) and negative (IAT and TP) correlations of the parameters under consideration. The weaker correlation between DT and RD is also apparent. Whereas, a lower negative correlation of AT indicates the complexity of its correlation with delay.

In summary, both Fig. 1 and Fig. 2 demonstrate the relationship among delay and different communication parameters, motivating the use of deep learning in predicting delay based on these parameters.

III. SYSTEM MODEL

A. APPLICATION FRAMEWORK

The application scenario considered in this study comprises an intelligent healthcare system where various delay-sensitive operations, for example, critical patient care, remote monitoring, hospital equipment management and emergency services are running. Delay, beyond a certain limit, can have fatal consequences. Performance data from multiple sites is periodically communicated to the remote server where the adopted DNN is trained to predict delay and send back the recommended values of different parameters considering the delay bounds. The application model is described in Fig. 3.

B. DEEP NEURAL NETWORK

DNN is adapted for modeling the predictive relationships between the stack parameters and delay. Since there are nine input parameters (seven pre-configured and two per-packet) and one prediction target, the feature vector is represented as:

$$(P_{n,1}, P_{n,2}, \ldots, P_{n,9}, D) \tag{1}$$

where P, n and D represent the features (parameters), tuple number in the data which are 48384 to be exact and the prediction target (delay), respectively. Data is divided into

