Received: 16 December 2022 | Revised: 15 March 2023 | Accepted: 27 March 2023

DOI: 10.4218/etrij.2022-0459

ORIGINAL ARTICLE

ETRI Journal WILEY

Bi-LSTM model with time distribution for bandwidth prediction in mobile networks

Hyeonji Lee[1] | Yoohwa Kang[2] | Minju Gwak[1] | Donghyeok An[1]

[1]Department of Computer Engineering, Changwon National University, Changwon, Republic of Korea

[2]Network Research Division, Electronics and Telecommunications Research Institute, Daejeon, Republic of Korea

Correspondence
Donghyeok An, Changwon National University, Changwon, Republic of Korea.
Email: donghyeokan@changwon.ac.kr

Funding information
Ministry of Science and ICT, South Korea, Grant/Award Number: 2020-0-00974

Abstract

We propose a bandwidth prediction approach based on deep learning. The approach is intended to accurately predict the bandwidth of various types of mobile networks. We first use a machine learning technique, namely, the gradient boosting algorithm, to recognize the connected mobile network. Second, we apply a handover detection algorithm based on network recognition to account for vertical handover that causes the bandwidth variance. Third, as the communication performance offered by 3G, 4G, and 5G networks varies, we suggest a bidirectional long short-term memory model with time distribution for bandwidth prediction per network. To increase the prediction accuracy, pretraining and fine-tuning are applied for each type of network. We use a dataset collected at University College Cork for network recognition, handover detection, and bandwidth prediction. The performance evaluation indicates that the handover detection algorithm achieves 88.5% accuracy, and the bandwidth prediction model achieves a high accuracy, with a root-mean-square error of only 2.12%.

KEYWORDS
bandwidth prediction, deep learning, handover detection, LSTM, network recognition

1 | INTRODUCTION

Mobile traffic is growing quickly because of the proliferation of smart devices and advancements in mobile communication technologies. From 31.16% in the first quarter of 2015 to 58.99% in the second quarter of 2022, smart devices generate a growing portion of Internet traffic [1]. Users of smart devices frequently use video streaming services, such as Netflix and Amazon Prime Video. In particular, live video streaming services are becoming increasingly popular, and by 2025, traffic related to such services on the Internet is expected to account for 17% of the total traffic [2]. As smart devices are highly mobile, offering a steady streaming service is challenging because of frequent dynamic network bandwidth changes that substantially impact live video streaming.

Adaptive bitrate streaming (ABS) is adopted for streaming video in dynamic network environments. ABS allows adapting the video quality to fluctuating network capacity [3]. ABS enables high-quality video if the network bandwidth is high and reduces the quality otherwise. However, ABS fails to suitably react to sudden bandwidth changes. ABS may timely respond to changes in the network environment if the bandwidth is predicted in a mobile network, in which large bandwidth changes likely occur due to user mobility. Hence, the bandwidth should be predicted after identifying the presently accessed network to account for the different communication properties of 3G, 4G, and 5G mobile networks [4]. In 4G networks, a latency of 10 ms is expected, while it reduces to under 1 ms in 5G networks. In addition, the peak data rates in 4G and 5G networks are 1 Gbps and 20 Gbps, respectively, while the user-experienced data rates are 10 Mbps and 100 Mbps, respectively. Real-time services with low latency and requiring high transmission reliability, such as extended reality, demand preparations for network quality of service (QoS) changes through bandwidth prediction.

Figure 1 shows the market share of 3G, 4G, and 5G networks worldwide. While 4G has a market share of more than 70% in Europe and North America, 2G and 3G remain representative in Asia and Latin America [5]. Overall, 3G, 4G, and 5G networks coexist globally. When various mobile communication technologies are simultaneously accessible, mobile smart devices can switch the type of mobile network as they displace. Mobile smart devices experience horizontal handover in the same network and vertical handover across different mobile networks. Sudden bandwidth variations occur under vertical handover [6, 7]. Even when ABS is employed, vertical handover causes unexpected fluctuations in network bandwidth, and live streaming experiences playback interruption and buffering. In addition, transmission control protocol (TCP) is disconnected during a vertical handover because the Internet protocol (IP) address changes, preventing users from accessing real-time services until reestablishing the TCP connection. Under vertical handover, TCP cannot adapt to the large variations in bandwidth. To respond quickly to bandwidth fluctuations and connection termination, handover detection is necessary.

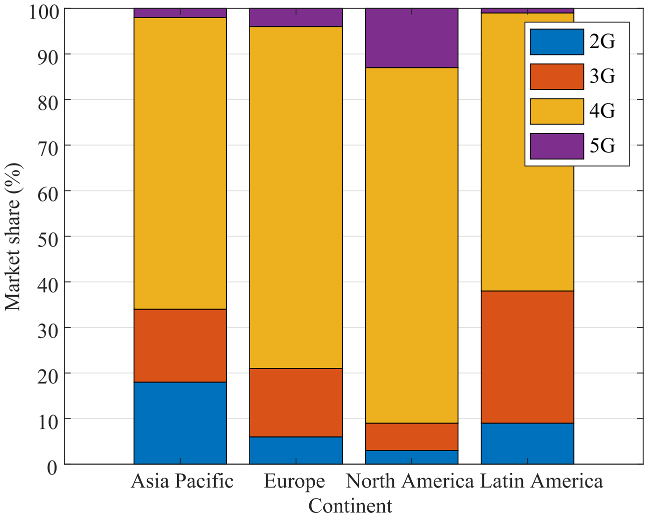


FIGURE 1 Market share of telecommunication technologies.

For stable live streaming on mobile networks, the network bandwidth should be predicted. As the network performance of 3G, 4G, and 5G communication technologies varies, the network bandwidth should be estimated according to the type of mobile network connection. Before bandwidth prediction, the currently active network should be identified. Therefore, to adequately predict the network bandwidth in a mobile network, the connected mobile network should be identified. Then, the bandwidth of the recognized mobile network should be predicted accordingly.

Several methods for handover detection have been proposed. To judge the occurrence of handover, a deep neural network and a variety of machine learning techniques, such as naïve Bayes, support vector machine (SVM), and multilayer perceptron, have been employed [8–10]. Based on the user movement pattern and current location, handover detection was performed in Han and others [11]. However, previous studies focused on the decision for handover execution rather than handover detection. When the movement patterns vary, they cannot contribute to accurate handover detection. On the other hand, bandwidth prediction has mainly been based on either throughput modeling or communication history/patterns [12–16]. These approaches are ineffective for mobile networks in dynamic communication environments because a stable communication environment is assumed. Several techniques for bandwidth prediction based on deep learning, such as the long short-term memory (LSTM) architecture, have been explored [7, 17, 18]. Existing deep learning studies focused on environments with a single type of mobile network, whereas we considered environments with coexisting mobile networks, including 3G, 4G, and 5G networks, in this study.

We aim to predict the bandwidth in mobile networks implementing the 3G, 4G, 5G technologies by using measurable data from smart devices. First, we propose a machine-learning-based network recognition technique. From a dataset, 10 representative features are selected as the input. After evaluating the network recognition performance of eight machine learning methods, we determine that the gradient boosting machine (GBM) is the most appropriate for network recognition. Second, we propose a vertical handover detection algorithm. Bandwidth fluctuations occur due to handovers across networks. The proposed algorithm recognizes vertical handover to deal with internetwork bandwidth variations. The algorithm may allow to quickly handle such variations. Third, we propose a bandwidth prediction model called bidirectional LSTM (Bi-LSTM) mode with time distribution (TD) to estimate the bandwidth of each available network. After pretraining based on traces involving 3G, 4G, and 5G networks, fine-tuning for each type of mobile network is applied to improve the prediction accuracy. We achieve outstanding performance for the handover detection algorithm and bandwidth prediction model for each type of network. In addition to mobile users, stationary users can adapt to bandwidth variations over time using our approach. Hence, uninterrupted streaming is guaranteed, and users can access video streaming and other demanding services without experiencing connection problems regardless of bandwidth variations.

