An Efficient and Light-weight Predictive Channel Assignment Scheme for Multi-Band B5G Enabled Massive IoT: A Deep Learning Approach

Sadman Sakib, Tahrat Tazrin, Mostafa M. Fouda, *Senior Member, IEEE,* and Zubair Md. Fadlullah, *Senior Member, IEEE,* and Nidal Nasser, *Senior Member, IEEE*

***Abstract*—Multi-hop Device-to-Device (D2D) enabled relay networks are envisaged to be utilized by the Internet of Things (IoT) and massive Machine Type Communication (mMTC) traffic for the purpose of offloading data in Beyond Fifth Generation (B5G) networks. The emerging challenge of spectrum scarcity and overloading of cellular base stations can be addressed using such relay nodes in terms of spectrum and energy efficiency. In order to improve spectral efficiency, in this paper, we intend to employ several frequency bands concurrently in the relay node rather than the traditional concept of specifying one channel on a specific band at a time. A deep learning-based predictive channel selection method is leveraged to unravel the potential challenges associated with the dynamic channel conditions in the multi-band relay networks. For predicting the most appropriate channel based on its quality, Signal-to-Interference-plus-Noise-Ratio (SINR) is adopted as the metric, which is predicted by the proposed Convolutional Neural Network (CNN) model. The best modulation and coding rates of the predicted band are attained in order to transmit the packets received from the source or previous relay node to the successive relay node/destination. Two proactive channel assignment strategies, referred to as controlled and smart prediction schemes, are employed to exhibit the performance of the shallow and deep-CNN models. The proposed model is evaluated on multiple publicly available datasets from diverse network systems and compared with several machine/deep learning methods. Our proposal leads to encouraging results for proactively predicting the conditions of the channels and choosing the most suitable ones in multi-band relay systems.**

***Index Terms*—Internet of Things (IoT), Device-to-Device (D2D), Beyond 5G (B5G) network, deep learning, Convolutional Neural Network (CNN).**

I. Introduction

In Beyond Fifth Generation (B5G) networks, wireless relay-based communication technologies (e.g., Device-to-Device (D2D) communications) are developing as a propitious technique in the era of Internet of Things (IoT) and massive Machine Type Communication (mMTC) [1]–[5]. The legacy cellular networks were initially employed to fulfill human-driven services, and hence were unable to keep up with the surging IoT/mMTC traffic. Various mobile devices, such as user-smartphones, Unmanned Aerial Vehicles (UAVs) and so forth, can be exploited as D2D nodes or relays [6], in order to augment cellular base stations as represented in Fig. 1, which does not require any additional transmission power [7]–[9]. Therefore, a greater coverage area can be attained to deploy IoT devices over remote communities and a greater radius of natural resources (e.g., forest, oil-rigs, energy supply lines, and so forth). In such systems, it is vital to preserve channel quality and minimize the delay and packet drop rate for a more efficient spectrum and throughput. In the traditional data transmission technique, namely Decode and Forward (DF) [10], [11], data are encoded and forwarded from a source to relay nodes where decoding and demodulation are performed and re-encoded to pass to the succeeding relay/destination node [12]. Nevertheless, this process of reception followed by decoding, encoding, and forwarding at the relay node triggers a considerable delay and high packet drop rate. Another technique called Truncated Decode and Forward (TDF) was proposed by one of the co-authors [13], that utilized multiple bands for concurrent data reception and transmission over a wireless network. The propositioned concept was such that, while receiving data through one of the channels of the node, data will be simultaneously transmitted to the following node through another channel of the node. This paper justifies the improvement of throughput by implementing a preemptive

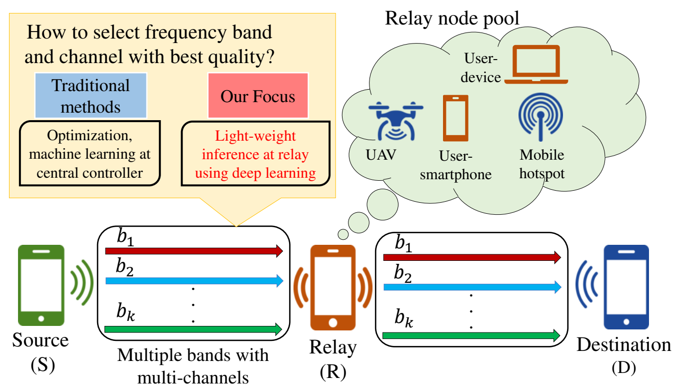


Fig. 1: Our research focus compared to the traditional focus for selecting the best channel of multi-band relay networks.

Corresponding author: Nidal Nasser.
S. Sakib, and T. Tazrin are with the Department of Computer Science, Lakehead University, Thunder Bay, Ontario, Canada. Emails: {ssak2921, ttazrin}@lakeheadu.ca
M. M. Fouda is with the Department of Electrical and Computer Engineering, College of Science and Engineering, Idaho State University, Pocatello, ID 83209, USA, and the Department of Electrical Engineering, Faculty of Engineering at Shoubra, Benha University, Cairo 11672, Egypt. E-mail: mfouda@ieee.org
Z. M. Fadlullah is with the Department of Computer Science, Lakehead University; and Thunder Bay Regional Health Research Institute (TBRHRI), Thunder Bay, Ontario, Canada. Email: Zubair.Fadlullah@lakeheadu.ca
N. Nasser is with the College of Engineering, Alfaisal University, KSA. Email: nnasser@alfaisal.edu

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/JIOT.2020.3032516

approach of predicting and then selecting the best channel and band for data transmission in advance in multi-hop relay systems offloading massive IoT traffic.

The prediction of network traffic flows and Channel State Information (CSI) using Artificial Intelligence (AI) have been heavily analyzed throughout the years [14]–[16]. However, none of these systems have exploited the notion of predicting optimal band/channel in a multi-band relay system, as shown in Fig. 1 where each relay node employs miscellaneous bands (e.g., 5GHz, 2.4GHz, and 920MHz). These stated relay nodes are considered to be resource-constrained and, therefore, are unable to train a complex deep learning model locally. In order to solve this problem of the multi-band, multi-channel prediction task with marginal error, we intend to implement various pre-trained AI models [17]–[19] such as Linear Regression (LR) [20], Auto Regression (AR) [21], Artificial Neural Network (ANN) [22], and Convolutional Neural Network (CNN) [23] with shallow and deep layers. In the adopted shallow architecture and deep architecture for both the ANN and CNN models, we employed one hidden layer for the shallow construct, and multiple hidden layers (four layers) for the deep construct. These pre-trained DL models will be installed locally in the relay nodes to make precise channel quality predictions. From the experimental results, it is found that, among these models, shallow-CNN provided the most promising outcome in predicting the best channel for transmitting data in the resource-constrained relay nodes. After an optimal channel is predicted in the relay node through our distinct deep learning model, the modulation, coding rate, and sending rate are acquired from the Modulation and Coding Scheme (MCS) table, that are used to calculate the link rates. The necessity for these data transfers is determined by first transferring the header of the data frame to the relay node followed by transfer of rest of the data based on a respective decision. The data is then further forwarded to the destination node from the relay node through another band. Since data is being transferred simultaneously in this method, the delay is minimized significantly, and thus this method performs better than the conventional approach where data is transmitted only after the reception of the entire data frame. The model is validated using extensive computer-based simulations and real datasets in order to ensure accuracy and efficacy.

The rest of the paper is constructed as follows. Sec. II surveys the relevant research work. The proposed multi-band network architecture is described in Sec. III. The following Sec. IV describes the existing problems related to channel selection in multi-band systems. Our proposed input representation and deep learning model are demonstrated in Sec. V. Then, in the Sec. VI, the algorithmic analysis of the proposed method is conducted. An illustrative example of how the proposed model produces the output from the input through different stages is manifested in Sec. VII. The performance of our proposal is assessed in Sec. VIII and contrasted with those of LR, and AR. Finally, Sec IX concludes the paper.

II. Related Work

In this section, an extensive literature review has been performed by assessing from the standpoint of two domains – implementations of machine/deep learning models to predict diverse parameters related to network condition and constructions of various algorithms to improve spectral efficiency of the overall scheduling process.

A. Wireless Network Condition Prediction Using AI models

An extensive survey conducted by Mao et al. [17] established the significance and possibilities of deep learning in numerous wireless network scenarios. AI techniques have been utilized to predict network traffic efficiently. Predictors from multiple classes, including classic time series, ANN model-based, and wavelet transform-based predictors, have been proved to be viable for predicting network traffic [24]. A deep learning architecture based on the deep belief network is proposed for predicting network traffic in the wireless mesh network [25]. In the wireless network domain, AI-based models have also been exceptionally prevalent for predicting link qualities and links susceptible to failure [26]–[28]. The work in [29], evaluated the link quality using channel rank measurement along with machine learning algorithms based on the network's Received Signal Strength Indicator (RSSI) and Link Quality Indicator (LQI). Herath et al. [30] predicted a successive series of signal strength data based on previous time-slots using Recurrent Neural Networks (RNNs) such as Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) models. The proposed models in [30] were able to surpass baseline techniques like linear regression and autoregression; however, their work assumed such complex computations to be conducted at the base stations with adequate computational resources. Therefore, the deep learning prediction can be further enhanced for light-weight real-time predictive channel assignment by transferring to resource-constrained relay nodes.