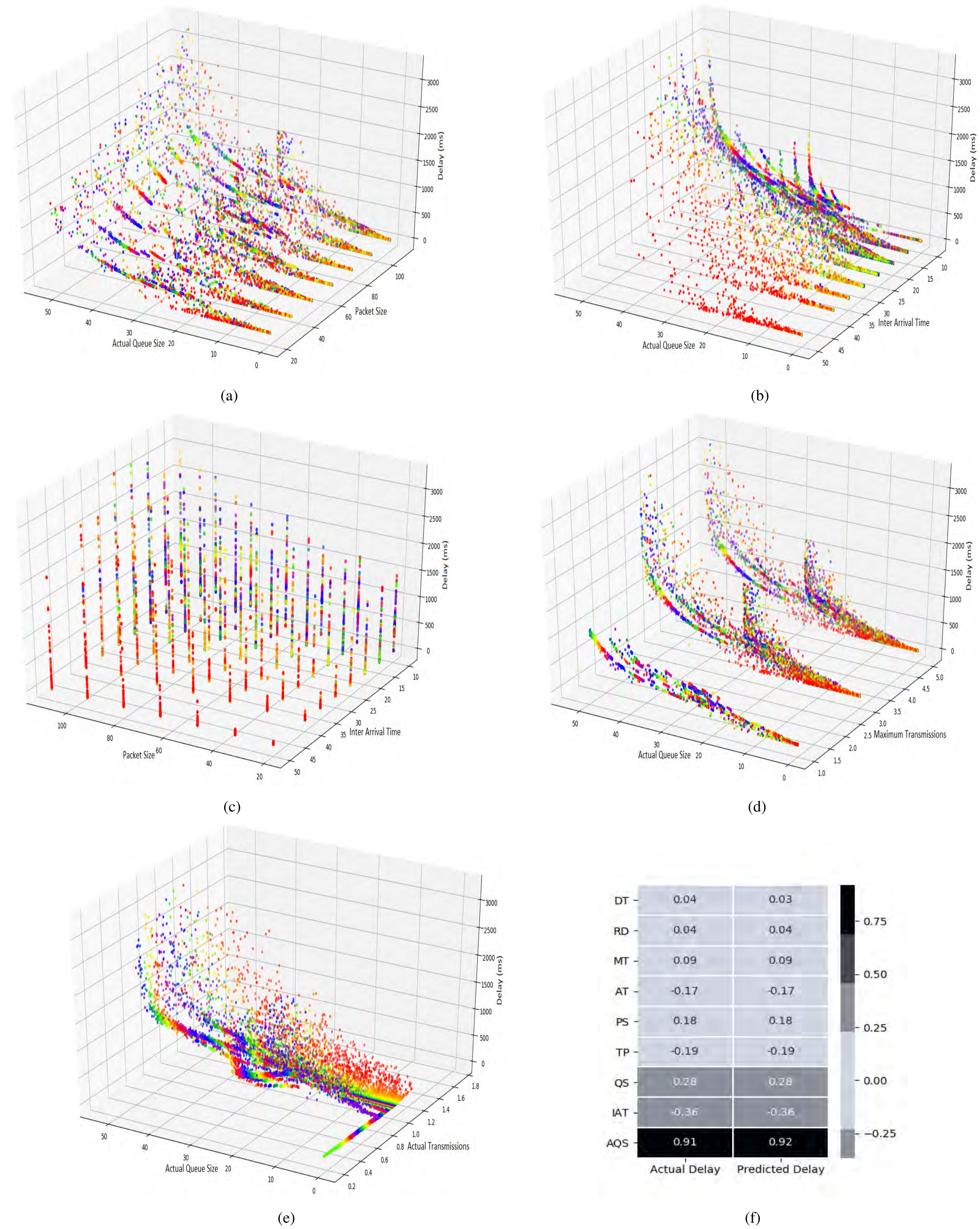


FIGURE 2. Relationships between communication parameters and delay. (a)AQS and PS vs delay. (b) AQS and IAT vs delay. (c) PS and IAT vs delay. (d) AQS and MT vs delay. (e) AQS and AT vs delay. (f) Correlation matrix.

three fragments where 50% is used for training, 20% is used for validation, and 30% for testing the model. As regard DNN hyper-parameters; the number of layers, neurons per layer, epochs, and learning rate are set to 10, 64 and 2000, and 0.001, respectively. The activation function used is rectified linear unit since the prediction target is continuous in nature. These hyper-parameters are chosen empirically to facilitate unconstrained learning with a sufficiently large network. A fine-grained analysis of these hyper-parameters, as well as the fraction of training data, is carried out in section IV-B, however. The computation in DNN is carried out in two phases regarded as forward and backward

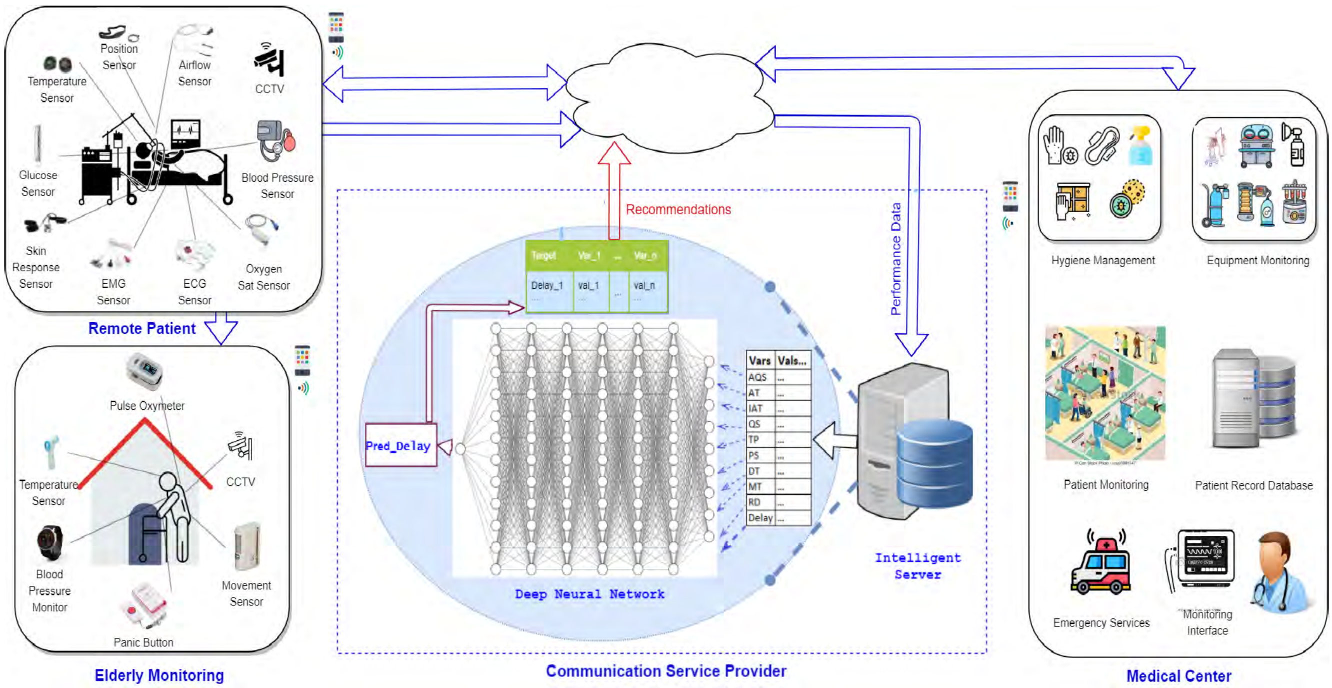


FIGURE 3. System model.

