IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 20, 2019, accepted July 8, 2019, date of publication July 11, 2019, date of current version July 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2928216

Prioritization of Mobile IoT Data Transmission Based on Data Importance Extracted From Machine Learning Model

YUICHI INAGAKI, RYOICHI SHINKUMA, TAKEHIRO SATO, AND EIJI OKI
Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan

Corresponding author: Yuichi Inagaki (yinagaki@icn.cce.i.kyoto-u.ac.jp)

This work was supported in part by the Japan Science and Technology Precursory Research for Embryonic Science and Technology (JST PRESTO) under Grant JPMJPR1854, and in part by the KAKENHI, Japan Society for the Promotion of Science (JSPS), under Grant JP17H01732.

ABSTRACT Predicting real-time spatial information from data collected by the mobile Internet of Things (IoT) devices is one solution to the social problems related to road traffic. The mobile IoT devices for real-time spatial information prediction generate an extremely high volume of data, making it impossible to collect all of it through mobile networks. Although some previous works have reduced the volume of transmitted data, the prediction accuracy of real-time spatial information is still not ensured. Therefore, this paper proposes an IoT device control system that reduces the amount of transmitted data used as input for real-time prediction while maintaining the prediction accuracy. The main contribution of this paper is that the proposed system controls data transmission from the mobile IoT devices based on the importance of data extracted from the machine learning model used for the prediction. Feature selection has been widely used for extracting the importance of data from the machine learning model. Feature selection methods were also used to reduce communication overhead in distributed learning. Unlike the conventional usage of feature selection methods, the proposed system uses them to control the data transmission of the mobile IoT devices with priority. In this paper, the proposed system is evaluated with a real-world vehicle mobility dataset in two practical scenarios using the random forest model, which is an extensively used machine learning model. The evaluation results show that the proposed system reduces the amount of transmitted input data for real-time prediction while achieving the same level of prediction accuracy as benchmark methods.

INDEX TERMS Real-time spatial information, vehicular IoT, data prioritization, machine learning, feature selection.

I. INTRODUCTION

The increasing impact of social problems related to road traffic is a major concern facing our future society. Traffic accidents are still a major problem in many societies today. According to a report on road safety by the World Health Organization (WHO), road traffic injuries are currently estimated to be the ninth leading cause of death across all age groups globally and are predicted to become the seventh leading cause of death by 2030 [1]. It also states that 3% of the global GDP is estimated to be lost as a result of road traffic deaths and injuries. Road traffic congestion is another serious problem in many countries. A report by the Centre for Economics and Business Research suggests that the total economy-wide cost across four advanced countries (the UK, France, Germany, and the USA) was $200.7 billion in 2013, and is forecasted to rise to $293.1 billion by 2030 [2].

Predicting real-time spatial information from data collected by mobile Internet of Things (IoT) sensors is one solution to solve the social problems related to road traffic [3]. Mobile IoT devices such as smart cars (including autonomous cars), smartphones, wearable devices, and unmanned aerial vehicles (UAVs) play a major role in such an application: namely, they work to collect data. Some studies have discussed algorithm design for collecting data from sensors on vehicles using mobile crowdsensing [4], [5]. The data collected by mobile IoT devices are uploaded to edge servers, which process the uploaded data and apply machine learning techniques to predict real-time spatial information such as road-traffic volume, optimal travel path, and precise positions

The associate editor coordinating the review of this manuscript and approving it for publication was Min Jia.

of pedestrians and cyclists. Real-time spatial information prediction is in demand for many services. An example service is the autonomous driving support system, which gathers real-time data from onboard sensors and provides exact location and relation to other road users [6]. The cyber-physical system (CPS), in which the real-time spatial information prediction system is included, is increasingly in demand. The market was worth $18 billion in 2017 and is likely to grow by 8.7% annually for the next ten years [7].

However, mobile IoT devices for real-time spatial information prediction collect an enormous amount of upstream data — much more than can be collected through the uplink bandwidth in mobile networks. Mobile IoT devices collect images, videos, or light detection and ranging (LiDAR) [8] data continuously, and it is impossible to collect all of such data through the uplink bandwidth of long term evolution (LTE) or LTE-Advanced (LTE-A) networks today. Even with 5G networks, it is impossible to collect all of the high-resolution images, videos, and LiDAR data.