B. Multi-Band Scheduling Over Relay Networks

In a prior study conducted by one of the coauthors' [13], the spectral efficiency in a multi-band relay transmission scenario was heightened while keeping the end-to-end communication delay minimal. A TDF method was integrated at the relay node instead of the traditional DF method to perform demodulation, de-interleaving, de-puncturing, and Viterbi decoding. To evaluate the channel quality, Signal-to-Noise-Ratio (SNR) was considered for each channel, and an MCS table was used in addition to calculating the transmission duration and the channel with lowest transmission time was selected for data transfer. In an extension to the work [31], the Signal-to-Interference-plus-Noise-Ratio (SINR) was assumed while selecting the channel in addition to a finite buffer size at the relay node. Moreover, the impact of physical proximity and channel conditions upon the interface of a multi-band system in order to improve the efficiency was stated in [32].

The research works stated above reveals a significant gap in the realms of channel selection and channel assignment in wireless systems. Therefore, it can be established that the prevailing studies have not explicitly explored the problem of multi-band channel prediction in relay networks for proactive assignment.

III. Proposed System Model

In this section, we discuss the multi-hop, multi-band network architecture, and transmission model that has been regarded in this paper.

A. Network Topology

The relay network topology is represented as a cohesive cellular-D2D system containing a set of N transmitter-receiver (Tx-Rx) pairs. The Tx-Rx pair can contain different configurations as follows: (i) a source node S, such as an IoT device, that transfers its collected data to a mobile User Equipment (UE) (e.g., an energy and performance constrained user-smartphone); (ii) a D2D relay node R that transmits data to the succeeding node; and (iii) A D2D Rx node also called destination node D (such as cellular gateway or base station), that receives data from a D2D relay node. The Tx-Rx pair can communicate over links having L frequency bands with a finite number of channels C. The regarded network topology's routing matrix can be stated as $R = [r_{nlc}] \in \{0, 1\}^{\{N \times L \times C\}}$. If the data is routed across band l and channel c between nodes, the element r_{nlc} equals 1; otherwise, it is 0. In order to improve throughput of this whole architecture to perform improved scheduling, the channel quality q_C should be detected precisely. To perform a predictive channel assignment system, the ground truth, i.e., the channel quality in terms of SINR, can be established by the following formulation:

$$SINR = 10log_{10}\frac{P_S}{(P_I + P_N)}, \tag{1}$$

where P_S, P_I, and P_N denote the desired signal power, interference signal power, and noise power, respectively. P_S is measured as follows.

$$P_S = P_T(\frac{v}{4\pi dl})^2, \tag{2}$$

where P_T, v, d, and l denote the transmission power, the velocity of light and the distance between the Tx-Rx pair and the frequency band of the currently assigned channel, respectively.

In addition to that, the interference signal power P_I is computed as follows.

$$P_I = \int_{\omega} P_T(\frac{v}{4\pi dl})^2 \ n_I * \rho(x, y)dS, \tag{3}$$

where dS indicates an infinitesimal area (i.e., the communication range) in the circles centered both at the Tx and Rx nodes given by ω; $\rho(x, y)$ denotes the probability density function of the interfering nodes in the communication area of the Tx-Rx pair; and n_I represents the total number of nodes distributed around the Tx-Rx pair. For each channel c on each band l, a separate n_I value needs to be considered.

B. Packet Transmission Model

A TDF model was considered to carry out packet transmission, that employs multiple channels for simultaneous data transfer from source to destination node using (N-2) relay nodes. In order to comprehend the necessity of relay node transmission, the header of the data block is forwarded to the subsequent relay node initially. If a successful header transmission occurs, the rest of the data is also forwarded to the relay node. The link rate L_r of the links between the source and relay (S-R), and relay to destination (R-D), can be calculated as follows:

$$L_r = smr_c, \tag{4}$$

where s, m, and r_c denote the symbol rate, symbol density, and coding rate, respectively. s can be calculated as follows.

$$s = \frac{W}{1 + B}, \tag{5}$$

where W and B represent the roll-off rate and bandwidth, respectively. Next, m is calculated as:

$$m = \log_2 M, \tag{6}$$

where M denotes the number of waveforms on which the binary digits are mapped. Additionally, the transmission time of header T_h, and data from source to relay T_{S-R} and relay to destination T_{R-D} for each channel is calculated based on the data size S_d. In other words,

$$T_h = \frac{S_d}{L_r} \tag{7}$$

IV. Problem Statement

This section discusses the traditional optimization-based approaches employed in the literature and the challenges related to optimal channel selection in dynamic network conditions in the multi-band relay system, in a resource-constrained manner. A communication method comparable to that of [33] recognizes the channel selection problem in multi-band systems as Mixed Integer Non-Linear Programming (MINLP) optimization problem. This communication architecture computes such a problem by relaxing the NP-hard problem into Master Problems (MPs) and forming a column generation-based procedure to determine the MPs in the multi-radio base stations, considering they have adequate computational resources. However, a centralized server requires substantial processing power and memory or a powerful Software Defined Network (SDN) controller in order to solve such an optimization problem. Therefore, to address this issue, the computations should be performed locally, in a light-weight manner at the resource-constrained relay node. Hence, our system model in (Section III) took into account such resource-constrained relay nodes that are unable to solve the optimization problem locally in dynamic network conditions such as varying network traffic, data size, the various distances among the relay nodes, etc. The problem can be defined as utilizing a string of past data for T_w time steps, which can be represented as, $X_T = \{x\}_{T-(T_w-1)}^{T}$, based on measures like SINR, SNR, CSI or RSSI, in order to predict the next few time steps P_w, i.e., $Y_T = \{y\}_{T+1}^{T+P_w}$. Such inferencing task for channel quality prediction is expected to be performed in the resource-constrained relay nodes averting the heavy computations. Thus, the channel assignment time can be considerably reduced along with the improvement of throughput for the whole data transmission architecture as potential packet loss at the relay node can be averted. In the following section,

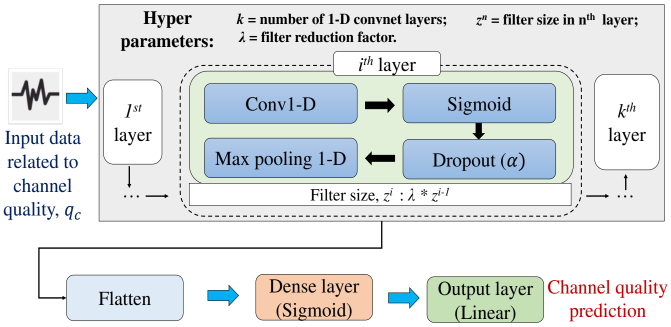


Fig. 2: Proposed CNN-based training and inference model.

we fathom a sustainable deep learning-based solution to the problem mentioned above.

Algorithm 1: *Predict Channel Quality (Features).*

Input : Various features (f) of signal strength with time

Output: Quality of a particular channel

1. Choose the best values for all the hyperparameters (i.e., k, S_F and Ω), based on experimental results from the considered corresponding values set
2. Scale and pass f through input layer and forward to hidden layers
3. Forward the sum of hidden layers through Ω followed by sub-sampling operation
4. Extract new features in the convolutional layers and forward to fully connected layer
5. Predict q_C in output layer

V. Proposed Deep Learning-Based Algorithm

This section presents a deep learning-based approach as a solution to the problem stated in Section IV. In this approach, the model is first trained in a centralized network with available traffic datasets containing channel conditions and other network signal measures. For various bands, the trained model in intended to correctly predict the channel quality (q_C) in advance. At each relay node, the pre-trained channel inference model is then transported. The proposed deep learning-based CNN model is depicted in Fig. 2. Here, the number of hidden layers used to construct the model is represented by k. Each hidden layer consists of a 1-D (one-dimensional) convolution layer, a regularization layer (i.e., dropout), and a 1D Max pooling layer. The 1-D convolution layers have an initial filter size of S_F in the first layer and Sigmoid as activation function, Ω [17]. We have used a dropout layer as the regularization method, with a rate of α to avoid overfitting during the training phase [34]. Afterward, the 1-D max pooling layer is added, which reduces the feature size and decreases the computational cost requirement of the model [35]. The channel prediction model utilizes the past signal data in a sliding window of T_w size in order to predict the channel quality of the next P_w steps. The window is slid further P_w steps in order to predict the channel quality of the next time frame. Two strategies are exploited by the model in order to forecast the channel quality, which are: (i) $controlled\ prediction$, and $ii)$ $smart\ prediction$. The $controlled\ prediction$ strategy operates by choosing the actual signal strength features as the input of the deep-learning model rather than the newly predicted value. A speedier convergence can be achieved with this strategy; however, due to the lack of accessibility of the previous data during prediction in the test dataset, the model experiences difficulties with generalization. Alternatively, the $smart\ prediction$ strategy offers a prediction of future time steps based on the newly predicted values. Thus, the prediction error can be minimized through this online learning. To measure the performance of the model in the test window, we have employed Root Mean Square Error (RMSE) [36] as an error measurement method to find the difference between the predicted value (V_p) and the actual value (V_a) as calculated below employing Eq. 8.

$$RMSE = \sqrt{\frac{1}{P_w}\sum_{i=1}^{P_w}(V_a - V_p)^2}. \tag{8}$$

Algorithm 2: Predictive channel assignment of each relay node.