propagation, respectively. A forward pass is completed using output vector O calculation at a layer l:

$$O_{[l]} = W_{[l]} \cdot X_{[l-1]} + c_{[l]}, \tag{2}$$

where W represent weights given to the input feature vector X, and c is the parameter of regression. The activation function f is applied to this raw output O to calculate the actual output X of the current layer which also behaves as input to the next layer.

$$X_{[l]} = f_{[l]}(O_{[l]}), \tag{3}$$

The completion of the forward pass is followed by backward pass which is used to update weights through computation of the rate of change d for O, W, c, X and f:

$$dO_{[l]} = dX_{[l]} \times df_{[l]}(O_{[l]}). \tag{4}$$

$$dW_{[l]} = \frac{1}{n}(dO_{[l]} \cdot X_{[l].T}), \tag{5}$$

$$dc_{[l]} = \frac{1}{n}\sum_{i=1}^{n} O_{[l]}, \tag{6}$$

$$dX_{[l-1]} = W_{[l].T} \cdot dX_{[l]}, \tag{7}$$

where n represents the number of records in the training data. Following the backward propagation, gradient descent is performed till convergence, updating W and b, aiming the error minimization:

$$W_{[l]} = W_{[l]} - \alpha \times dW_{[l]}. \tag{8}$$

$$b_{[l]} = b_{[l]} - \alpha \times dc_{[l]}. \tag{9}$$

C. EVALUATION

In order to analyze the prediction accuracy of the DNN Root Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) are used as shown in (10) and (11), respectively:

$$MAE = \frac{1}{n_t}\sum_{i=1}^{n_t} |E_i| \tag{10}$$

$$MAPE = \frac{100}{n_t}\sum_{i=1}^{n_t} \frac{|E_i|}{Y_i}. \tag{11}$$

whereas,

$$E_i = Y_i - Y_i', \tag{12}$$

where i represents the row index, Y_i and Y_i' represent actual and predicted values for i^{th} row in training data and n_t is the number of training examples.

IV. PERFORMANCE EVALUATION

In the following, we discuss the prediction results, characterize the error, and carry out a fine-grained analysis of DNN.

A. EFFECTS OF PARAMETERS AND THEIR COMBINATIONS

Fig. 4 presents the prediction results for different communication parameters and their combinations. MAE against each parameter is shown in Fig. 4(a). It is evident that AQS (with MAE of 29.7) outperforms all other parameters and is at least 3.8 times more accurate than any other parameter. Next is AT that achieves an MAE of 113.2. It is interesting to note that the top two parameters in achieving good prediction accuracy