The rest of this paper is organized as follows. In Section 2, we present related work. In Section 3, we discuss the dataset for network recognition, handover detection, and bandwidth prediction. Section 4 details network recognition per type of mobile network based on machine learning. In Sections 5 and 6, we introduce the handover detection algorithm and bandwidth prediction through the Bi-LSTM model with TD, respectively. In Section 7, the performance of handover detection and bandwidth prediction is reported. Finally, we draw conclusions in Section 8.

2 | RELATED WORK

Various studies on handover in mobile networks are available. In Huang and others [9], the user equipment (UE) employed a deep neural network considering the change in signal-to-interference-plus-noise ratio (SINR) to decide handover between cells in a 5G network. In Lima and others [10], various machine learning methods, including SVM and multilayer perceptron, performed handover decisions in long term evolution (LTE) networks. In Goutam and Unnikrishnan [8], a naïve Bayes algorithm was proposed for vertical handover decision from inputs of network coverage, bandwidth, receive side scaling, and QoS. Handover execution based on extreme gradient boost (XGBoost) between a millimeter-wave and LTE network was proposed in Nayakwadi and Fatima [19]. XGBoost-based handover mechanism increased the handover success rate and decreased the signal overhead. Unlike earlier studies focused on handover decision-making, we apply machine learning to handover detection. In Han and others [11], handover detection was achieved using data mining. To anticipate the future location of a mobile terminal, data mining was used to analyze object movement patterns. Using the predicted location, handover was detected. However, handover detection was incorrect if the movement patterns differed. In contrast, our proposal uses diverse network features for handover detection.

Studies on real-time bandwidth prediction have been carried out for many years. In Huang and Subhlok [12], a pattern based on the TCP window size was used to forecast the TCP throughput. A heuristic algorithm was used in the absence of a matching pattern. In Mirza and others [14], based on prior file transfer data, support vector regression, an SVM approach for regression, was used to predict the throughput. Various studies have used formulations to predict the TCP throughput. In the literature [15, 16], throughput models based on the round-trip time (RTT) and loss rate performed throughput prediction. However, the prediction accuracy using these indicators was low given the difficulty to correctly measure the RTT and loss rate. To improve the accuracy of TCP throughput prediction, a throughput model based on the available bandwidth and router buffer size was proposed in Hwang and Yoo [13]. However, throughput prediction studies based on history or formulas could not be used in a mobile network given the dynamically changing communication environment instead of the commonly assumed stable communication environment.

Deep learning has been applied to bandwidth prediction. TRUST predicted the TCP throughput in mobile networks by analyzing movement patterns and then estimating the throughput using an LSTM model according to these patterns [18]. In Mei and others [17], an LSTM recurrent neural network predicted the bandwidth in a mobile network. To choose the best model from pretrained LSTM models for online bandwidth prediction, model switching and Bayesian fusion were applied. In Li and others [7], for various movement scenarios, an attention-based LSTM model employed bandwidth prediction considering bandwidth fluctuations based on the form of transportation of the user, such as bus or train. To predict bandwidth, the attention-based LSTM model first used an SVM to identify the movement scenario and then employed LSTM and an attention mechanism to predict the bandwidth. However, previous studies predicted the throughput in mobile networks considering only HSDPA (High-Speed Downlink Packet Access) or LTE communications. On the other hand, we predict the bandwidth considering mobile networks implementing the 3G, 4G (or LTE), and 5G technologies. Similarly, in Yun and others [20], machine learning was used for bandwidth prediction and network resource reservation for efficient large-capacity data transfer in high-performance networks.

3 | DATASET

For network recognition, handover detection, and bandwidth prediction, we used the dataset collected by the

Mobile and Internet Systems Laboratory at University College Cork [21, 22]. The dataset has 83 trace files and measurements over 3142 min. Twenty-six network properties, including relative signal strength indicator (RSSI), uplink bandwidth, and downlink bandwidth, were measured every second using the G-NetTrack Pro monitor [23]. Without requiring root access, G-NetTrack Pro gathered data on a variety of channels, contexts, cells, and throughputs on Android devices. The dataset was measured in two types of mobility patterns: stationary and moving in a car. The bandwidth was evaluated using two applications: file download and video streaming in Netflix and Amazon Prime Video. The 5G network was an Irish network, and in areas without 5G base stations, access to a 4G or 3G network was available. For scenarios using different networks, such as 3G, 4G, and 5G, we employed the dataset in a mobile environment. The dataset obtained from file download was used to focus on the network bandwidth. Hence, during movement, 16 file download traces were used with measurement duration of 27607s. For training, 12 trace files with a measurement time of 20008 s were used. For testing, 4 trace files with a measurement time of 7599 s was used. A total of 27607 samples were obtained because data were measured every second over the measurement interval, with 20008 samples used for training and the remaining 7599 samples used for testing.

Figure 2 shows the download rate per trace over time. The green, orange, and blue curves indicate 3G, 4G, and 5G networks, respectively. As shown in the figure, 14 traces used two or three types of networks. As expected, the fastest average download rate was obtained for 5G networks followed by 4G and finally 3G networks. For each trace, the usage time of each type of network varied. Figure 3 shows the measured usage times. Compared with 3G networks, 5G networks were used more frequently and achieved a longer overall usage time. However, as shown in Figures 2 and 3, the connections to 3G networks lasted longer than those to 5G networks given the larger 3G coverage and less frequent disconnections compared with 5G technology.