Output: Optimal channel to be assigned to outgoing link of the relay node.

for *Each band* **do**
  for *All available channels* **do**
    $\mathbf{Q_C} +=$ Invoke algo. 1 to obtain q_C
  end
  Sort values in $\mathbf{Q_C}$ of each band from best to worst
  The coding rate of best channel r_c is collected from MCS table (Table I)
  Compute Link rate L_r using Eq. (4)
  Calculate T_{S-R}, T_h and T_{R-D} using Eq. (7)
end
while $T_{S-R} > T_{R-D} + T_h$ **do**
  Select $min(T_h + T_{R-D} - T_{S-R})$ such that $(T_h + T_{R-D} - T_{S-R}) > 0$
end

The deep learning-based CNN model to predict the quality of the channel is described in Algorithm 1. The signal strength parameters such as band-specific channel quality measurements of the channels in the training window T_w (e.g., SINR measured by Eq. 1 or others such as RSSI, CSI) along with other features such as energy information, packet loss, throughput, delay are passed to the deep learning-based model. The best value for hyperparameters, such as the number of hidden layers (k), activation function (Ω), and size of the filter (S_F) are chosen after hyperparameter tuning. The features are passed through input layers to the hidden layers where a sequence of convolutions is performed. TheF sum of the convolutions is forwarded through activation function and a sub-sampling operation is performed after that. New features are extracted from the layers and the features are forwarded to fully connected layer where the output, i.e., the channel quality

TABLE I: Considered Modulation and Coding Scheme (MCS) [13].

SINR	Modulation Scheme	m	r_c
SINR ≤ 2	No Tx	0	0
2 <SINR $\leq$ 5	BPSK	1	1/2
5 <SINR $\leq$ 9	QPSK	2	1/2
9 <SINR $\leq$ 11	QPSK	2	3/4
11 <SINR $\leq$ 15	16QAM	4	1/2
15 <SINR $\leq$ 18	16QAM	4	2/3
18 <SINR $\leq$ 20	16QAM	4	3/4
20 <SINR $\leq$ 25	64QAM	6	2/3
25 <SINR	64QAM	6	3/4

(q_C) is predicted for prediction window P_w. The channel quality is then used to perform scheduling using the optimal channel as shown in Algorithm 2. A vector of channel quality $\mathbf{Q_C}$ for all the channels in each band is predicted using the adopted CNN model. The vector elements are then sorted according to their respective channel quality values, and thus, the best channel is selected. Subsequently, the selected channel is used to find the coding rate r_c from MCS table (Table I). The link rate and transmission time of the channels are then computed as per Eq. 4 and Eq. 7, respectively. The channel where the transmission time of source to relay is larger than that of relay to destination is thus selected in advance. If the condition is not satisfied, the algorithm will look for another channel that fulfills the specified condition.

VI. Algorithmic Analysis

In this section, we investigate the algorithm's computational complexity and the time cost to run the proposed deep learning-based channel quality estimator for multiple bands. The analysis essentially centers around the algorithm complexity in the training phase and running phase via calculating the frequency of each operation (e.g., addition, subtraction, multiplication, division, and square root, etc.). The time cost of every procedure is denoted by ADD, SUB, MUL, DIV, and SQRT to precisely express the complexity. We analyze the training and prediction steps' complexity of the proposed model in terms of the number of different operations required by various steps of the algorithm. The outcomes from the complexity analysis are further explored in the performance evaluation section (Sec. VIII) for conducting a numerical analysis of the proposed model to manifest results in terms of processing time, throughput, and memory consumption to demonstrate the proposed model's applicability in the IoT environment.

A. Pre-processing Phase

Before starting the training phase, we scaled the dataset using the standardization technique to normalize the feature values in a particular range and it also helps in speeding up the calculations in the algorithm as mentioned in the steps 1-2 of Algorithm 1. Each i_{th} feature from input feature f is passed to the training phase after applying the standardization formula denoted by Eq. 9,

$$x_i^{'} = \frac{(x_i - \bar{x_i})}{\sigma} \tag{9}$$

$$\sigma = \sqrt{\frac{\sum(x_i - \bar{x_i})^2}{N_{x_i}}} \tag{10}$$

Here, x_i is the feature vector and $\bar{x_i}$, σ (expressed in Eq. 10) are the mean and the standard deviation of the x_i, respectively. For each i_{th} feature x_i, if we consider the length of the feature vector as N_{x_i}, then to compute the mean $\bar{x_i}$ requires $(N_{x_i} - 1)$ ADD and one DIV operations. Then, to calculate σ from Eq. 10, it requires $(N_{x_i} - 1)$ ADD, N_{x_i} SUB, one DIV, one MUL, and one SQRT operations. Finally for computing the scaled values employing Eq. 9, one DIV and $(N_{x_i} - 1)$ SUB operations are necessary. Since the scaling is performed for the number of features in the input vector denoted as f_n, the overall computation complexity of the pre-processing process for all the features can be expressed in terms of $(2 * f_n * (N_{x_i} - 1))$ ADD, $(f_n * (2N_{x_i} - 1))$ SUB, $(3f_n)$ DIV, (f_n) MUL, and (f_n) SQRT operations.

B. Training Phase

The main purpose of the training phase is to harness previous data regarding the channels' condition and predict future channel quality for P_w steps. In this phase, the proposed CNN model's computation complexity is analyzed during the learning or training period, referring to the steps 3-5 of Algorithm 1. In the training phase, the model gets trained for T_w time window, and later in the prediction phase, the trained model is used to estimate the channels' conditions. For each i_{th} convolution layer having the number of neurons i_n, filter size S_F and activation function Ω, the computational complexity of i_{th} convolution layer can be expressed as Eq. 11,

$$O_{conv} = (i_n * (S_F * (f_n - (S_F - 1))) - (f_n * (S_F - 1)) + i_n))ADD, (i_n * (S_F * (f_n - (S_F - 1))))MUL, (i_n)DIV. \tag{11}$$

After the convolutional layers, the model will have fully connected layers. If we assume the number of node in j_{th} fully connected or dense layer to be j_n, the computation complexity will be, O_{dl} = $(j_n * y_i * (f_i - 1))$ ADD, $(j_n * y_i * f_i)$ MUL. Step 5 of the Algorithm 1 will return the predictions of the training phase. Since, according to the steps 1-4 of Algorithm 2 considering the number of bands = B_n, number of channels in each band = C_n and the number of layers in the model to be k can be written as Eq. 12,

$$O_{training} = (B_n * C_n * k * (O_{conv} + O_{dl})) \tag{12}$$

Hence, for each type of operation, the complexity will be increasing by k (number of layers). Furthermore, for B_n bands and C_n channels, the required number of operations will be multipled by C_n and B_n.

C. Running Phase

After getting the predicted channel quality from all channels of all the bands, step 5 of Algorithm 2 will sort the predicted channel quality (q_c) for each band. Considering the sorting algorithm to be quick-sort of n number of channel quality of a band, the required operations for the sorting will be $3n$ ADD, $(n-1)$ SUB. In steps 6-7, for all the bands (B_n) the best channel's coding rate will be selected, and the link rate L_r will be calculated. Therefore, these steps will require B_n ADD, B_n MUL and B_n DIV. The following step 8 of the algorithm will determine the values of T_{S-R}, T_{R-D}, and T_h, which will result in $3 * B_n$ DIV. In the last few steps of the algorithm, specifically from step 10-12, $min(T_h + T_{R-D} - T_{S-R})$ will be determined. Considering the loop in these steps will run for m number of times, for B_n bands, it will require (B_n*(2*m ADD, m SUB)) operations to be performed. Therefore, in the running phase of the model, the computation complexity required by the proposed model can be written as Eq. 13,

$$O_{running} = (3n * B_n + B_n + B_n * 2 * m)ADD + ((B_n * (n-1)) + B_n * m)SUB + (B_n)MUL, (4 * B_n)DIV \tag{13}$$

VII. An Illustrative Example of the Proposed Model

In this section, we demonstrate a simple example for illustration of how our proposed algorithm operates, and feature size evolves in diverse layers of the model. Fig. 3 represents an illustrative example for this purpose. In this example, we have considered a shallow-CNN model consisting of a single layer. We assume 14 features, for instance, to pass through the layers of the proposed model. The number of filters is set to 150, and the kernel size is 3. Therefore, the overall filter size (S_F) is 150x3.