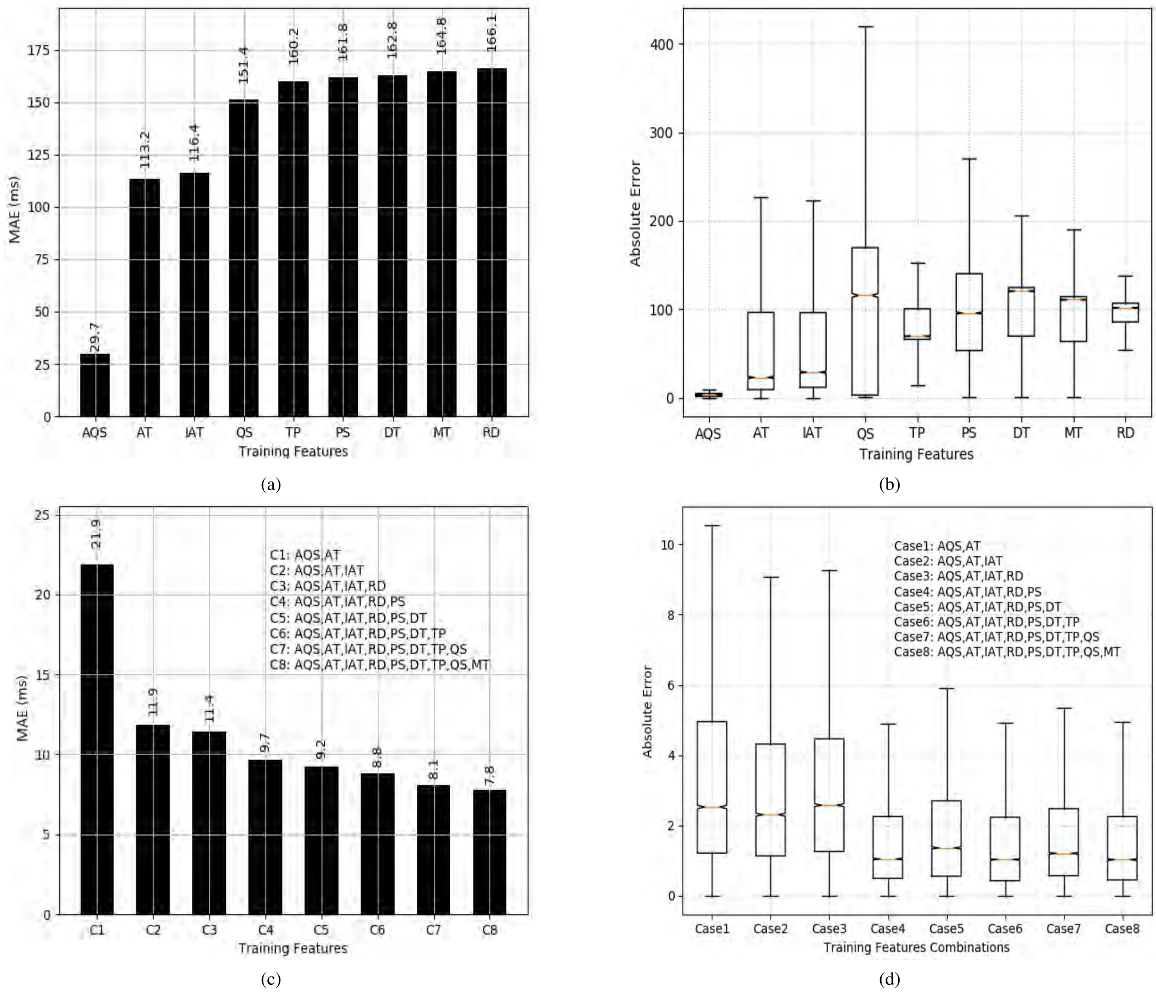
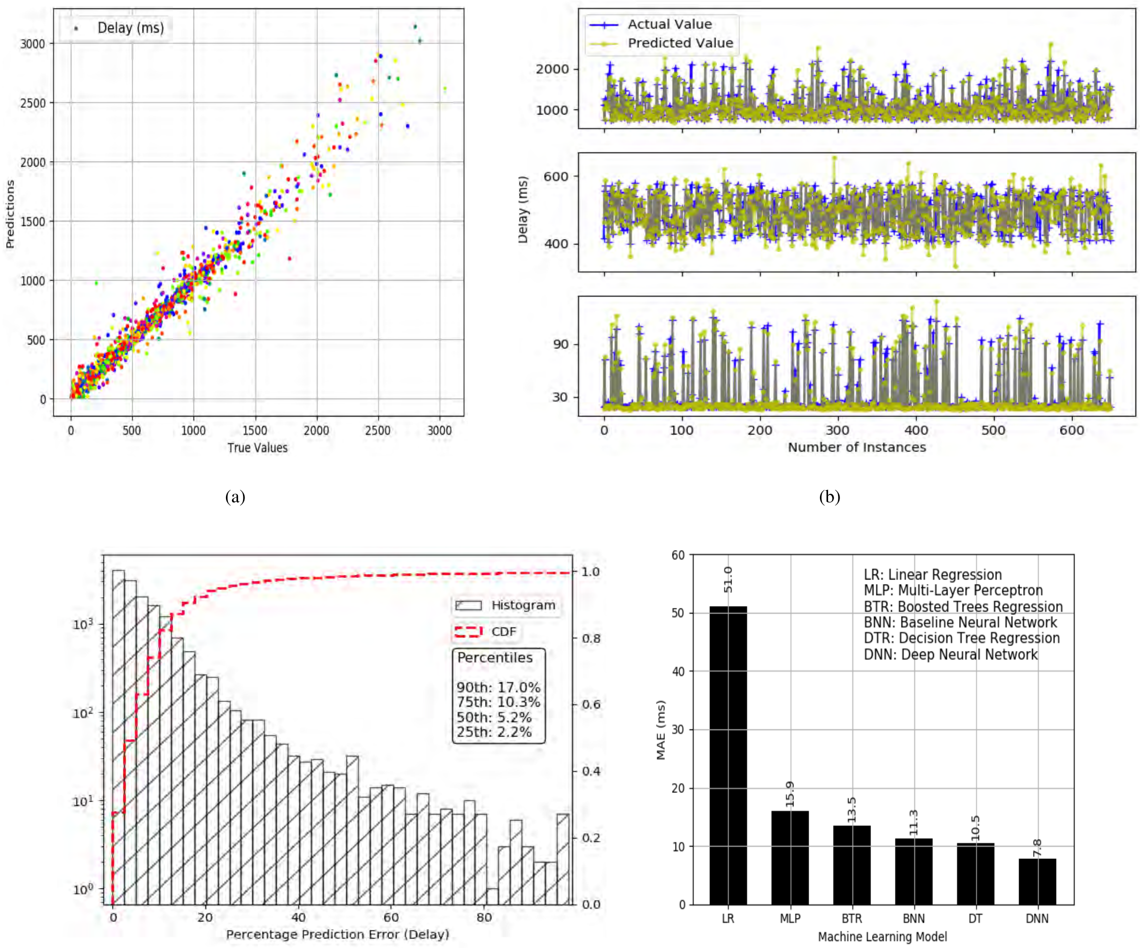