Cluster-based data aggregation reduces data transmission by clustering wireless sensors and aggregating raw data from each cluster before transmitting them to destined targets [9]–[12]. Sensors clustered into one cluster are usually located nearby each other, so collected data from these sensors are correlated and thus redundant to some extent. Cluster-based data aggregation eliminates this redundancy, thereby reducing the volume of data transmission. This approach focuses mainly on redundancy in data; no previous work has successfully reduced the volume of transmitted data used as input for real-time prediction while maintaining the prediction accuracy of real-time spatial information.

This work proposes an IoT device control system that reduces the volume of transmitted data used as input for real-time prediction while maintaining the prediction accuracy of real-time spatial information. The main contribution of this paper is that the proposed system prioritizes the transmissions of data collected by mobile IoT devices on the basis of the ''importance of data'' extracted from the machine learning model for prediction. The importance of data is a metric of how much the data collected by mobile sensors will contribute to the prediction accuracy of real-time spatial information. Feature selection has been widely used to extract the importance of data from the machine learning model. Feature selection methods were originally used to reduce computation time, improve prediction performance, or provide a better understanding of the data in machine learning or pattern recognition applications. Feature selection methods were also used to reduce communication overhead in distributed learning [13]. Unlike those conventional usages, the proposed system uses feature selection methods to control the data transmission of mobile IoT devices with priority. In this work, two performance evaluations are performed using real-world datasets, with each one assuming a different scenario. These evaluations use a Random Forest regressor [14] as the machine learning model for prediction and the impurity method [15] and perturb method [16] as feature selection methods. The results of these evaluations show that the proposed system reduces the volume of input data transmission for real-time prediction compared with benchmark methods while achieving the same prediction accuracy.

The rest of this paper is organized as follows. Section II reviews the prior efforts on data reduction in wireless sensor networks (WSNs). In Section III, existing feature selection methods that can be used to extract the importance of data from machine learning models are introduced. Section IV presents the problem formulation of this study and the details of the proposed system. Sections V and VI provide performance evaluations with scenarios of road-traffic volume prediction and mobility demand prediction, respectively. We conclude in Section VII with a brief summary and mention of future work.

II. RELATED WORK

This section reviews the prior efforts on data reduction in WSNs as the related work. In simple terms, data reduction in WSNs aims to reduce the volume of data to be delivered to the sink. Data reduction leads to increased energy efficiency of WSNs because less data transmission means less energy consumption in many WSNs. Data reduction techniques can be placed into four categories [17]: 1) aggregation, 2) adaptive sampling, 3) network coding, and 4) data compression. Aggregation, network coding, and data compression focus mainly on reducing data but do not pay much attention to the application after the data aggregation. Although adaptive sampling techniques reduce data considering the application requirements, this work differs in that the proposed system reduces data according to the importance of the data calculated directly from the machine learning model.

A. AGGREGATION

Data aggregation is defined as the process of aggregating the data from multiple sensors for the purpose of eliminating redundant transmission and providing aggregated information to the base station (BS) [18]. In aggregation techniques, nodes along a path towards the BS perform data aggregation to reduce the volume of data forwarded towards the BS. Aggregation techniques can be roughly categorized into cluster-based and non-cluster-based.

Cluster-based aggregation techniques construct sensor clusters where sensors transmit data to a local aggregator or cluster head (CH) and the CH aggregates data from all the sensors in its cluster and transmits the aggregated data to the sink. Ma et al. presented an algorithm that constructs a dominating set by using the spatial correlation between data measured by different sensors [10]. Yue et al. presented an algorithm called energy efficient and balanced cluster-based data aggregation (EEBCDA) [11]. This algorithm assumes that sensor data in the same cluster are highly correlated so that each CH is able to aggregate some data and reduce the overall volume of the data.

Non-cluster-based aggregation techniques are basically data aggregation without clustering. Azim et al. presented a technique called smart aggregation (SAG) for continuous-monitoring applications [19]. Jiang et al. presented a data aggregation technique designed on the basis of statistical information extraction [20].