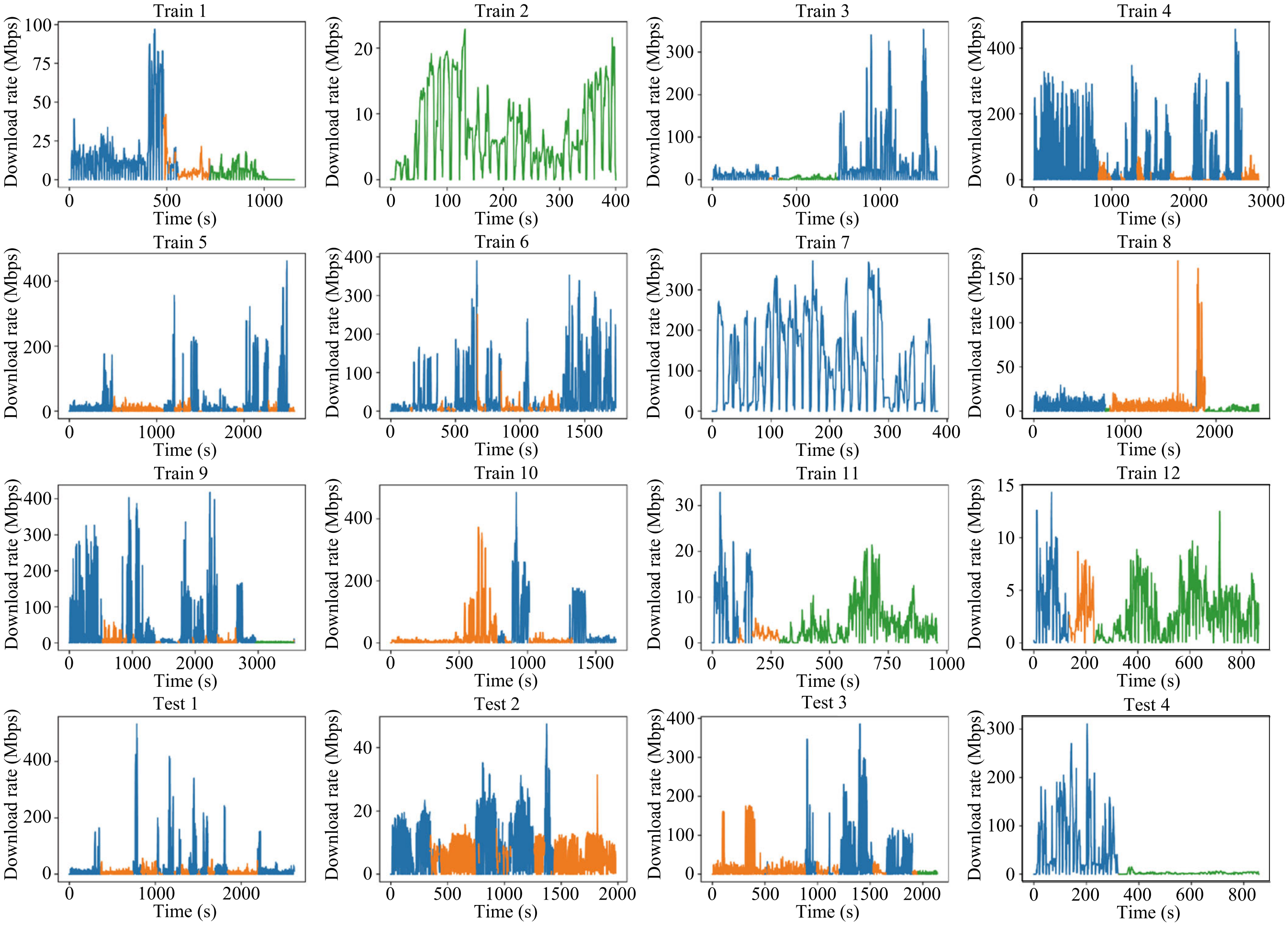

Train 1 Train 2 Train 3 Train 4 Train 5 Train 6 Train 7 Train 8 Train 9 Train 10 Train 11 Train 12 Test 1 Test 2 Test 3 Test 4

FIGURE 2 Download rate per trace over time.

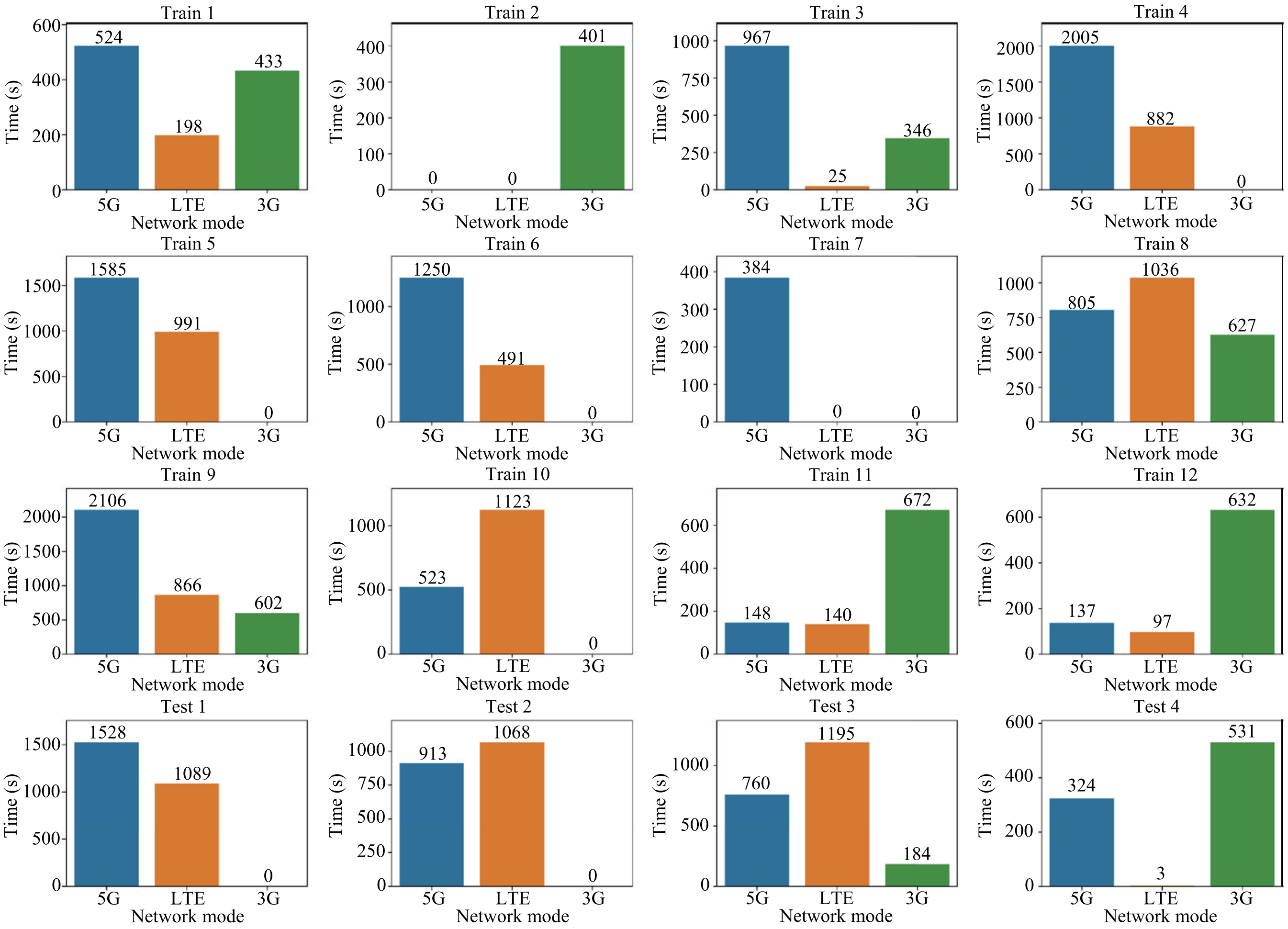


FIGURE 3 Usage time per mobile network within each trace.

4 | NETWORK RECOGNITION

We begin by discussing network recognition based on machine learning. To identify the network to which a smart device is connected, the proposed scheme employs machine learning. To determine the best machine learning technique for network recognition, we assessed the network recognition performance of eight candidates: logistic regression, naïve Bayes, SVM, random forest, decision tree, K-nearest neighbors (K-NN), GBM, and adaptive boosting (AdaBoost). To determine input features for effective network recognition using machine learning, we employed the permutation feature importance of random forest to select the most representative among the 26 network features collected in the dataset. The 10 features listed in Table 1 were used as input. Over the measurement period, features were periodically collected, and their averages and standard deviations were used as inputs. For instance, the reference signal received power (RSRP) was measured once every second over 10 s for a measurement period of the same duration. Then, our scheme calculated the mean and standard deviation across the 10 measurements to establish the input for machine learning.

TABLE 1 Features used as input for machine learning.