FIGURE 4. Prediction results for individual and combinations of communication parameters. (a) MAE for individual parameters. (b) Absolute error for individual parameters. (c) MAE for combinations of parameters. (d) Absolute error for combinations of parameters.

are the ones that contain fine-grained per-packet values as against the other pre-configured parameters. Following AQS and AT, are IAT and QS, achieving accuracies of 116.4 and 151.4, respectively. TP, PS, DT, MT, and RD are next in the sequence, all achieving almost similar accuracies. In order to present a closer perspective, box plots are used to highlight the absolute error quartiles in Fig. 4(b). The third quartiles of errors for AQS, AT, IAT, QS, TP, PS, DT, MT and RD are 4.8, 96.5, 96.0, 169.5, 100.5, 140.0, 124.5, 114.5, and 107.0, respectively. This indicates that the spread of error is not necessarily aligned with MAE. Fig. 4(c) presents the prediction error for the combinations of different features. Starting with AQS, iteratively the next best features that minimize the errors are AT, IAT, RD, PS, DT, TP, QS and MT, offering improvements of 45%, 4%, 14%, 5%, 4%, 7% and 3%, respectively. Therefore, it is evident that all features make some contribution in predicting delay thought the proportional contribution monotonically keeps decreasing with the addition of features. The confusion matrix (Fig. 2(f)) shows the correlations of the parameters with actual as well as predicted delay values and the values for both actual as well as predicted values are the same. Fig. 4(d) shows the box plots for the same combinations of parameters. The third quartiles of errors with addition of AT, IAT, RD, PS, DT, TP, QS and MT are 4.96, 4.32, 4.47, 4.26, 2.26, 2.71, 2.24, 2.48 and 2.25, respectively. Therefore, it can be concluded that the 75th error percentile of 2.24 ms, where delay ranges between 7 and 3100 ms, indicates an excellent prediction accuracy.

Fig. 5 further characterizes the prediction error. Actual values are plotted against predicted values in Fig. 5(a) and Fig. 5(b) in the form of scatter-plot and line-plot, respectively. Both plots indicate highly precise prediction results. A histogram and CDF of percentage errors are shown in Fig. 5(c), where 50, 75 and 90th percentiles indicate that the predicted values are within at least 5.2, 10.3 and 17% of the actual values, respectively.