Firstly, the input will be passed to the convolution layer, and the filters will try to find out insights and patterns in the data by performing the sum of element-wise multiplication of the input and filter. We assume striding value of 1 for the filter, and hence after each filter is applied on the dataset, the output size is going to be reduced to 12 from the initial size of 14. Therefore, for 150 filters, the output size is going to be 150x12. Now let us determine how the filters convolve over the input vector and update the values. Assuming the input, f = [-8.41, 5.90, 7.35, 3.13 , -2.41 , -1.60, 5.92, 4.91, 22.7, 0, 15.92, 0, -8.39, -4.97] and the filter 1, τ_1=[0.76, 0.08, 0.60]. The results after the filter are applied on the input vector can be denoted as f_{τ_i}, and the first step of f_{τ_i} is determined as $(0.76 * -8.41 + 0.08 * 5.90 + 0.60 * 7.35) = -1.48$. Subsequently, the calculated f_{τ_i} will be passed to the sigmoid activation function (Ω) and hence, $\Omega(f_{\tau_i})$=0.190. This procedure will be applied at each step using a striding value of 1 on the whole input sequence, which will result in an output of size 12 for a single filter. Consequently, for 150 filters, the final output shape in this step will be 150x12.

The dropout phase is a regularization technique of the proposed method will not change the shape of the output as it will only ignore a few nodes in order to avoid overfitting. The following step is the max pooling with a pool size of 2 and striding of 2. Therefore, the maximum value will be selected from the pairwise comparison at each step. The size of the output of this step is 6 for each input from a particular filter. Hence, for all 150 filters, the output size will be 150x6. After the max-pooling layer, at the penultimate stage, the flatten layer will be applied to the feature set, which will reconstruct a multi-dimensional matrix of features into a vector that can be fed into a fully connected dense layer of the neural network classifier. Therefore, the 150x6 sized output of the max-pooling layer will be converted into a 900 length vector. Lastly, for determining the output scalar, the output vector of the flatten layer will go through a fully connected layer. Summation of element-wise multiplication of the output vector of the flatten layer and the weight vector will be passed through the activation function (Ω) to get the final output value (1.015), which indicates the channel quality of the corresponding band. This procedure is conducted in terms of each band to obtain the predicted value for channel quality.

In this example case, we have considered a shallow-CNN of our proposed model, where we assumed the number of convolution layers to be one to perceive how the input vector transforms for different layers. The deeper architecture of the proposed model will employ the output of the preceding layer as input to the subsequent layer and will contain similar calculations in each of the layers.

VIII. Performance Evaluation

In this section, we demonstrate the experimental results to evaluate the performance of the proposed deep learning-based proactive channel assignment to the multi-band relay nodes. A brief description of the datasets used to feed to our proposed deep learning model is first presented, followed by experimental results and discussion.

A. Data Preparation

We have employed three datasets, denoted by DS1, DS2, and DS3, respectively, for performing the channel quality prediction. Brief details of the datasets are given below.

1) DS1: This dataset consists of RSSI data obtained with a mobile robot in two environments: indoor and outdoor [37]. RSSI data of five wireless receivers in indoor conditions are collected using the a youBot mobile robot. For the outdoor environment, data for signal strength and location are collected from a mobile robot in a semi-outdoor environment.
2) DS2: This dataset contains signal strength measurement of a Zigbee-based wireless network [38]. It contains around 8000 data samples between a Tx-Rx pair is placed over a distance of 10 to 35 meters, and it consists of information of energy, throughput, delay, and loss of data transfer from source to destination nodes.
3) DS3: This particular dataset describes an extensive set of traces that represent the radio channel conditions between the base station and the end-user device [39]. The records were employed to design and simulate a mobile networking environment practically. The LTE signal strength values, Reference Signal Received Power (RSRP), SNR, and RSSI. All of these features contribute a comprehensive insight into the channel condition. Each of the annotated traces is of different environments (e.g,.

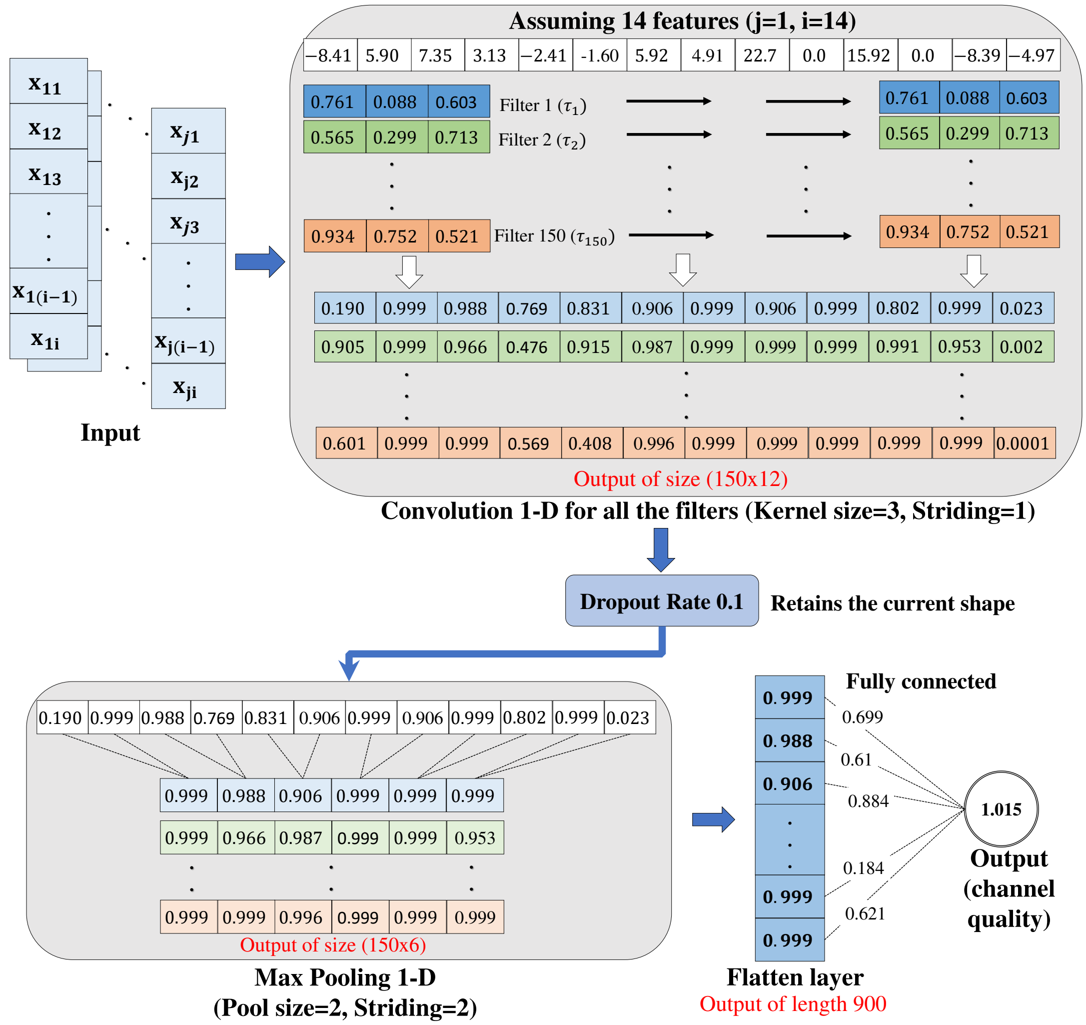


Fig. 3: An illustration of how the data size evolves in the proposed CNN model with single layer.

bus, train, pedestrian, static and train) and at different speeds.

B. Results and Discussion

We applied Linear Regression (LR), Auto Regression (AR), and a Artificial Neural Network (ANN) [28] to compare the performance of our proposed CNN-based controlled and smart channel prediction methods. The training and prediction window for both deep and shallow architectures of the ANN and CNN models are selected based on tuning of these parameters. The selected tuned values are: training window, T_w = 200, prediction window, P_w = 50% of the training window size (i.e. 100), and the number of hidden layers in the DL models, $k \in \{1, 4\}$. Note that these parameters are selected based on the results of manual hyperparameter tuning without resorting to grid search.

Tables II and III list the performance of ANN, and CNN compared to the LR and AR technique, which is obtained from DS1 (indoor and outdoor conditions). The results are represented in terms of the average RMSE across all the channels. We evaluated the performance of the deep learning models in both controlled and smart prediction scenarios by employing a shallow architecture (a single hidden layer), and a deep architecture (four hidden layers). In all the cases, the deep learning-based prediction performance exhibits better performance in contrast with that of AR and LR. Between controlled and smart prediction schemes, the model demonstrated better estimation performance in terms of the smart prediction than its controlled prediction counterpart. This implies that the deep learning-based techniques were able to explore the search space more robustly in contrast with the baseline techniques. Therefore, they were able to generalize the diverse channel conditions,

TABLE II: Comparison of average RMSE values across all time-steps for LR, AR, ANN, and CNN-based methods for DS1-indoor environment.