Feature	Description
DL bitrate	Downlink bandwidth (Mbps)
UL bitrate	Uplink bandwidth (Mbps)
NRxSRP	RSRP of neighboring cell
RSRP	Reference signals received power
RSSI	Received signal strength indicator
NRxSRQ	RSRQ of neighboring cell
Speed	Displacement speed
SNR	Signal-to-noise ratio
RSRQ	Reference signal received quality
CQI	Channel quality indicator

We measured the accuracy, precision, recall, and recognition time to assess the network recognition

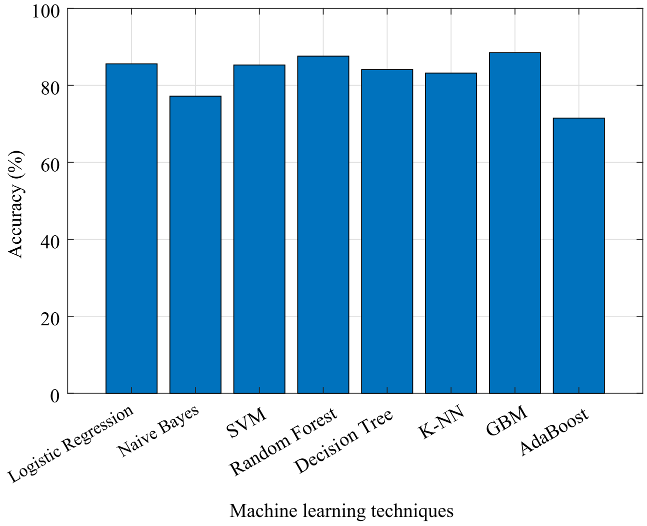


FIGURE 4 Accuracy of network recognition using different machine learning techniques.

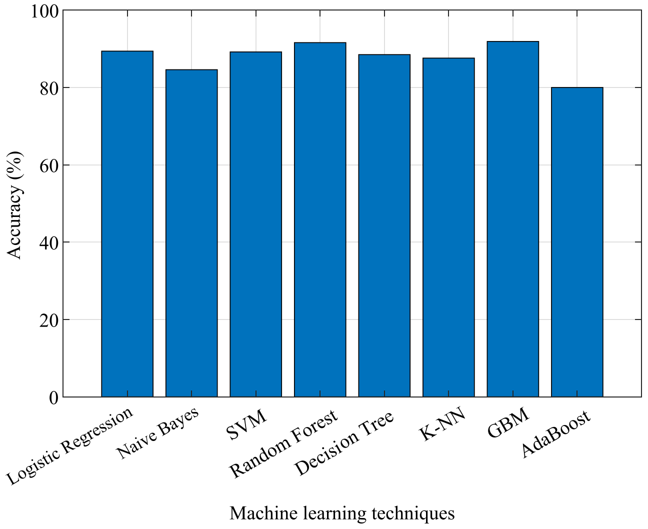


FIGURE 5 Precision of network recognition using different machine learning techniques.

performance. The percentage of accurate responses across all the predictions defines the accuracy, which is obtained as follows:

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FN}+\text{FP}+\text{TN}}, \tag{1}$$

where TP, TN, FN, and FP indicate numbers of true positives, true negatives, false negatives, and false positives, respectively. The precision is defined as the ratio of true positives to expected positives as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}. \tag{2}$$

The recall is the proportion of correct predictions to correct cases and calculated as follows:

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}}. \tag{3}$$

The recognition time is the amount of time taken by a machine learning technique to identify a network.

The accuracy and precision of network recognition for the eight machine learning techniques are shown in Figures 4 and 5, respectively. GBM outperformed the other machine learning techniques with an accuracy of 88.5%. In terms of precision, GBM and random forest achieved high performance, with values of 91.9% and 91.6%, respectively. The recall of network recognition is shown in Figure 6. The highest performance was achieved by GBM at 91.5%. Figure 7 shows the time required for network recognition. Logistic regression, naive Bayes, and decision tree recognition were the fastest techniques for network recognition. However, these techniques performed poorly in terms of accuracy, precision, and recall. As the GBM achieved superior network recognition accuracy, precision, and recall as well as a reasonable recognition time of 15 ms, we used it as the machine learning technique for network recognition.

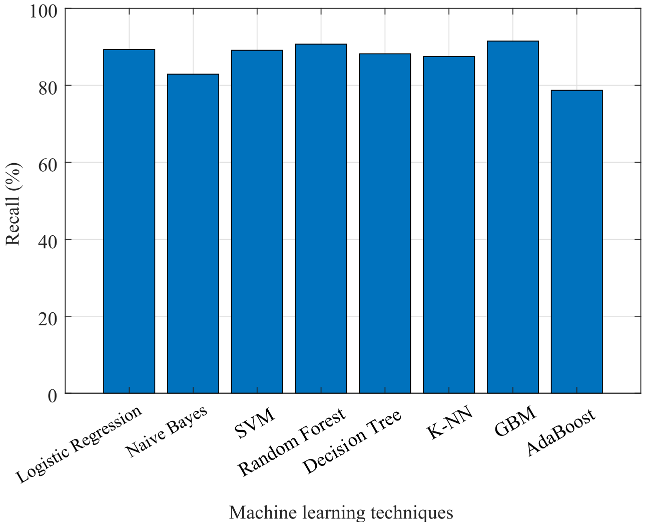


FIGURE 6 Recall of network recognition using different machine learning techniques.

5 | HANDOVER DETECTION

We introduce a handover detection algorithm based on network recognition and explain three typical handover scenarios. The proposed algorithm employs the machine-

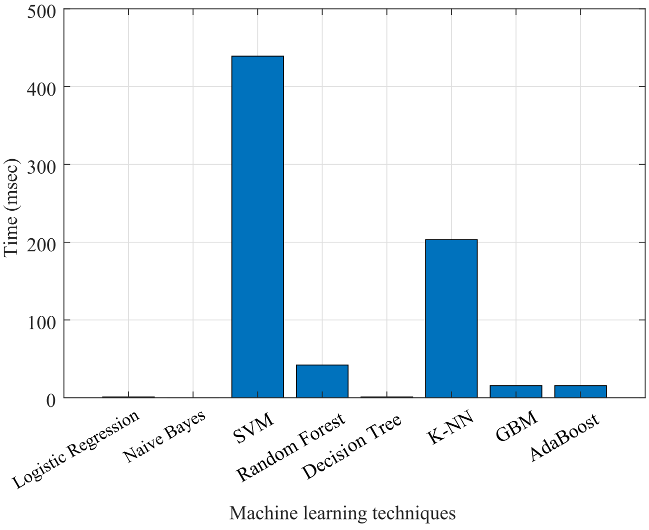


FIGURE 7 Network recognition time for different machine learning techniques.

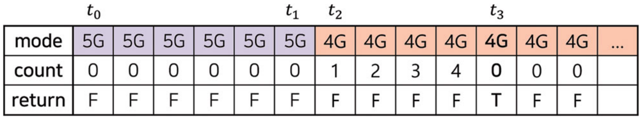
learning-based network recognition technique to regularly identify the current network. A handover is recognized if a network other than the currently used one is consistently identified above a threshold. For instance, if a 4G network is consistently detected above the threshold and the current network is 5G, handover is detected from the 5G to the 4G network.

Algorithm 1 describes the proposed handover detection algorithm. A network already recognized by a smart device is designated by a base variable, and a network that was just recognized and distinct from the base is designated by a temp variable. The current variable refers to the network that is currently identified by the proposed network recognition scheme. When operation DETECT_HANDOVER is executed, the current value, also known as currently recognized network identifier, is supplied as a parameter. If the base and current network identifiers are different, a network other than the base has been detected. If the temp and current variables are equal, a counter is increased because the same network is being identified. The counter is set to 1, and the current value is kept in the temp variable in case that the current and temp variables differ, indicating a new network connection. If the base and current network identifiers are the same but the base and temp variables differ, connection to a different network for a short time or improper network detection occurred. Hence, the temp variable is initialized with the value of the base variable. When the counter reaches a threshold, a handover has taken place because the same network has lately been identified numerous times. The base variable is updated with the value of the temp variable, and the counter is initialized when handover is detected.