A comparison of different machine learning models is shown in Fig. 5(d) Values of MAE indicate the superiority



(a) (b) (c) (d)

FIGURE 5. **Prediction results of DNN. (a)Actual values vs Predicted values. (b) Actual values vs Predicted values. (c) Mean percentage error. (d) Comparison of machine learning models.**

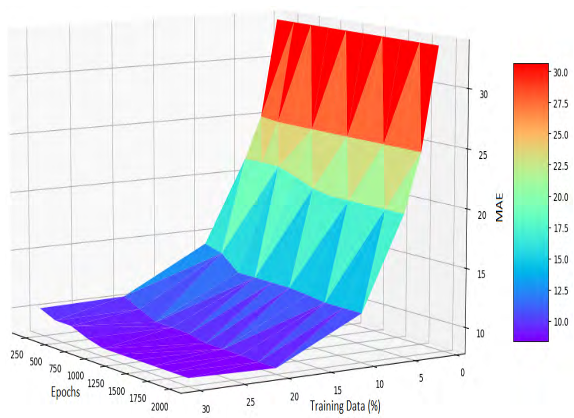


FIGURE 6. **Epochs vs training data vs delay.**

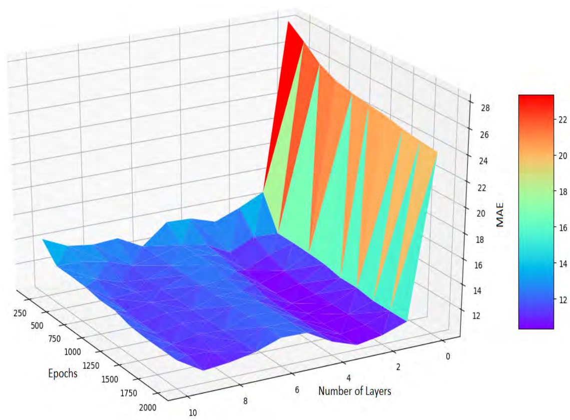


FIGURE 7. **Epochs vs number of hidden layers vs delay.**

of DNN where linear regression, multi-layer perceptron, boosted trees regression, baseline neural network, decision trees regression, and DNN achieve MAE values of 51.0, 15.9, 13.5, 11.3, 10.5, and 7.8, respectively.

B. EFFECT OF DIMENSIONALITY

The results for fine-tuned DNN are shown in Fig. 8. The performance of DNN with varying sizes of training data, number of layer and number of neurons per layer are illustrated

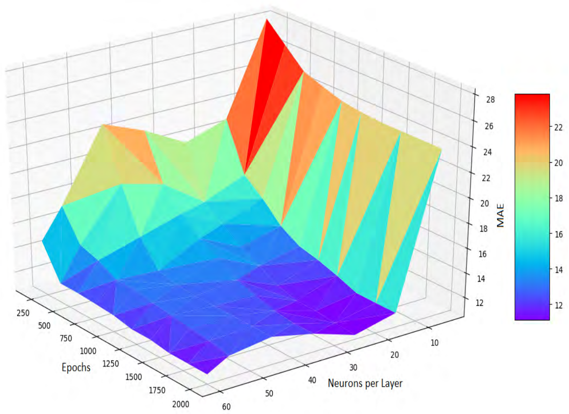


FIGURE 8. Epochs vs neurons per layer vs delay.

in Fig. 8(a) – Fig. 8(c), respectively. According to Fig. 8(a) the prediction error decreases as the fraction of data used for training the DNN model increase. The MAE improves rapidly when training data is increased from 1 to 10%. However, the improvement after 10% is only fractional. This shows that DNN can perform very well even with a small fraction of data is used for training the model. It is further evident that the number of epochs to achieve stable results is around 600, after which no further significant improvement is gained. In Fig. 8(b), results are presented for the number of layers using 10% of the data for training the model. It is clear that the most significant gain in MAE is achieved when the model moves from zero hidden layers to one. Addition of any further hidden layer after the first layer again fails to contribute any significant improvement. It is further observed that 400 epochs suffice to reach a stable value for MAE. Using the fine-grained observations of 10% training data and a single hidden layer, Fig. 8(c) highlights the effect of different numbers of neurons used in the hidden layer. It is clear from Fig. 8(c) that 15 neurons suffice to achieve a stable value for MAE with 1000 epochs. However, as the number of neurons is increased beyond 35, the numbers of epochs to achieve a stable figure for MAE decrease.

In summary, from the fine-grained analysis of DNN, it can be concluded that a small fraction of training data (10%), a single hidden layer, and a very few neurons (15) achieve sufficient prediction accuracy. This observation strengthens the case for adopting DNN for QoS predictions without needing a huge amount of data and affording a very complex and time-consuming neural network structure and computation.