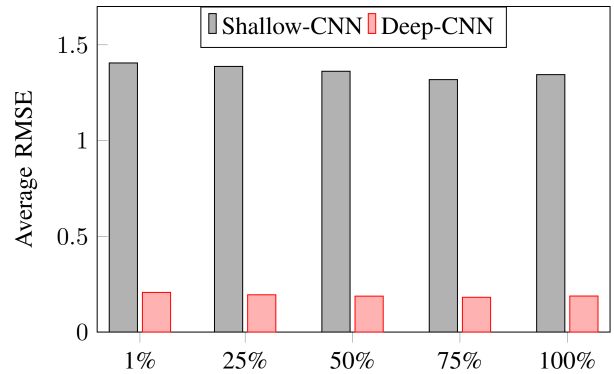
Method	Controlled Prediction	Smart Prediction
LR	6.933	1.441
AR	25.715	1.386
Shallow-ANN	0.726	0.392
Deep-ANN	0.725	0.414
Shallow-CNN	0.595	**0.371**
Deep-CNN	0.643	0.388

TABLE III: Comparison of average RMSE values across all time-steps for LR, AR, ANN, and CNN-based methods for DS1-outdoor environment.

Method	Controlled Prediction	Smart Prediction
LR	2.753	34.656
AR	3.198	5.860
Shallow-ANN	1.210	1.308
Deep-ANN	1.144	1.297
Shallow-CNN	1.057	**0.964**
Deep-CNN	0.971	0.994

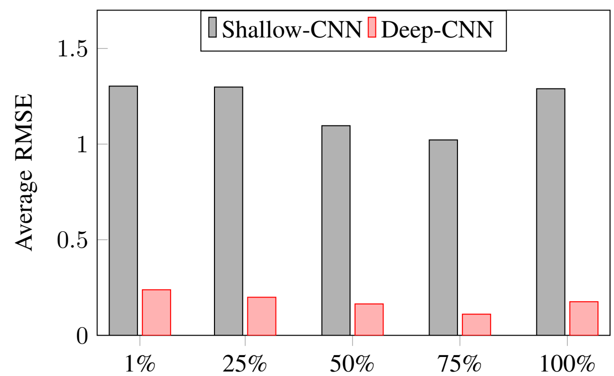
especially in the smart prediction strategy. Hence, we adopted the smart prediction scheme for the next experimental setups to further optimize the hyperparameters and evaluate the results. In the case of the outdoor environment, the proposed CNN-based method can outperform the baseline approaches by a significant margin. This massive performance gap between CNN and baseline techniques demonstrates that in noisy outdoor environments, the traditional algorithms are unable to predict channel conditions accurately compared to deep-learning-based techniques. Thus, it may be concluded that, although the baseline techniques perform well in a single-hop, single-radio wireless network for predicting channel conditions [30], are not appropriate for real-time channel prediction in multi-band relay networks. Furthermore, the performance comparison between ANN, and CNN reveals that CNN is more consistent and stable than the other neural networks. Hence, we have not considered ANN for further experiments, and elected one and four-layer architectures of CNN, referred to as shallow and deep-CNN, respectively, for further analysis.

1) Hyperparameter Tuning: To choose the best hyperparameters of the proposed CNN-based models, we have conducted a systematic investigation. Firstly, as a part of determining the best prediction window-size, we tuned the prediction window-size, P_w, by adopting varying ratios with respect to the training window T_w. Table. IV illustrates the RMSE values for various distances from dataset DS2 for P_w values varied among 25%, 50%, and 75% of T_w, respectively. In all these cases, the baseline techniques are significantly outperformed by the proposed CNN-based approach. As the distance between the sender and the receiver nodes increased from 10 to 35 meters, the estimation error (i.e., RMSE) of LR showed a dramatically progressing trend. Therefore, this implies that the baseline techniques are not able to interpret channel quality in diverse situations with a growing distance as much as the proposed deep-learning-based technique. Hence, the CNN-based approach can perform the prediction task with much less error compared to that incurred in AR and LR. Among these three proportions, both the shallow and deep-CNN models manifest the best average RMSE values of 0.71 and 0.65, respectively, when 50% of the training window T_w is taken into consideration for the prediction window size (P_w). Initially, the training window size (T_w) is set to 200 time steps. Hence, after the hyperparameter tuning, the selected prediction window size (P_w) is 100 time steps. Among the baseline techniques, particularly, the AR suffers drastically for all the considered distances. Therefore, we considered P_w to be half (50%) of the training window T_w to accurately predict



Filter size (S_F) with respect to training window size (T_w)

Fig. 4: Comparison of CNN with respect to filter size (for DS1-indoor environment).



Filter size (S_F) with respect to training window size (T_w)

Fig. 5: Comparison of CNN with respect to filter size (for DS1-outdoor environment).

the most suitable bands and channels in order to transmit IoT data at a greater throughput in the multi-hop relay system. The trade-off in choosing the training and prediction window is that the training window should not be too large to reduce the time-delay to train the model. The prediction window size should not be more than 50% of the training window size to decrease prediction error, and it should not be too small in size as it will slow down the prediction process.

Next, as a part of the hyperparameter tuning, we tuned the filter size (S_F) of the 1-D convolutional layers with respect to the training window (T_w) employing DS1. Figs. 4 and 5 demonstrate the outcomes in terms of average RMSE for different size of filter. In both cases, the least error is recorded when the initial filter size (S_F) is equal to 75% of the training window (T_w). Therefore, we elected this parameter to be the starting filter size of our proposed model. The filter size reduction factor, α, is set to 0.5 as we have decreased filter size by 50% each time for the deeper layers compared to the previous layer. Furthermore, to pick the most suitable activation function (Ω) for the proposed deep learning-based CNN model, we have performed manual hyperparameter tuning from a set of proper activation functions (i.e., Sigmoid, Tanh, ReLU, and SELU) [40]. Fig. 6 portrays the results for

TABLE IV: Comparison of proposed shallow and deep-CNN models with baseline techniques for different prediction window (P_w) sizes with respect to training window (T_w) using DS2.

Distance (meters)	**RMSE of different methods for Size of (P_w) with respect to (T_w)**											
	25%				**50%**				**75%**			
	LR	AR	Shallow-CNN	Deep-CNN	LR	AR	Shallow-CNN	Deep-CNN	LR	AR	Shallow-CNN	Deep-CNN
10	2.35	58.97	1.22	0.99	1.76	61.30	0.78	0.71	1.91	2.83	0.86	1.68
15	2.13	79.63	0.82	0.77	2.35	63.79	0.90	0.64	2.02	52.23	0.61	1.29
20	2.63	20.95	1.85	1.00	1.98	56.21	0.72	0.66	1.74	2.28	0.75	1.49
25	2.63	56.37	0.45	0.80	2.56	51.61	0.46	0.55	2.16	59.91	0.72	0.75
30	2.95	73.08	0.61	1.05	1.99	75.81	0.57	0.55	2.33	30.24	0.69	0.92
35	3.16	65.45	1.93	0.99	2.21	56.24	0.86	0.81	2.73	7.93	0.76	1.69
Mean RMSE	2.64	59.08	1.15	0.93	2.14	60.82	**0.71**	**0.65**	2.14	25.90	0.73	1.30

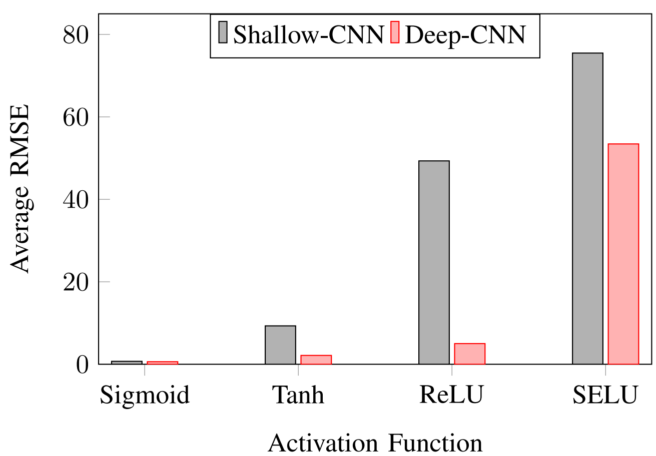


Fig. 6: Comparison of different activation functions for the proposed CNN model using DS2.