B. OTHER METHODS

Adaptive sampling methods adjust the sampling rate at each sensor while preserving the application requirements such as coverage or information precision. For example, a supervision application can reduce data by using low-power and low-precision detectors under a normal condition and switching on power-consuming and high-precision cameras only when an event is reported [21]. Yan et al. presented an activity recognition application that adjusts the sampling rate of user activity to reduce the redundant data when the user is sitting or running [22].

Network coding (NC) reduces the traffic in broadcast scenarios by sending a linear combination of several packets instead of sending a copy of each packet. NC takes advantage of the fact that communications are slow compared to computations and more power-consuming. Wang et al. combined network coding with a connected dominating set [23]. Hou et al. presented adapted network coding in which a node sends one message for every N messages received when broadcasting, saving up to $(N-1)/N$ of bandwidth [24].

Data compression reduces the number of bits needed to represent the message by applying sophisticated encoding methods. Since most existing compression algorithms cannot be applied to WSNs due to the resource limitation of sensor nodes, Kimura et al. reviewed compression algorithms specifically designed for WSNs [25].

III. FEATURE SELECTION

Feature selection was originally considered as a method for selecting a set of variables (features) from the input that can efficiently describe the input data while reducing effects from noise or irrelevant variables and still provide good prediction results [26]. Feature selection methods can reduce computation time, improve prediction performance, and provide a better understanding of the data in machine learning or pattern recognition applications. Feature selection differs from other dimension reduction methods such as principal component analysis (PCA) in that it does not create new features since it uses the input features themselves to reduce their number.

The proposed system prioritizes the transmissions of data collected by mobile IoT devices on the basis of the importance of data extracted from the machine learning model for prediction using feature selection. The importance of data extracted from the machine learning model using feature selection is a metric of how much the collected data by mobile sensors will contribute to the prediction accuracy of real-time spatial information. Reducing transmission of less important data according to the importance of data obtained from feature selection methods enables the proposed system to reduce the volume of transmitted data used as input for real-time prediction while maintaining the prediction accuracy. Details on how feature selection is used in the proposed system in specific scenarios are described in Sections V and VI.

IV. PROPOSED SYSTEM DESIGN

A. APPLICATION SCENARIO

The overview of the proposed system is shown in Fig. 1. We assume a system that provides users with real-time spatial information based on data collected from mobile IoT devices. The proposed system prioritizes data on mobile IoT devices on the basis of data importance extracted from the machine learning model for prediction, which enables it to reduce the total data traffic for real-time prediction while maintaining prediction accuracy.

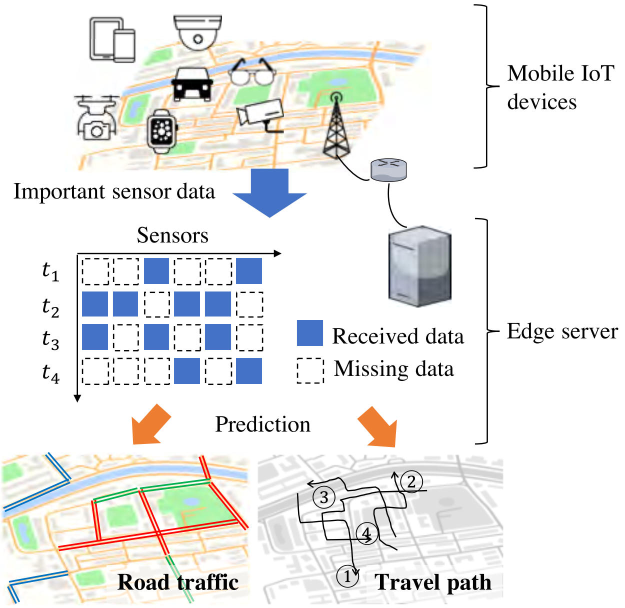


FIGURE 1. System overview.

The proposed system consists of two main components: mobile IoT devices and an edge server. Mobile IoT devices (such as probe vehicles, smartphones, and UAVs) prioritize collected data and send high importance input data for prediction to the edge server. The edge server aggregates the data received from mobile IoT devices, complements the missing parts of the data, and performs prediction.