Algorithm 1 Handover detection

```
current: current recognized network

count = 0
base = temp = current

procedure DETECT_HANDOVER(current)
    if base != current then
        if temp == current then
            count++
        else if temp != current then
            count = 1
            temp = current
        end if
    else if base != temp then
        temp = base
    end if

    if count == threshold then
        base = temp
        count = 0
        return true
    end if

    return false
end procedure
```

The first scenario of handover from a 5G to a 4G network is described in Figure 8. The handover recognition threshold is 5. The base, temp, and current variables are listed in Table 2 at different instants. The values of the base, temp, and current variables are identical for the 5G network from t_0 to t_1. At time t_2, the 4G network is detected, and the temp and current values are updated to 4G. At t_3, the 4G network is identified over five consecutive timesteps. As a result, the handover detection



	t_0					t_1	t_2				t_3			
mode	5G	5G	5G	5G	5G	5G	4G	4G	4G	4G	**4G**	4G	4G	...
count	0	0	0	0	0	0	1	2	3	4	**0**	0	0	
return	F	F	F	F	F	F	F	F	F	F	**T**	F	F	

FIGURE 8 Handover scenario 1.

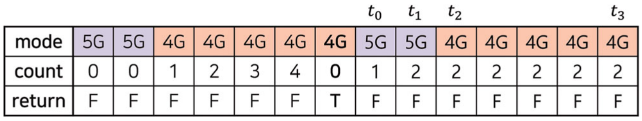
TABLE 2 Variable values for handover scenario 1 at different instants.

	t_0	t_1	t_2	t_3
base	5G	5G	5G	4G
temp	5G	5G	4G	4G
current	5G	5G	4G	4G

algorithm recognizes the handover and switches the base value to 4G.

Figure 9 describes the second scenario, in which a 5G network switches to a 4G network. After briefly reconnecting to the 5G network, a connection is established to the 4G network. The values of the base, temp, and current variables are 4G, 5G, and 5G, respectively, from t_0 to t_1, as listed in Table 3. A 4G network is recognized at t_2, and the temp variable is updated to 4G. At t_3, the 4G network is identified at five consecutive timesteps, but as the base and current variables are equal, this is not regarded as a handover.

The final handover scenario is described in Figure 10. Switching from a 5G to a 4G network and then switching from the 4G network to a 3G network occur. Table 4 lists the values of the base, temp, and current variables from t_1 to t_4. The 4G network is identified at time t_1, and the algorithm recognizes the handover from the 5G to the 4G network at time t_2. After t_2, it joins the 5G network, but before the count hits the threshold, it connects to the 3G network at t_3. At t_4, the counter increases to 5, and handover from the 4G to the 3G network is detected.

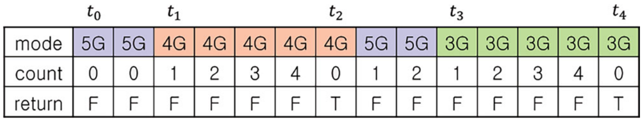


							t_0	t_1	t_2					t_3
mode	5G	5G	4G	4G	4G	4G	4G	5G	5G	4G	4G	4G	4G	4G
count	0	0	1	2	3	4	0	1	2	2	2	2	2	2
return	F	F	F	F	F	F	T	F	F	F	F	F	F	F

FIGURE 9 Handover scenario 2.

TABLE 3 Variable values for handover scenario 2 at different instants.

	t_0	t_1	t_2	t_3
base	4G	4G	4G	4G
temp	5G	5G	4G	4G
current	5G	5G	4G	4G



	t_0		t_1				t_2			t_3				t_4
mode	5G	5G	4G	4G	4G	4G	4G	5G	5G	3G	3G	3G	3G	3G
count	0	0	1	2	3	4	0	1	2	1	2	3	4	0
return	F	F	F	F	F	F	T	F	F	F	F	F	F	T

FIGURE 10 Handover scenario 3.

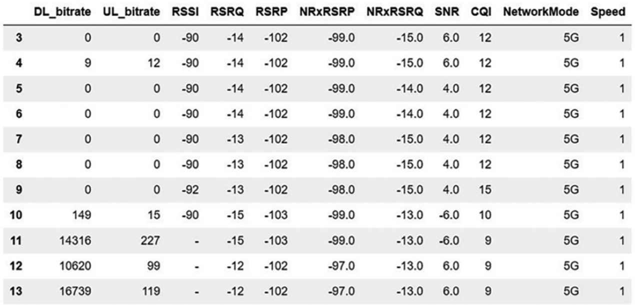
TABLE 4 Variable values for handover scenario 3 at different instants.

	t_0	t_1	t_2	t_3	t_4
Base	5G	5G	4G	4G	3G
Temp	5G	4G	4G	3G	3G
Current	5G	4G	4G	3G	3G

6 | BANDWIDTH PREDICTION

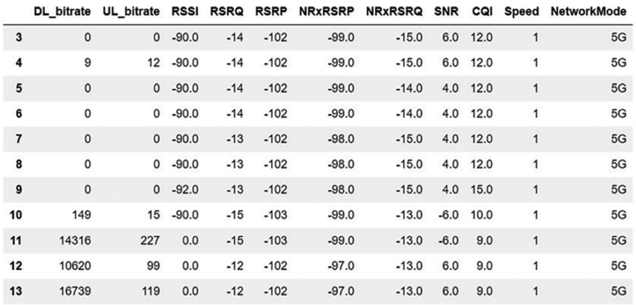
For bandwidth prediction in a mobile network, the three main components of the proposed scheme are data preprocessing, pretraining for the deep learning model, and fine-tuning of the pretrained model. First, the missing values in the dataset are identified during data preprocessing. If there are missing values, the corresponding entries are filled with zeros. Data before and after preprocessing are shown in Figures 11 and 12. As the indexes from 11 to 13 in Figure 11 have no RSSI values, the corresponding entries are set to zero, as indicated in Figure 12. Then, normalization between 0 and 1 using minmaxscaler is applied to the dataset values.

The dataset shows frequent vertical handover across 3G, 4G, and 5G networks. As a result, the access time of each network is short. When the connection period per network is brief, time-series data processing using an optimized LSTM model has a detrimental impact on bandwidth prediction. Hence, we apply pretraining to the model for each trace regardless of the type of mobile network to improve the accuracy of bandwidth prediction.



	DL_bitrate	UL_bitrate	RSSI	RSRQ	RSRP	NRxRSRP	NRxRSRQ	SNR	CQI	NetworkMode	Speed
3	0	0	-90	-14	-102	-99.0	-15.0	6.0	12	5G	1
4	9	12	-90	-14	-102	-99.0	-15.0	6.0	12	5G	1
5	0	0	-90	-14	-102	-99.0	-14.0	4.0	12	5G	1
6	0	0	-90	-14	-102	-99.0	-14.0	4.0	12	5G	1
7	0	0	-90	-13	-102	-98.0	-15.0	4.0	12	5G	1
8	0	0	-90	-13	-102	-98.0	-15.0	4.0	12	5G	1
9	0	0	-92	-13	-102	-98.0	-15.0	4.0	15	5G	1
10	149	15	-90	-15	-103	-99.0	-13.0	-6.0	10	5G	1
11	14316	227	-	-15	-103	-99.0	-13.0	-6.0	9	5G	1
12	10620	99	-	-12	-102	-97.0	-13.0	6.0	9	5G	1
13	16739	119	-	-12	-102	-97.0	-13.0	6.0	9	5G	1

FIGURE 11 Data before preprocessing.