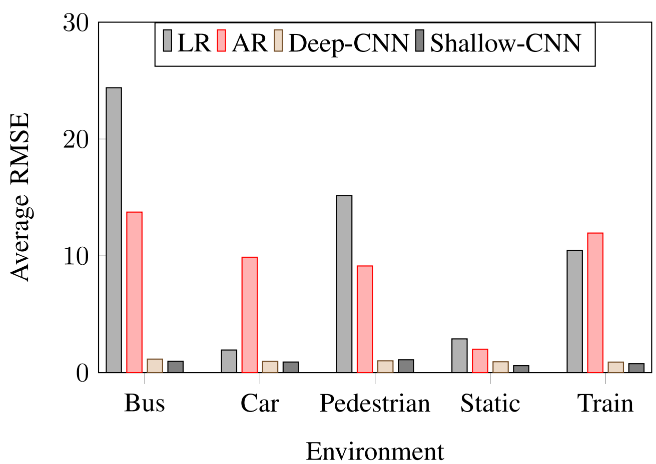


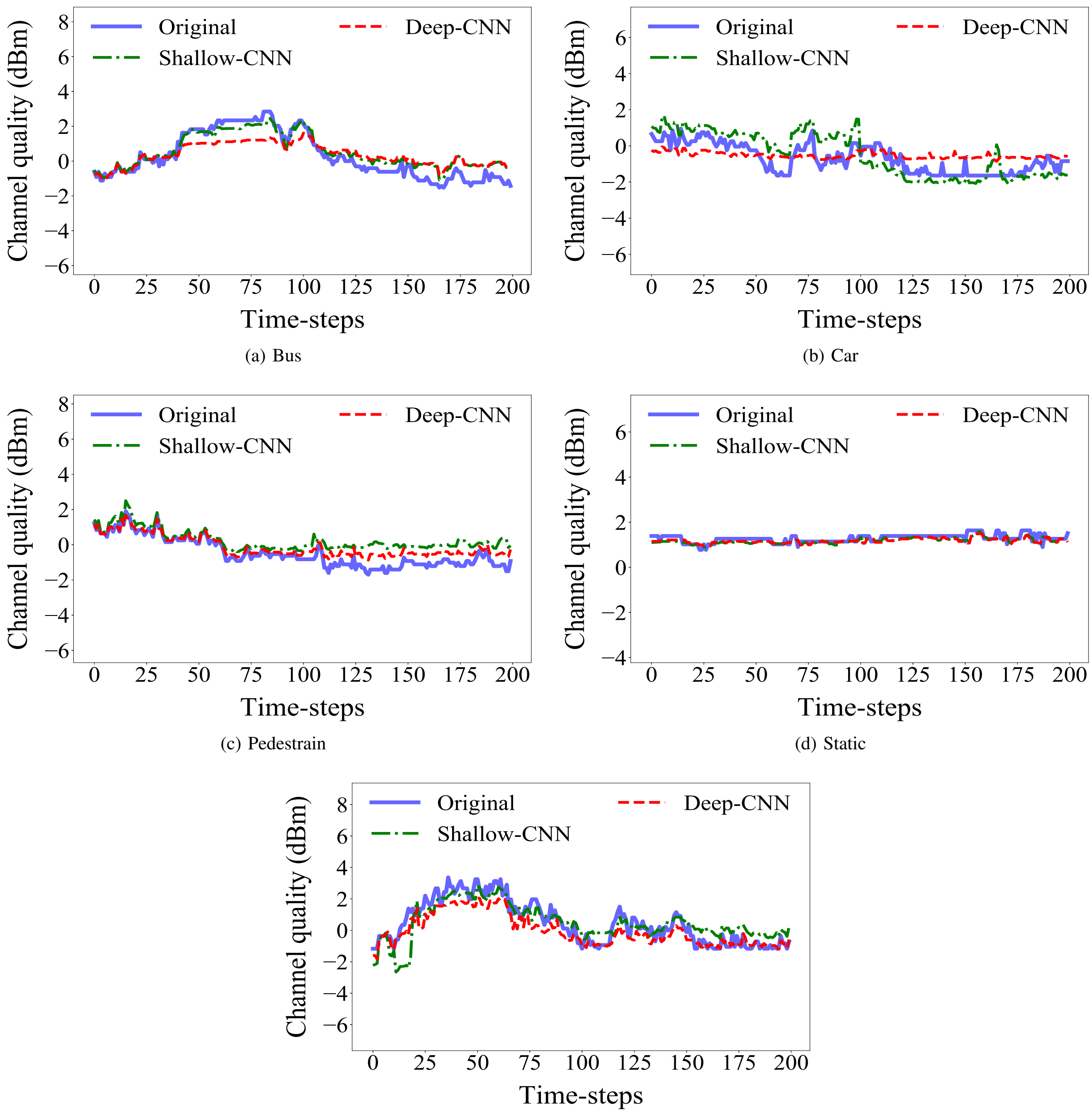
Fig. 7: Comparison between CNN and baseline techniques for different environments of DS3.

the different activation functions in terms of average RMSE noted using DS2. The best performance is evident when the Sigmoid activation function is used in all the layers of the proposed shallow and deep architecture of CNN. Therefore, this verifies the selection of Sigmoid activation function for the channel quality prediction task in the multi-band relay communication system.

After the hyperparameter tuning experimental phase, we evaluated the model with another distinct dataset (DS3). Fig. 7 depicts the comparison of error (average RMSE) between deep-CNN, shallow-CNN, AR, and LR in varying environments of DS3. The figure demonstrates that shallow-CNN outperformed deep-CNN by a slight margin and LR by a considerable margin. Therefore, this result verifies the proposed CNN-based model's acceptability for being selected as a channel quality estimator for real-time channel prediction in multi-band relay networks.

We also plotted the predicted channel conditions over time and compared with the original values for visual comparison. Fig. 8 manifests a fragment of a few example cases of the actual and predicted q_C values for varying conditions from DS3 employing smart predictive strategy. The experimental outcomes exhibit that the proposed CNN model's performance is very much indistinguishable to the original signal for most time steps. In the case of static and pedestrian, the overall prediction performance of the CNN models is more robust than other situations. In terms of the other environmental conditions, also the models' prediction is representative of the actual channel quality and is able to find the trend of channel quality.

2) Numerical Analysis: As the proposed CNN-based method's prediction performance manifested encouraging results, we conducted a numerical analysis of the model in terms of memory consumption, processing time delay, and throughput. Fig. 9 manifests the processing delay for the model, while Fig. 10 illustrates the model's memory consumption at each time-step with respect to the node's overall capability. In this case, the trade-off is: although the predictive performance of the proposed DL-based approach is significantly greater than that of the baseline techniques, it consumes more memory and requires a higher time-dealy. However, the additional processing delay can be considered negligible by considering the prediction efficiency of the proposed technique. Admittedly, the proposed model will predict all the channels' conditions over many time-steps ahead of time, considerably minimizing the overall communication delay. Consequently, the acquired experimental outcomes illustrate that the proposed DL-based CNN model is suitable for efficient channel prediction and proactive channel distribution in a multi-band relay network system. Fig. 11 depicts the throughput comparison for transmitting data over the heterogeneous band relay-based network.



(a) Bus

(b) Car

(c) Pedestrain

(d) Static

(e) Train

Fig. 8: CNN-based prediction methods compared to the original channel quality for different environments of DS3.

We have considered an average data generation rate of 1MB/s from the nodes. A latency of 1 second is considered because of poor channel selection from the baseline techniques (i.e., LR, AR) as the predictive performances for these are worse than the proposed CNN model. We estimated the CNN-based model's latency considering the difference of error for poor channel selection between CNN and the baseline techniques (in percentage). Afterward, we determined the throughput by considering the processing delay and the possible delay in choosing a lousy quality channel. Even though AR and LR's processing time is less than that of the proposed DL-based CNN model, the throughput is approximated to be considerably lower. Hence, it unveils that traditional baseline techniques cannot be as effective as the proposed CNN in dynamic network conditions in multi-band relay networks, and possible packet loss may occur for specific relay nodes where the resources are inadequate.

The experimental results clearly demonstrate that the proposed shallow-CNN model emerges as the most viable lightweight, predictive channel inference model to deploy at the resource-constrained relay node for enhanced spectral efficiency and throughput for offloading IoT traffic. The proposed DL-based model's generalization capability in terms of prediction performance was evaluated on three different datasets from diverse setup. Furthermore, the prediction performance, as well as memory consumption, processing time, and through-

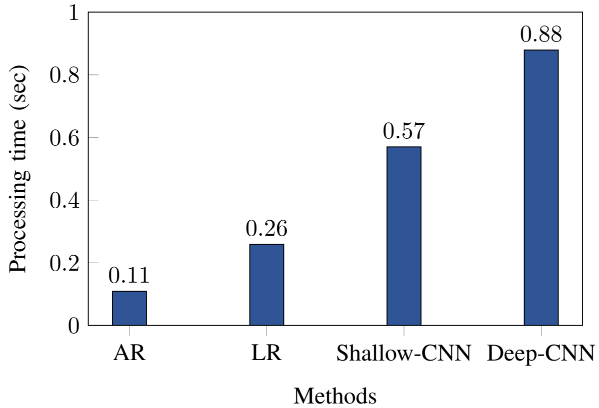


Fig. 9: Processing time of different methods.