V. CONCLUSION

In this paper, DNN is adopted for predicting delay in IEEE 802.15.4 based applications in IoT. Results reveal that DNN achieves a prediction accuracy of over 98% in predicting delay. Different hyper-parameters are fine-tuned in order to further understand the behavior of DNN. The fine-grained analysis revealed that DNN achieves a very good prediction accuracy even when trained with 10% of the data, a single hidden layer with just 15 neurons. A comparison of DNN with other popular machine learning models is also carried out. The results from this study strengthen the fact that deep learning possesses the potential to predict QoS metrics in IoT with very good accuracy without having to deal with intractable NP-hard optimization problems. In the future, we aim to develop a working prototype of the proposed approach with a focus on factors surrounding the frequency of invocation of DNN, the choice of data in the temporal domain and others.

REFERENCES

[1] M. Luca, P. Luigi, and V. Antonio, "Evolution of wireless sensor networks towards the Internet of Things: A survey," in *Proc. 19th Int. Conf. (SoftCOM)*, Sep. 2011, pp. 1–6.
[2] G. Strazdins, A. Elsts, K. Nesenbergs, and L. Selavo, "Wireless sensor network operating system design rules based on real-world deployment survey," *J. Sensor Actuat. Netw.*, vol. 2, no. 3, pp. 509–556, 2013.
[3] R. Bushra and R. M. Husain, "Applications of wireless sensor networks for urban areas: A survey," *J. Netw. Comput. Appl.*, vol. 60, pp. 192–219, Jan. 2016.
[4] L. M. Borges, F. J. Velez, and A. S. Lebres, "Survey on the characterization and classification of wireless sensor network applications," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 1860–1890, 4th Quart., 2014.
[5] A. B. Noel, A. Abdaoui, T. Elfouly, M. H. Ahmed, A. Badawy, and M. S. Shehata, "Structural health monitoring using wireless sensor networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1403–1423, 3rd Quart., 2017.
[6] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Comput. Netw.*, vol. 67, pp. 104–122, Jul. 2014.
[7] D. Yuan, S. S. Kanhere, and M. Hollick, "Instrumenting wireless sensor networks—A survey on the metrics that matter," *Pervasive Mobile Comput.*, vol. 37, pp. 45–62, Jun. 2017.
[8] I. Al-Anbagi, M. Erol-Kantarci, and H. T. Mouftah, "A survey on cross-layer quality-of-service approaches in WSNs for delay and reliability-aware applications," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 525–552, 1st Quart., 2016.
[9] P. Huang, L. Xiao, S. Soltani, M. W. Mutka, and N. Xi, "The evolution of mac protocols in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 101–120, 1st Quart., 2013.
[10] Z. Fei, B. Li, S. Yang, C. Xing, H. Chen, and L. Hanzo, "A survey of multi-objective optimization in wireless sensor networks: Metrics, algorithms, and open problems," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 550–586, 1st Quart., 2017.
[11] X. Zhou *et al.*, "Pido: Predictive delay optimization for intertidal wireless sensor networks," *Sensors*, vol. 18, no. 5, p. 1464, 2018.
[12] H. Tuli, and S. Kumar, "Prediction analysis of delay in transferring the packets in adhoc networks," in *Proc. 3rd Int. Conf. Comput. Sustain. Global Develop. (INDIACom)*, Mar. 2016, pp. 660–662.
[13] J. P. Singh, P. Dutta, and A. Chakrabarti, "Weighted delay prediction in mobile ad hoc network using fuzzy time series," *Egyptian Informat. J.*, vol. 15, no. 2, pp. 105–114, 2014.
[14] A. Alkazzaz, "Predicting and classifying packet transmission efficiency in bio-inspired wireless sensor networks," M.S. thesis, Virginia Commonwealth Univ., Richmond, VA, USA, 2014. [Online]. Available: http://scholarscompass.vcu.edu/etd/3489?utm_source=scholarscompass.vcu.edu%2Fetd%2F3489&utm_medium=PDF&utm_campaign=PDFCoverPages
[15] G. White, A. Palade, C. Cabrera, and S. Clarke, "IoTPredict: Collaborative QoS prediction in IoT," *IEEE PerCom*, Mar. 2018, pp. 1–10.
[16] A. Ayhan, H. U. Yildiz, A. M. Ozbayoglu, and B. Tavli, "Neural network based instant parameter prediction for wireless sensor network optimization models," in *Wireless Networks*. Berlin, Germany: Springer, 2018, pp. 1–14.
[17] M. Kulin, E. de Poorter, T. Kazaz, and I. Moerman, "Poster: Towards a cognitive MAC layer: Predicting the MAC-level performance in Dynamic WSN using Machine learning," in *Proc. EWSN*, 2017, pp. 214–215.
[18] T. Liu and A. E. Cerpa, "Data-driven link quality prediction using link features," *ACM Trans. Sensor Netw.*, vol. 10, no. 2, p. 37, 2014.
[19] M. Ateeq, F. Ishmanov, M. K. Afzal, and M. Naeem, "Multi-parametric analysis of reliability and energy consumption in IoT: A deep learning approach," *Sensors*, vol. 19, no. 2, p. 309, 2019.
[20] F. Songwei and Z. Yan. (2015). *CRAWDAD Dataset Due/Packet-Delivery (V. 2015-04-01)*. [Online]. Available: https://crawdad.org/due/packet-delivery/20150401. doi: 10.15783/C7NP4Z.