	DL_bitrate	UL_bitrate	RSSI	RSRQ	RSRP	NRxRSRP	NRxRSRQ	SNR	CQI	Speed	NetworkMode
3	0	0	-90.0	-14	-102	-99.0	-15.0	6.0	12.0	1	5G
4	9	12	-90.0	-14	-102	-99.0	-15.0	6.0	12.0	1	5G
5	0	0	-90.0	-14	-102	-99.0	-14.0	4.0	12.0	1	5G
6	0	0	-90.0	-14	-102	-99.0	-14.0	4.0	12.0	1	5G
7	0	0	-90.0	-13	-102	-98.0	-15.0	4.0	12.0	1	5G
8	0	0	-90.0	-13	-102	-98.0	-15.0	4.0	12.0	1	5G
9	0	0	-92.0	-13	-102	-98.0	-15.0	4.0	15.0	1	5G
10	149	15	-90.0	-15	-103	-99.0	-13.0	-6.0	10.0	1	5G
11	14316	227	0.0	-15	-103	-99.0	-13.0	-6.0	9.0	1	5G
12	10620	99	0.0	-12	-102	-97.0	-13.0	6.0	9.0	1	5G
13	16739	119	0.0	-12	-102	-97.0	-13.0	6.0	9.0	1	5G

FIGURE 12 Data after preprocessing.

TABLE 5 Hyperparameters for pretraining.

Hyperparameter	Value
Timestep	10
Dropout rate	0.2
Optimizer	Adam
Learning rate	0.0001
Loss function	MSE
No. epochs	1500
Batch size	128

TABLE 6 Hyperparameters for fine-tuning.

	Value		
Hyperparameter	5G	4G	3G
Learning rate	0.00002	0.00001	0.00002
No. of epochs	2	5	5
Batch size	192	128	16

Pretraining is performed using the Bi-LSTM model with TD model. This model consists of two LSTM layers, three dropout layers, one TD layer, and one Bi-LSTM layer. Each layer has 48 units. The hyperparameters for model training are listed in Table 5.

Regardless of the network, the pretrained model can predict the bandwidth. The model for each type of network is then fine-tuned based on the pretrained model to improve the accuracy of bandwidth prediction. For instance, the pretrained model is fine-tuned using data from 5G networks to predict the bandwidth of those networks. Pretraining and fine-tuning use the Bi-LSTM model with TD model. The hyperparameters used for fine-tuning are listed in Table 6. The learning rate and number of epochs are less than those of pretraining to reduce the loss for each type of network because fine-tuning progresses with more learning on the pretrained model. As the batch size affects the model performance [24, 25], each network model is fine-tuned with a variable batch size. The pretrained model hyperparameters, including timestep, dropout rate, optimizer, and loss function, are also used for fine-tuning.

The 10 features listed in Table 1 are the inputs used by the Bi-LSTM model with TD model. The Shannon–Hartley theory states that an important feature is the number of allocated resource blocks because the usable bandwidth is proportional to that number. The reference signal received quality (RSRQ) is calculated using RSRP, RSSI, and the number of resource blocks, as indicated in (4) [26], where N_{RB} is the number of resource blocks in the measurement bandwidth. As the proposed Bi-LSTM model with TD model employs RSRQ, RSRP, and RSSI as inputs, the number of resource blocks is implicit in the proposed model and used for bandwidth prediction. However, the number of resource blocks allotted to a UE may vary depending on the number of active users in a cell, which affects the prediction of the UE available bandwidth [27]. Nevertheless, the proposed bandwidth prediction model uses RSRQ without considering the number of users owing to the lack of this information in the dataset.

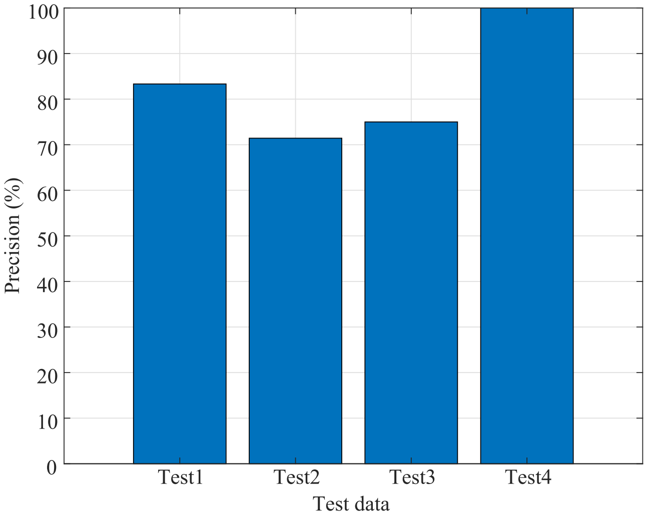


FIGURE 13 Precision of handover detection algorithm.

$$\mathrm{RSRQ} = \frac{N_{\mathrm{RB}} \times \mathrm{RSRP}}{\mathrm{RSSI}}. \tag{4}$$

7 | EVALUATION

We evaluated the performance of the proposed approach. First, we measured the performance of the proposed handover detection algorithm and then that of the bandwidth prediction model.

7.1 | Handover detection

We evaluated the performance of the handover detection algorithm by measuring the precision, recall, and $F1$ score on the test set. In every test scenario, as handover detection delays with an increasing measurement period, we set the measurement period to 20 s. First, we measured the precision, which is the proportion of true positive results to the actual number of positive results. The measured precision is shown in Figure 13. The handover

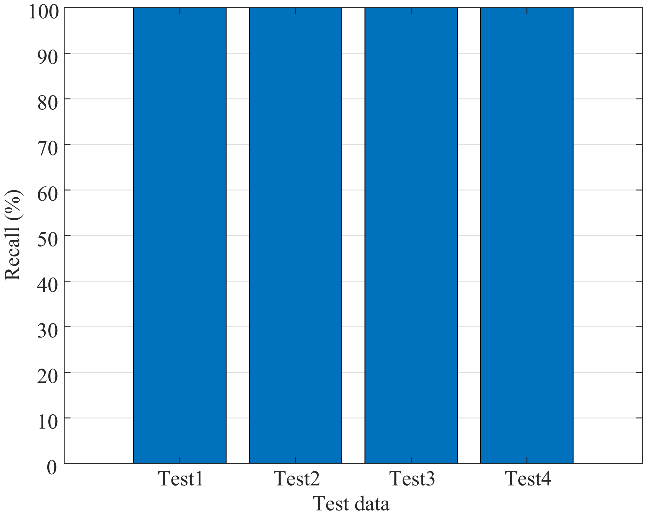


FIGURE 14 Recall of handover detection algorithm.

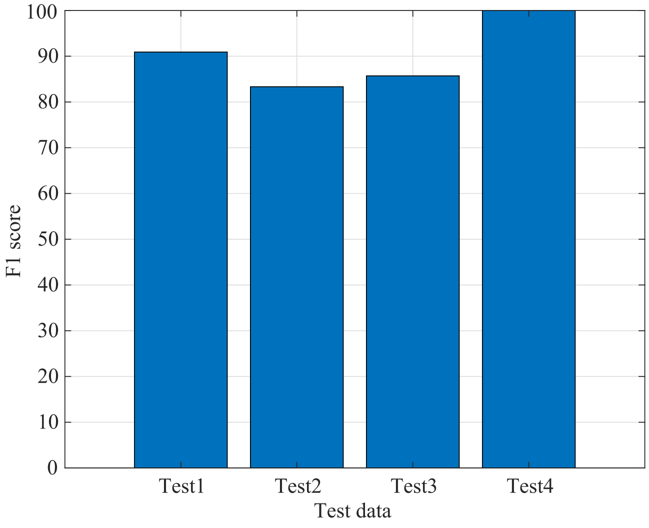


FIGURE 15 $F1$ score of handover detection algorithm.

detection precision in every scenario exceeds 70%, with an average of 82.44%. The measured recall is shown in Figure 14 and indicates the proportion of true positives to true cases. In the measurement results, it performed the best on all tests. According to the precision and recall, the proposed handover detection algorithm detected all true vertical handovers, but precision suffered because it predicted more vertical handovers than the actual number.