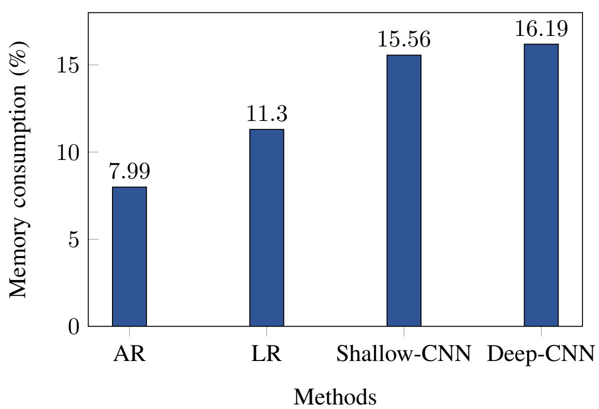


Fig. 10: Memory consumption of different methods.

put, were compared with popular AI-based prediction models. The encouraging outcomes illustrate that the proposed model performed efficiently in all three datasets. Hence, the model can be generalized and utilized for other varieties of wireless network setups along with the IoT environment.

IX. Conclusion

This paper focuses on the prevalent resource-constrained multi-band relay nodes for offloading massive IoT traffic in next generation networks and the measures taken to overcome the barriers of most appropriate channel selection for efficient data transmission. The transmission process involves sending the data header first, followed by the rest of the data on one band, and simultaneously forwarding to the next node through another band. In order to select a channel, a lightweight deep-learning technique is proposed that will accurately select the best channel to transmit and receive data based on its quality. To construct the Convolutional Neural Network (CNN) model, authentic datasets were used for both smart and controlled prediction strategies to forecast the channel quality. Afterwards, the channel quality measures are utilized by a scheduling algorithm in order to determine the coding and modulation rates to forward the data to the next node. The model was assessed by comparing it with other predictive algorithms such as LR, AR, and ANN. The performance of these algorithms at the relay node was comparatively inferior

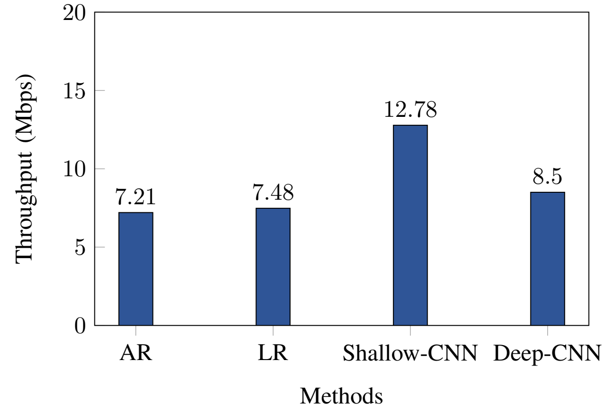


Fig. 11: Throughput of different methods considering the additional processing delay due to possible poor channel selection.

to CNN based channel prediction algorithm with respect to throughput, memory consumption, and processing delay. In addition to that, the results of deep-CNN and shallow-CNN was analogous, however, the processing delay and power consumption of shallow-CNN was relatively lower, thus, improving throughput. Hence, the propositioned shallow-CNN model was nominated as the most feasible architecture to predict channel state of a resource-constrained relay node.

Acknowledgment

The work has been partly supported by the New Frontiers in Research Fund 2019 Exploration (NFRFE-2019-00313) and Natural Sciences and Engineering Research Council of Canada (NSERC) (RGPIN-2020-06260).

References

[1] A. Arfaoui, S. Cherkaoui, A. Kribeche, and S. M. Senouci, "Context-Aware Adaptive Remote Access for IoT Applications," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 786–799, Jan. 2020 doi: 10.1109/JIOT.2019.2953144.

[2] S. Oteafy and H. Hassanein, "Leveraging Tactile Internet cognizance and operation via IoT and Edge technologies," *Proceedings of the IEEE*, vol. 107, no. 2, pp. 364–375, Feb. 2019, doi: 10.1109/JPROC.2018.2873577.

[3] Y. Han, B. D. Rao, and J. Lee, "Massive Uncoordinated Access With Massive MIMO: A Dictionary Learning Approach," *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 1320–1332, Feb. 2020, doi: 10.1109/TWC.2019.2952843.

[4] J. Lianghai, B. Han, M. Liu, and H. Schotten, "Applying Device-to-Device Communication to Enhance IoT Services," *IEEE Communications Standards Magazine*, vol. 1, no. 2, pp. 85–91, Jul. 2017, doi: 10.1109/MCOMSTD.2017.1700031.

[5] X. Chen, D. W. K. Ng, W. Yu, E. G. Larsson, N. Al-Dhahir, and R. Schober, "Massive Access for 5G and Beyond," *arXiv preprint arXiv:2002.03491*, Feb. 2020.

[6] A. Moubayed, A. Shami and H. Lutfiyya, "Wireless Resource Virtualization With Device-to-Device Communication Underlaying LTE Network," in IEEE Transactions on Broadcasting, vol. 61, no. 4, pp. 734-740, Dec. 2015, doi: 10.1109/TBC.2015.2492458.

[7] A. Ettefagh, M. Kuhn, I. Hammerstrom, and A. Wittneben, "On the Range Performance of Decode-And-Forward Relays in IEEE 802.11 WLANs," in *IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications*, Helsinki, Finland, 11–14 Sep. 2006, doi: 10.1109/PIMRC.2006.254478.

[8] P. Zhang, J. Yuan, J. Chen, J. Wang, and J. Yang, "Analyzing Amplify-and-Forward and Decode-and-Forward Cooperative Strategies in Wyner's Channel Model," in *2009 IEEE Wireless Communications and Networking Conference*, Budapest, Hungary, 5–8 Apr. 2009, doi: 10.1109/WCNC.2009.4917597.

[9] A. Khina, O. Ordentlich, U. Erez, Y. Kochman, and G. W. Wornell, "Decode-and-forward for the Gaussian relay channel via standard AWGN coding and decoding," in *2012 IEEE Information Theory Workshop*, Lausanne, Switzerland, 3–7 Sep. 2012, pp. 457–461, doi: 10.1109/ITW.2012.6404714.

[10] N. Egashira et al., "Integrated Synchronization Scheme for WLAN Systems Employing Multiband Simultaneous Transmission," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, San Francisco, CA, USA, 19–22 Mar. 2017, doi: 10.1109/WCNC.2017.7925799.

[11] F. Gholami, H. Meghdadi, and A. Shahzadi, "Throughput Analysis for Decode-and-Forward Relaying Protocol with Wireless Energy Harvesting and Information Processing," in *2018 Internet of Things, Embedded Systems and Communications (IINTEC)*, Hamammet, Tunisia, 20–21 Dec. 2018, pp. 132–135, doi: 10.1109/IINTEC.2018.8695285.

[12] A. Chaaban and A. Sezgin, "Multi-Hop Relaying: An End-to-End Delay Analysis," *IEEE Transactions on Wireless Communications*, vol. 15, no. 4, pp. 2552–2561, Apr. 2016, doi: 10.1109/TWC.2015.2504528.

[13] Z. M. Fadlullah, Y. Kawamoto, H. Nishiyama, N. Kato, N. Egashira, K. Yano, and T. Kumagai, "Multi-hop wireless transmission in multiband WLAN systems: Proposal and future perspective," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 108–113, Feb. 2019, doi: 10.1109/MWC.2017.1700148.

[14] G. Cerar, H. Yetgin, M. Mohorčič, and C. Fortuna, "Machine Learning for Wireless Link Quality Estimation: A Survey," *arXiv preprint arXiv:1812.08856*, Nov. 2019.

[15] A. R. Abdellah, O. A. K. Mahmood, A. Paramonov, and A. Koucheryavy, "IoT traffic prediction using multi-step ahead prediction with neural network," in *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, Dublin, Ireland, 28–30 Oct. 2019, doi: 10.1109/ICUMT48472.2019.8970675.

[16] Y. Kim, P. Wang, and L. Mihaylova, "Structural Recurrent Neural Network for Traffic Speed Prediction," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, UK, 12–17 May 2019, pp. 5207–5211, doi: 10.1109/ICASSP.2019.8683670.

[17] Q. Mao, F. Hu, and Q. Hao, "Deep Learning for Intelligent Wireless Networks: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2595–2621, Jun. 2018, doi: 10.1109/COMST.2018.2846401.

[18] Y. Lin, X. Dai, L. Li, and F.-Y. Wang, "An Efficient Deep Reinforcement Learning Model for Urban Traffic Control," *arXiv preprint arXiv:1808.01876*, Aug. 2018.

[19] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Machine learning for wireless networks with artificial intelligence: A tutorial on neural networks," *arXiv preprint arXiv:1710.02913*, 2017.

[20] M. Nasri and M. Hamdi, "LTE QoS Parameters Prediction Using Multivariate Linear Regression Algorithm," in *2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, Paris, France, 19–21 Feb. 2019, pp. 145–150, doi: 10.1109/ICIN.2019.8685914.

[21] Y. Zhou and J. Li, "Research of Network Traffic Anomaly Detection Model Based on Multilevel Autoregression," in *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, Dalian, China, 19–20 Oct. 2019, pp. 380–384, doi: 10.1109/ICCSNT47585.2019.8962517.