Note that the machine learning model has been trained in advance with all the available data collected by mobile IoT devices. This assumption is acceptable because our work aims to reduce the volume of data used as input for real-time prediction. Furthermore, since the time requirement of training data is not strict, they can be collected as a background process through mobile networks during the off-peak period or through other communication networks with sufficient bandwidth.

B. SYSTEM MODEL

A detailed view of the proposed system is shown in Fig. 2.

1) MOBILE IoT DEVICE

Mobile IoT devices (such as probe vehicles, smartphones, and UAVs) continuously collect data at a specific sampling

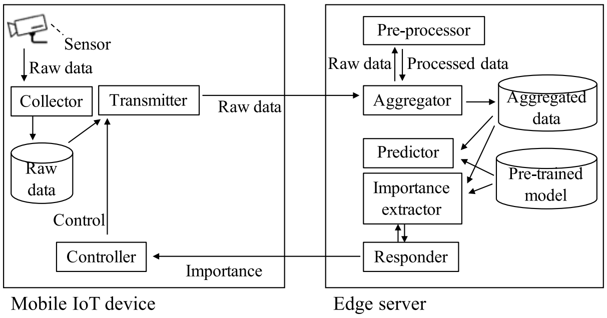


FIGURE 2. System design.

interval. Part of the collected data is sent to an edge server for prediction. To decide whether a data should be sent or not, the controller fetches the importance of the block where the mobile IoT device is currently located. The controller orders the transmitter to send the data or not based on the importance of the current block received from the edge server.

2) EDGE SERVER

The edge server receives collected data from mobile IoT devices, aggregates and preprocess the data, and predicts the real-time spatial information of the next time slot. The aggregator and pre-processor on the edge server receive data from the transmitters of several mobile IoT devices, pre-process the data, and complete the missing parts of the data. The predictor predicts real-time spatial information. The importance extractor extracts the importance of blocks from the pre-trained machine learning model for prediction. The responder sends the importance of blocks to each mobile IoT device.

C. CONTROL METHODS

The control procedure of the proposed system consists of five processes: 1) pre-training of the machine learning model, 2) calculation of the importance of blocks, 3) control of data transmission, 4) aggregation of transmitted data, and 5) prediction. 1) and 2) are preprocessing, which is performed once before the first time slot begins. 3), 4), and 5) are performed in each time slot. Table 1 lists the notations in this paper.

TABLE 1. Notations.

	Description
$D(\bar{I})$	Set of data received from mobile IoT devices under data importance threshold $\bar{I}$
$S(\cdot)$	Size of data
$A(\cdot)$	Prediction accuracy
$\bar{A}$	Required prediction accuracy
X	Input variable of the machine learning model for prediction ($T \times N_B$ matrix)
y	Output variable of the machine learning model for prediction (vector of N_B elements)
T	No. of time slots used in one prediction
N_B	No. of blocks
$F_{t,b}$	Feature importance of (t, b) element of the input variable
I_b	Importance of the block b
$\bar{I}$	Threshold for importance of transmitted data

1) PRE-TRAINING OF MACHINE LEARNING MODEL

The machine learning model is trained on the edge server in advance before the first time slot begins.

The proposed system considers the prediction of future real-time spatial information as a regression task. Regression, in general, is a type of task that estimates a numerical value given some input [27]. To solve the task, the learning algorithm is asked to learn a function that maps an input variable to an output variable. The proposed system uses a supervised machine learning model to solve the task. Supervised learning algorithms, in general, deal with a training dataset that contains a set of data and a label or target associated with each of the data [27].

In the proposed system, the machine learning model for prediction receives the aggregated past sensor data collected from mobile IoT devices in each block as an input and calculates the future real-time spatial information as output. The machine learning model for prediction receives an input variable X, which consists of aggregated sensing data collected in the last few time slots, and predict output variable y, which is the real-time spatial information of each block in the next time slot. The input variable X of the machine learning model for prediction is a $T \times N_B$ matrix, where T is the number of time slots used in one prediction and N_B is the number of blocks. Each row of the input matrix represents the aggregated sensor data collected in N_B blocks at $T, T-1, \ldots, 1$ slots ago, respectively. The output y of the machine learning model for prediction is a vector of N_B elements. Each element of the output vector represents the real-time spatial information of each block in the next time slot.