The $F1$ score was calculated considering both precision and recall. The $F1$ score is calculated as

$$F1\ \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

where *Precision* and *Recall* are the precision given by (2) and recall given by (3), respectively. An increase in the $F1$ score indicates that the handover detection performance enhances. Figure 15 shows the $F1$ score for the test set. Most of the $F1$ scores were higher than 80, demonstrating that the proposed handover detection algorithm adequately detected vertical handover.

7.2 | Bandwidth prediction

We also evaluated the accuracy of bandwidth prediction for each network by calculating the root-mean-square error (RMSE) as follows:

$$\text{RMSE} = \sqrt{\frac{1}{L}\sum_{n=1}^{L}(y_n - p_n)^2}. \tag{6}$$

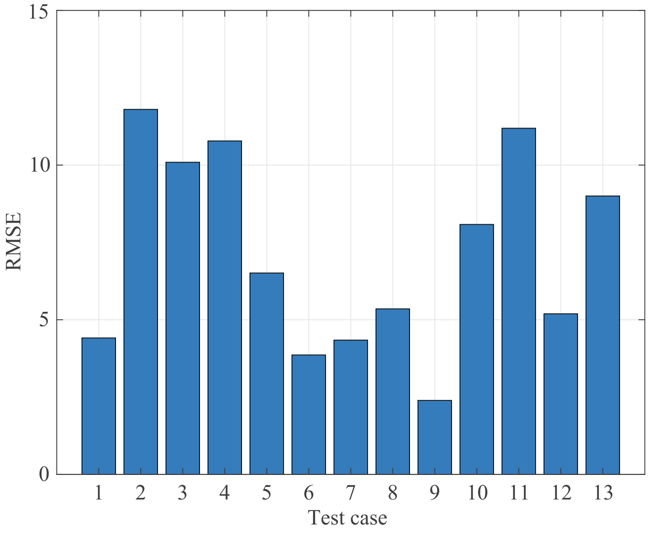


FIGURE 16 Accuracy of bandwidth prediction for 5G network.

The difference between the actual value, y_n, and predicted value, p_n, is used to calculate an error value that is squared. The same operation is performed on all the L samples, and the sum of the errors is calculated. Then, the average error is calculated by dividing the sum by L. The square root is finally taken to fit the original data dimension.

Only 13 5G communication samples were extracted from four test sets to evaluate the bandwidth prediction accuracy of 5G networks. Figure 16 shows the measured accuracy of bandwidth prediction, and the average RMSE of the proposed model was 7.15. The bandwidth prediction of our model was accurate despite the large bandwidth fluctuations in the 5G network. Eight 4G network samples were extracted from four test sets. Figure 17 shows the bandwidth prediction on eight 4G network samples. The average RMSE was 2.12, and most of the samples showed a low RMSE. Hence, the proposed model accurately predicted the 4G network bandwidth. Owing to the low connection frequency of 3G networks, two samples were extracted from the test set. Figure 18 shows the measured accuracy of bandwidth prediction for 3G networks. The average RMSE was 0.57, indicating an excellent accuracy of bandwidth prediction.

Figures 16, 17, and 18 show that the average RMSE, which measures the accuracy of bandwidth prediction, degrades as the communication generations increase from 3G to 4G and finally 5G. Considering resource blocks allotted to the UE is more complicated because a higher communication generation uses a higher carrier frequency and requires more resource blocks. We used a limited dataset that was measured in Android devices without root access owing to the lack of a specific dataset, as described in Section 6. The accuracy of bandwidth prediction for 5G networks may increase if we incorporate the number of resource blocks allotted to the UE as input in future work.

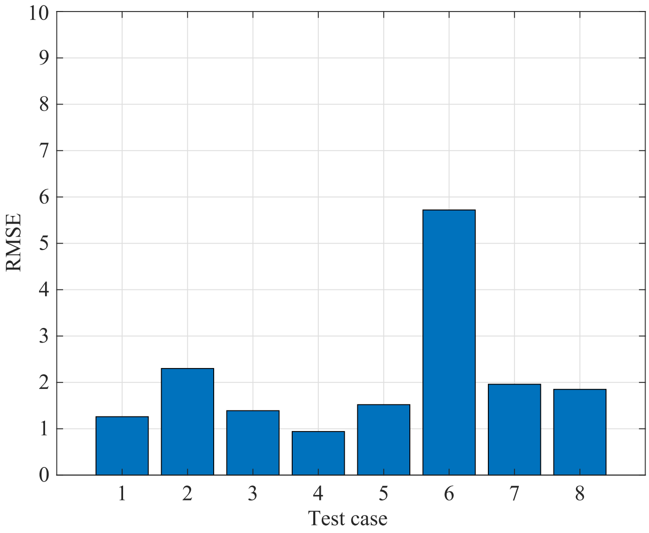


FIGURE 17 Accuracy of bandwidth prediction for 4G network.

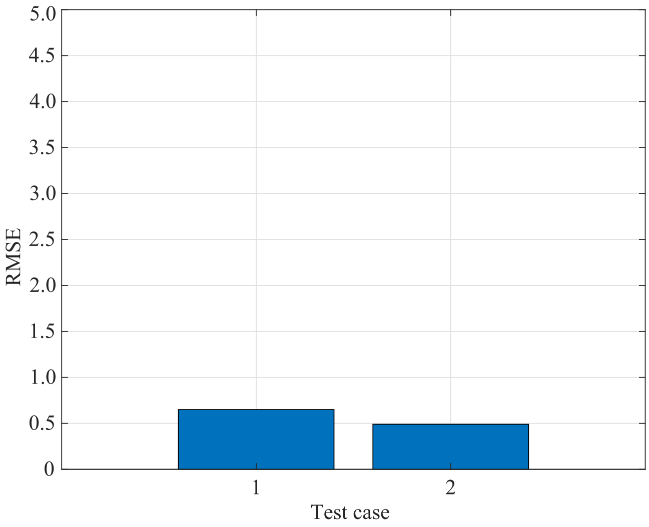


FIGURE 18 Accuracy of bandwidth prediction for 3G network.

8 | CONCLUSION

We propose a bandwidth prediction approach for mobile networks. Machine learning and the permutation feature importance of random forest are used to identify the currently connected network. Based on the network recognition, we apply a vertical handover detection algorithm. When a handover is detected, the Bi-LSTM model with TD model for the corresponding type of mobile network is used to predict the bandwidth. Handover detection and bandwidth prediction achieved high performance in evaluation tests. Future studies will include the addition of the number of resource blocks allocated to the UE to the input features for bandwidth prediction, aiming to increase the accuracy of bandwidth prediction, especially for 5G networks.

AUTHOR CONTRIBUTIONS

Hyeonji Lee and Yoohwa Kang contributed equally to this work.

ACKNOWLEDGEMENTS

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) under grant funded by the Korean government (MSIT) (No. 2020-0-00974, Development of Ultra-reliable and Low-Latency 5G+ Core Network and TSN Switch Technologies).

CONFLICT OF INTEREST STATEMENT

The authors declare that there are no conflicts of interest.