MUHAMMAD ATEEQ received the bachelor's degree from Bahauddin Zakariya University at Multan, in 2005, and the master's degree in computer science from the COMSATS Institute of Information Technology, Wah Cantonment, in 2007. He is currently pursuing the Ph.D. degree in computer science with COMSATS University Islamabad. He has been with academia, for last 11 years. He is currently an Assistant Professor of computer science with The Islamia University of Bahawalpur. His research interest includes using data driven techniques to improve quality of service in wireless communication.

FARRUH ISHMANOV received the B.S. degree in information systems from the Tashkent State University of Economics, Uzbekistan, in 2007, and the M.S. and Ph.D. degrees from the Department of Information and Communication Engineering, Yeungnam University, South Korea, in 2009 and 2014, respectively. He was with the Multimedia Laboratory, Tashkent State University of Economics, during his undergraduate studies. In 2015, he joined the Department of Electronics and Communication Engineering, Kwangwoon University, South Korea, where he is currently an Assistant Professor. His research interests include resource management, and security in wireless sensor networks and the IoT. He received the Korean Government IITA Scholarship for pursuing the M.S degree.

MUHAMMAD KHALIL AFZAL (SM'16) received the B.S. and M.S. degrees in computer science from the COMSATS Institute of Information Technology at Wah Cantonment, Wah Cantonment, Pakistan, in 2004 and 2007, respectively, and the Ph.D. degree from the Department of Information and Communication Engineering, Yeungnam University, South Korea, in 2014. From 2008 to 2009, he was a Lecturer with Bahauddin Zakariya University at Multan, Multan, Pakistan. From 2009 to 2011, he was a Lecturer with King Khalid University, Abha, Saudi Arabia. He is currently an Assistant Professor with the Department of Computer Science, COMSATS Institute of Information Technology at Wah Cantonment. His research interests include wireless sensor networks, ad hoc networks, smart cities, and the IoT. He is also serving as a Reviewer for the IEEE Access, *Computers and Electrical Engineering* (Elsevier), the *Journal of Network and Computer Applications* (Elsevier), *Future Generation Computer Systems*, and the IEEE Transactions on Vehicular Technology and the Guest Editor for *Future Generation Computer Systems* (Elsevier), the IEEE Access, and the *Journal of Ambient Intelligence and Humanized Computing* (Springer).

MUHAMMAD NAEEM received the B.S. and M.S. degrees in electrical engineering from the University of Engineering and Technology at Taxila, Taxila, Pakistan, in 2000 and 2005, respectively, and the Ph.D. degree from Simon Fraser University, Burnaby, BC, Canada, in 2011. From 2000 to 2005, he was a Senior Design Engineer with Comcept (Pvt.) Ltd., Islamabad, Pakistan, where he participated in the design and development of smart card-based GSM and CDMA pay phones at the Department of Design. From 2012 to 2013, he was a Postdoctoral Research Associate with the Wireless Networks and Communications Research (WINCORE) Laboratory, Ryerson University, Toronto, ON, Canada. Since 2013, he has been an Assistant Professor with the Department of Electrical Engineering, COMSATS Institute of Information Technology at Wah Cantonment, Wah Cantonment, Pakistan, and a Research Associate with the WINCORE Laboratory. He is also a Microsoft Certified Solution Developer. His research interests include the optimization of wireless communication systems, nonconvex optimization, resource allocation in cognitive radio networks, and approximation algorithms for mixed-integer programming in communication systems. He was a recipient of the NSERC CGS Scholarship.