As mentioned in Section 4.1, to train a machine learning model for prediction, an adequate amount of past sensing data should be collected from mobile IoT devices as training data, but it does not necessarily need to be collected in real time. Thus, the training data can be collected when the network is off-peak, such as at night or when cars or drones are stopped in parking lots or depots.

2) CALCULATION OF IMPORTANCE OF BLOCKS

The importance of a block is calculated from the pre-trained machine learning model for prediction on the edge server using a feature selection method. The importance of block I_b is defined as $I_b = \sum_{t=1}^{T} F_{t,b}$, where $F_{t,b}$ is the feature importance of the (t, b) element of the input matrix. $F_{t,b}$ is calculated from the pre-trained model using the feature selection method.

3) CONTROL OF DATA TRANSMISSION

In each time slot, mobile IoT devices decide whether to transmit the data observed in the time slot based on the importance of the data. The importance of data corresponds to the importance of the block where the data was observed. Mobile IoT devices transmit the data if $I_b \geq \bar{I}$, where I_b is the importance of the block that includes the current location of the device and $\bar{I}$ is a constant that defines the minimum importance for the data to be transmitted. I_b and $\bar{I}$ are obtained from the edge server.

The proposed system can control the volume of the transmitted data and prediction accuracy through $\bar{I}$. The volume of the transmitted data in a single time slot can be described as

$$\sum_{d \in D(\bar{I})} S(d), \tag{1}$$

where $D(\bar{I})$ is the set of transmitted data from mobile IoT devices and $S(d)$ is the size of data d. $D(\bar{I})$ includes data from a mobile IoT device if and only if $I_b \geq \bar{I}$, where I_b is the importance of the block in which that device is located. The prediction accuracy in a time slot can be described as

$$A(D(\bar{I})), \tag{2}$$

where $A(\cdot)$ is the prediction accuracy when given a set of sensing data from mobile IoT devices. The prediction accuracy depends on the data received from mobile IoT devices in the time slot.

4) AGGREGATION OF TRANSMITTED DATA

The data collected from mobile IoT devices are aggregated to form an input matrix X for the machine learning model for prediction. The aggregation is needed because the proposed system does not always collect exactly one data from each block. The number of data collected from each block varies depending on the importance of the block and the number of mobile IoT devices in the block. An example of the aggregation process can be found in Section V-B.

5) PREDICTION

The proposed system uses the pre-trained model to predict future spatial-information. In each time slot, the model takes the aggregated sensing data as an input X and predicts the real-time spatial information of the next time slot as an output y.

V. PERFORMANCE EVALUATION BY ROAD-TRAFFIC VOLUME PREDICTION

A. EVALUATION SCENARIO

An evaluation was performed to verify the effectiveness of the proposed system described in Section IV, which reduces the total traffic for real-time prediction transmitted from mobile IoT devices while maintaining the prediction accuracy. This evaluation examines the relationship between the amount of transmitted data and prediction accuracy described in Eqs. (1) and (2) respectively for several $\bar{I}$.

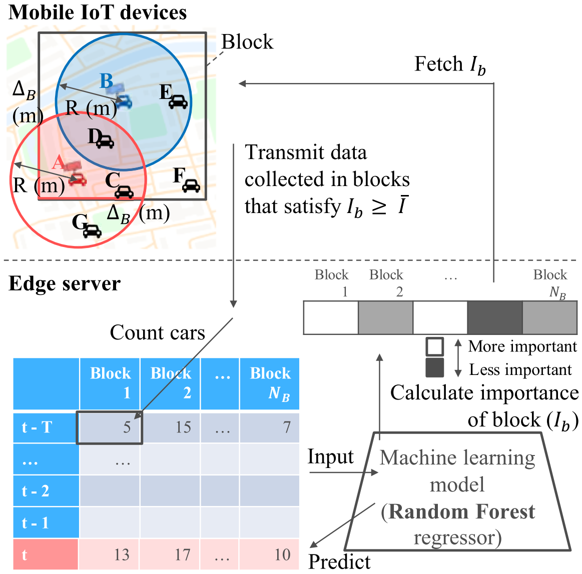
This performance evaluation focuses on a specific application that provides human or robotic drivers with road-traffic information predicted from sensing data collected by onboard cameras or LiDARs on probe vehicles. Parameters used in this evaluation are listed in Table 2.