ORCID

Yoohwa Kang https://orcid.org/0000-0001-6311-7431

REFERENCES

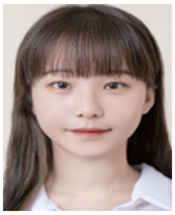
1. Percentage of mobile device website traffic worldwide from 1st quarter 2015 to 2nd quarter 2022. https://www.statista.com/statistics/277125/share-of-website-traffic-coming-from-mobile-devices/ Accessed: 2022-10-06.

2. Consumer internet data traffic worldwide by application category from 2016 to 2022. https://www.statista.com/statistics/454951/mobile-data-traffic-worldwide-by-application-category/ Accessed: 2022-10-06.
3. J. Kua, G. Armitage, and P. Branch, *A survey of rate adaptation techniques for dynamic adaptive streaming over http*, IEEE Commun. Surv. Tutorials **19** (2017), no. 3, 1842–1866.
4. S. E. Co, *5G vision*, white paper, 2015.
5. *GSM Association, The mobile economy 2022*, 2022.
6. E. Ko, D. An, I. Yeom, and H. Yoon, *Congestion control for sudden bandwidth changes in TCP*, Int. J. Commun. Syst. **25** (2012), no. 12, 1550–1567.
7. P. Li, X. Jiang, G. Jin, Y. Yu, and Z. Xie, *ALSTM: An attention-based LSTM model for multi-scenario bandwidth prediction*, (IEEE 27th International Conference on Parallel and Distributed Systems (ICPADS), Beijing, China), 2021, pp. 98–105.
8. S. Goutam and S. Unnikrishnan, *Decision for vertical handover based on naive Bayes algorithm*, (International Conference on Advances in Computing, Communication and Control (ICAC3), Mumbai, India), 2019, pp. 1–6.
9. Z.-H. Huang, Y.-L. Hsu, P.-K. Chang, and M.-J. Tsai, *Efficient handover algorithm in 5G networks using deep learning*, (Globecom 2020-2020 IEEE Global Communications Conference, Taipei, Taiwna), 2020, pp. 1–6.
10. J. Lima, A. Medeiros, E. Aguiar, V. A. D. S. Junior, and T. Guerra, *User-level handover decision making based on machine learning approaches*, J. Commun. Inform. Syst. **37** (2022), no. 1, 104–108.
11. N. Han, S. Qiao, D. Liu, P. Ding, Y. Zhang, X. Xiong, M. Wang, and L. A. Gutierrez, *Handover detection approach based on trajectory data mining techniques*, J. Eng. **2018** (2018), no. 16, 1534–1537.
12. T. Huang and J. Subhlok, *Fast pattern-based throughput prediction for TCP bulk transfers*, (CCGRID 2005. IEEE International Symposium on Cluster Computing and the Grid, Cardiff, UK), 2005, pp. 410–417.
13. J.-H. Hwang and C. Yoo, *Formula-based TCP throughput prediction with available bandwidth*, IEEE Commun. Lett. **14** (2010), no. 4, 363–365.
14. M. Mirza, J. Sommers, P. Barford, and X. Zhu, *A machine learning approach to TCP throughput prediction*, IEEE/ACM Trans. Netw. **18** (2010), no. 4, 1026–1039.
15. J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, *Modeling TCP reno performance: a simple model and its empirical validation*, IEEE/ACM Trans. Netw. **8** (2000), no. 2, 133–145.
16. N. Parvez, A. Mahanti, and C. Williamson, *An analytic throughput model for TCP new reno*, IEEE/ACM Trans. Netw. **18** (2009), no. 2, 448–461.
17. L. Mei, R. Hu, H. Cao, Y. Liu, Z. Han, F. Li, and J. Li, *Realtime mobile bandwidth prediction using LSTM neural network and Bayesian fusion*, Comput. Netw. **182** (2020), 107515.
18. B. Wei, W. Kawakami, K. Kanai, J. Katto, and S. Wang, *TRUST: a TCP throughput prediction method in mobile networks*, (IEEE Global Communications Conference (GLOBECOM, Abu Dhabi, United Arab Emirateds), 2018, pp. 1–6.
19. N. Nayakwadi and R. Fatima, *Machine learning based handover execution algorithm for heterogeneous wireless networks*, (Fifth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Bangalore, India), 2020, pp. 54–58.
20. D. Yun, W. Liu, C. Q. Wu, N. S. V. Rao, and R. Kettimuthu, *Performance prediction of big data transfer through experimental analysis and machine learning*, (IFIP Networking Conference (Networking), Paris, France), 2020, pp. 181–189.
21. UCC MISL 5G dataset. https://github.com/uccmisl/5Gdataset Accessed: 2022-10-25.
22. D. Raca, D. Leahy, C. J. Sreenan, and J. J. Quinlan, *Beyond throughput, the next generation: a 5g dataset with channel and context metrics*, (Proceedings of the 11th ACM multimedia systems conference, Istanbul, Turkey), 2020, pp. 303–308.
23. G-NetTrack Pro. https://gyokovsolutions.com/manual-g-nettrack/ Accessed: 2023-3-13.
24. N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, *On large-batch training for deep learning: generalization gap and sharp minima*, arXiv preprint, 2016. https://doi.org/10.48550/arXiv.1609.04836
25. D. Masters and C. Luschi, *Revisiting small batch training for deep neural networks*, arXiv Preprint, 2018. https://doi.org/10.48550/arXiv.1804.07612
26. 3GPP TS 36.214 evolved universal terrestrial radio access (e-utra); physical layer; measurements (version 14.2.0 release 14), 2017. 3GPP.
27. J. Um, I. Kim, and S. Park, *Implementation of platform for long-term evolution cell perspective resource utilization analysis*, ETRI J. **43** (2021), no. 2, 232–245.

AUTHOR BIOGRAPHIES

Hyeonji Lee received the BS degree in computer engineering from Changwon National University, Changwon, Republic of Korea, in 2022. She is currently a master's candidate at the Department of Computer Engineering, Changwon National University. Her research interests include low-latency transport layer protocols and deep learning.

Yoohwa Kang received the MS degree from the Pohang University of Science and Technology, Pohang, Republic of Korea, in 2000, and the PhD degree from the Korea Advanced Institute of Science and Technology, Deajeon, Republic of Korea, in 2021. She has worked for the Electronics and Telecommunications Research Institute in Daejeon, Republic of Korea, since 2000. Her main research interest is the standardization of the network architecture for 5G-Advanced and 6G mobile technologies.



Minju Gwak received the BS and MS degrees in computer engineering from Changwon National University, Changwon, Republic of Korea in 2021 and 2023, respectively. She is currently an Engineer at DSME Information & Consulting, Geoje, Republic of Korea. Her research interests include traffic classification, deep learning, and low-latency communications.

Donghyeok An received the BS degree in computer science from Handong University, Pohang, South Korea, in 2006, and the PhD degree in computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea in 2013. He was a postdoctoral and visiting professor at Sungkyunkwan University, Suwon, Republic of Korea, from 2013 to 2014. He was a senior engineer at Samsung Electronics, Republic of Korea from 2014 to 2015 and an assistant professor at the Department of Computer Engineering, Keimyung University, Daegu, Republic of Korea, from 2015 to 2017. He is currently an associate professor at the Department of Computer Engineering, Changwon National University, Changwon, Republic of Korea. His research interests include low-latency protocols, 5G core architectures, time-sensitive networks (TSNs), and machine learning for communication networks.

How to cite this article: H. Lee, Y. Kang, M. Gwak, and D. An, *Bi-LSTM model with time distribution for bandwidth prediction in mobile networks*, ETRI Journal (2023), 1–13. DOI 10.4218/etrij.2022-0459