[22] A. R. Abdellah, O. A. K. Mahmood, A. Paramonov, and A. Koucheryavy, "IoT traffic prediction using multi-step ahead prediction with neural network," in *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, Dublin, Ireland, 28–30 Oct. 2019, pp. 1–4, doi: 10.1109/ICUMT48472.2019.8970675.

[23] C. Parera, A. E. C. Redondi, M. Cesana, Q. Liao, and I. Malanchini, "Transfer Learning for Channel Quality Prediction," in *2019 IEEE International Symposium on Measurements Networking (M&N)*, Catania, Italy, 8–10 Jul. 2019, pp. 1–6, doi: 10.1109/IWMN.2019.8805017.

[24] M. Iqbal, M. Zahid, D. Habib, and L. John, "Efficient prediction of network traffic for real-time applications," *Journal of Computer Networks and Communications*, vol. 2019, pp. 1–11, Feb. 2019, Art. no. 4067135, doi: 10.1155/2019/4067135.

[25] L. Nie, X. Wang, L. Wan, S. Yu, H. Song, D. Jiang, and K. Zhang, "Network traffic prediction based on deep belief network and spatiotemporal compressive sensing in wireless mesh backbone networks," *Wireless Communications and Mobile Computing*, vol. 2018, Jan. 2018, Art. no. 1260860, doi: 10.1155/2018/1260860.

[26] T. Liu and A. E. Cerpa, "Temporal Adaptive Link Quality Prediction with Online Learning," *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 3, pp. 1–41, May 2014, doi: 10.1145/2594766.

[27] J. Wang et al., "Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach," in *IEEE International Conference on Computer Communications (INFOCOM'17)*, Atlanta, GA, USA, 1–4 May 2017, doi: 10.1109/INFOCOM.2017.8057090.

[28] L. Liu, B. Yin, S. Zhang, X. Cao, and Y. Cheng, "Deep Learning Meets Wireless Network Optimization: Identify Critical Links," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 1, pp. 167–180, Apr. 2018, doi: 10.1109/TNSE.2018.2827997.

[29] W. Rehan, S. Fischer, and M. Rehan, "Machine-Learning Based Channel Quality and Stability Estimation for Stream-Based Multichannel Wireless Sensor Networks," *Sensors (Basel)*, vol. 16, no. 9, Sep. 2016, Art. no. 1476, doi: 10.3390/s16091476.

[30] J. D. Herath, A. Seetharam, and A. Ramesh, "A Deep Learning Model for Wireless Channel Quality Prediction," in *IEEE International Conference on Communications (ICC)*, Shanghai, China, 20–24 May 2019, doi: 10.1109/ICC.2019.8761934.

[31] A. Hanyu et al., "Adaptive Frequency Band and Channel Selection for Simultaneous Receiving and Sending in Multiband Communication," *IEEE Wireless Communications Letters*, vol. 8, no. 2, pp. 460–463, Apr. 2019, doi: 10.1109/LWC.2018.2876258.

[32] H. Singh, J. Hsu, L. Verma, S. S. Lee, and C. Ngo, "Green operation of multi-band wireless LAN in 60 GHz and 2.4/5 GHz," in *2011 IEEE Consumer Communications and Networking Conference (CCNC)*, Las Vegas, NV, USA, 9–12 Jan. 2011, pp. 787–792, doi: 10.1109/CCNC.2011.5766599.

[33] M. Li, S. Salinas, P. Li, X. Huang, Y. Fang, and S. Glisic, "Optimal Scheduling for Multi-Radio Multi-Channel Multi-Hop Cognitive Cellular Networks," *IEEE Transactions on Mobile Computing*, vol. 14, no. 1, pp. 139–154, Jan. 2015, doi: 10.1109/TMC.2014.2314107.

[34] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research (JMLR)*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014.

[35] S. Sakib, M. M. Fouda, Z. M. Fadlullah, and N. Nasser, "Migrating Intelligence from Cloud to Ultra-Edge Smart IoT Sensor Based on Deep Learning: An Arrhythmia Monitoring Use-Case," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, Limassol, Cyprus, Jun. 15-19, 2020, pp. 595–600, doi: 10.1109/IWCMC48107.2020.9148134.

[36] K. Zhang, G. Chuai, W. Gao, X. Liu, S. Maimaiti, and Z. Si, "A new method for traffic forecasting in urban wireless communication network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, Mar. 2019, Art. no. 66, doi: 10.1186/s13638-019-1392-6.

[37] R. Parasuraman, S. Caccamo, F. Baberg, and P. Ogren, "CRAWDAD dataset kth/rss (v. 2016-01-05)," Jan. 2016, [Online]. Available: https://crawdad.org/kth/rss/20160105, doi: 10.15783/C7088F.

[38] S. Fu and Y. Zhang, "CRAWDAD dataset due/packet-delivery (v. 2015-04-01)," Apr. 2015, [Online]. Available: https://crawdad.org/due/packet-delivery/20150401, doi: 10.15783/C7NP4Z.

[39] B. Meixner, J. W. Kleinrouweler, and P. Cesar, "4G/LTE channel quality reference signal trace data set," in *Proceedings of the 9th ACM Multimedia Systems Conference (MMSys '18)*, Amsterdam, Netherlands, Jun. 2018, pp. 387–392, doi: 10.1145/3204949.3208132.

[40] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning," *arXiv preprint arXiv:1811.03378*, Nov. 2018.

BIOGRAPHIES

Sadman Sakib has been with the Department of Computer Science, Lakehead University, Thunder Bay, Ontario, Canada, as a graduate student since 2019. His research interests are in the field of intelligent computing/communication systems for mobile health (mHealth) applications for providing better health outcomes in northern Ontario and beyond. He was the winner of the best student's poster award of the Canadian Institutes of Health Research (CIHR) category in the graduate conference held at Lakehead University in 2020.

Tahrat Tazrin has been with the Department of Computer Science, Lakehead University, Thunder Bay, Ontario, Canada, as a graduate student since 2019. Her research interest in data analytics applications with a particular focus on developing logic-in-sensor EEG headsets.

Mostafa M. Fouda (SM'14) is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Idaho State University, ID, USA. He also holds the position of Associate Professor at Benha University, Egypt. He has served as an Assistant Professor with Tohoku University, Japan. He was a Postdoctoral Research Associate with Tennessee Technological University, USA. He received his B.S. (Hons.) and M.S. degrees in Electrical Engineering from Benha University, Egypt in 2002 and 2007, respectively, and his Ph.D. degree in Information Sciences from Tohoku University, Japan in 2011. His research interests include cyber security, machine learning, blockchain, IoT, 5G networks, smart healthcare, and smart grid communications. He has published over 30 papers in IEEE conference proceedings and journals. He has served on the technical committees of several IEEE conferences. He is also a reviewer in several IEEE Transactions/Magazines and an associate editor of IEEE Access. He is a Senior Member of IEEE.

Zubair Md. Fadlullah (SM'13) is currently an Associate Professor with the Computer Science Department, Lakehead University, and a Research Chair of the Thunder Bay Regional Health Research Institute (TBRHRI), Thunder Bay, Ontario, Canada. He was an Associate Professor at the Graduate School of Information Sciences (GSIS), Tohoku University, Japan, from 2017 to 2019. His main research interests are in the areas of emerging communication systems, such as 5G New Radio and beyond, deep learning applications on solving computer science and communication system problems, UAV based systems, smart health technology, cyber security, game theory, smart grid, and emerging communication systems. He received several best paper awards at conferences including IEEE/ACM IWCMC, IEEE GLOBECOM, and IEEE IC-NIDC. He is currently editor of IEEE Transactions on Vehicular Technology (TVT), IEEE Network Magazine, IEEE Access, IEEE Open Journal of the Communications Society, and Ad Hoc & Sensor Wireless Networks (AHSWN) journal. Dr. Fadlullah is a Senior Member of the Institute of Electrical and Electronics Engineers (IEEE), and IEEE Communications Society (ComSoc).

Nidal Nasser (SM-IEEE) completed his Ph.D. in the School of Computing at Queen's University, Kingston, Ontario, Canada, in 2004. He is currently a Professor of Software Engineering in the College of Engineering at Alfaisal University, Saudi Arabia. He worked in the School of Computer Science at University of Guelph, Guelph, Ontario, Canada. Dr. Nasser was the founder and Director of the Wireless Networking and Mobile Computing Research Lab @ Guelph. He is currently the founder and Director of the Telecommunications Computing Research Lab @ Alfaisal University. He has authored 180 journal publications, refereed conference publications and book chapters in the area of wireless communication networks and systems. He is currently serving as an associate editor of IEEE Wireless Communications Magazine, Wiley's International Journal on Communication Systems, and IEEE CommSoft E-letter. He has been a member of the technical program and organizing committees of several international IEEE conferences and workshops. Dr. Nasser received several outstanding research awards as well as a number of best paper awards.