TABLE 2. Parameters for performance evaluation by road-traffic volume prediction.

Parameter	Value
Sampling interval (Δ_T)	1 (minute)
Sampling intervals in one time slot (T)	60
Radius in which probe vehicles can detect cars (R)	50 (m)
Size of block (Δ_B)	1000 × 1000, 500 × 500 (m^2)
No. of blocks (N_B)	144, 576
Percentage of probe vehicles (P)	20 (%)
No. of estimators in Random Forest (N_E)	100

B. EVALUATION MODEL

Figure 3 shows the evaluation model used for this evaluation. It consists of mobile IoT devices and an edge server.



	Block 1	Block 2	…	Block N_B
t - T	5	15	…	7
…	…			
t - 2				
t - 1				
t	13	17	…	10

FIGURE 3. Evaluation model for performance evaluation by road-traffic volume prediction.

1) PRE-TRAINING OF MACHINE LEARNING MODEL

The Random Forest regressor model in the scikit-learn library [28] is used as the machine learning model for prediction for this evaluation. The input X of the model is $T \times N_B$ matrix, where T is the number of data samples in one time slot and N_B is the number of blocks. Each element x_{ij} in the matrix represents the aggregated road-traffic in block j at time slot $t - i$, where t is the current sampling time. The output y of the model is the road-traffic of each block at sampling time t.

The model is trained with road-traffic data of all 536 taxies before the evaluation. The road-traffic data of all 25 days is split into the first 20 days and the last five days for training and evaluation, respectively. The details on the dataset are described in Section V-C.

2) CALCULATION OF IMPORTANCE OF BLOCKS

Two feature selection methods are used to calculate the importance of blocks: the impurity method and the perturb method. The impurity method calculates feature importance on the basis of the 'impurity' index used in decision tree models [15]. The impurity method in this evaluation is implemented by the `feature_importances_` function of the Random Forest regressor of scikit-learn. By applying this function, the importance of each input feature, i.e., the importance of each element x_{ij}, is obtained. To simplify the evaluation, we calculate the importance of blocks by taking the sum for i. The perturb method calculates feature importance by adding noise to the subset of input features and examining the increase of error [16]. The perturb method calculates the importance of block j by

$$(RMSE(\hat{y}', y) - RMSE(\hat{y}, y))^2, \tag{3}$$

where y is the number of cars in blocks, $\hat{y}$ is the predicted value of y, and $\hat{y}'$ is the predicted value when the input values of block j in the training data are multiplied by 1.5.

3) CONTROL OF DATA TRANSMISSION

In each time slot, probe vehicles transmit the collected sensing data if and only if $I_b \geq \bar{I}$, where I_b is the importance of the block in which a probe vehicle is currently located. Probe vehicles know the I_b of each block in advance.

The number of cars can be detected from sensing data collected by onboard cameras or LiDARs using an object detection algorithm at the pre-processor on the edge server. In this evaluation, we streamlined this process and obtained directly the number of cars from an existing dataset.

4) AGGREGATION OF TRANSMITTED DATA

It is assumed that an aggregator on the edge server receives raw sensing data from probe vehicles and a pre-processor on the edge server identifies cars that are running around each probe vehicle. The number of detected cars in the block at the sampling time is defined as the size of the set plus 1. If multiple probe vehicles are in the block at the sampling time, this number is defined as the size of the union of sets of cars plus the number of probe vehicles. If no probe vehicles are in the block at the sampling time, zero-filling is used to complete the missing parts of data.

5) PREDICTION

Prediction is performed in each time slot using the pre-trained model described in Section V-B.1.

C. DATASET

A trace set of the mobility data of taxi cabs in San Francisco [29] is used in this evaluation. The dataset includes the location logs of 536 taxies for 25 days. N_B blocks in total are positioned in a rectangle area, as shown in Fig. 4. Since the logs are not necessarily recorded every Δ_T minutes, taxies are assumed to travel straight with constant velocity, and locations at every Δ_T minute are interpolated. Probe vehicles are selected randomly from a total of 536 taxies at a ratio of P. The number of cars detected by probe vehicles in each block at each sampling time is calculated as described in the previous section. A block that contains the taxi company depot is ignored because the block does not seem to generate data appropriate for evaluation. Figure 5 shows the average number of detected cars in each block.

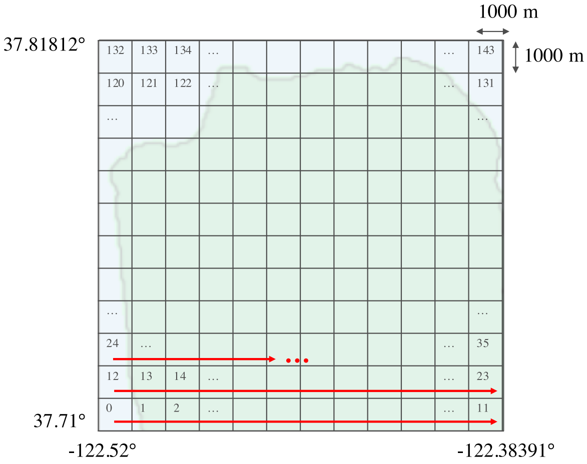


FIGURE 4. Layout of blocks (Δ_B = 1000 × 1000 (m^2)).

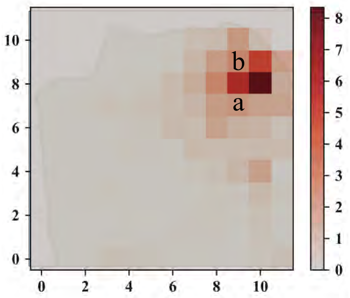


FIGURE 5. Average no. of detected cars in each block.

D. METRICS AND BENCHMARKS

This evaluation verifies that the proposed system reduces the amount of transmitted data used as input for real-time prediction compared with three benchmark methods when they achieve the same prediction accuracy by examining the relationship between the amount of transmitted data and prediction accuracy described in Eqs. (1) and (2) respectively. The prediction error and the total amount of data transmission are calculated for each $\bar{I}$.

To evaluate the prediction error, the root mean squared log error (RMSLE) [30] function is used. RMSLE is given by

$$RMSLE = \sqrt{\frac{1}{N_B}\sum_{b=1}^{N_B}(\log(y_b + 1) - \log(\hat{y_b} + 1))^2}, \tag{4}$$

where y_b is the number of cars running in block b and $\hat{y_b}$ is the predicted value of y_b.

To evaluate the total amount of data transmission, the normalized total amount of transmitted data is used. The normalized total amount of transmitted data r is the ratio of the total amount of sensing data transmitted to the total amount of sensing data collected by probe vehicles.

This evaluation uses three benchmark methods: random, uniform, and volume-based. The random method selects which block to use at random. This is a reasonable method because, in general, the data transmitted by mobile IoT sensors are usually dropped randomly when network capacity is limited. The uniform method selects n_B blocks out of the total N_B blocks uniformly. To select blocks uniformly, the uniform method spirally assigns numbers $0 \le k < N_B$ to each block. The set of numbers of selected blocks is decided by $\{k = \lfloor (nN_B)/n_B \rfloor \mid 0 \le \exists n < n_B\}$. The volume-based method selects blocks with the top n_B largest average road-traffic volume in the training dataset.

E. RESULTS

Figures 6 and 7 show the importance of each block extracted from the Random Forest regressor by using the impurity and perturb feature selection methods, respectively. Compared with the average number of detected cars in each block in Fig. 5, blocks with a large road-traffic volume tend to be important. However, the blocks whose adjacent blocks have large road-traffic volume (e.g., a, b) tend to be less important compared to their own road-traffic volume. This is because the data from two adjacent blocks are redundant to some extent. In general, the road-traffic volumes of two adjacent blocks correlate with each other. The impurity and perturb methods reflect this principle and avoid assigning high importance to two adjacent blocks in order to eliminate that redundancy.

Figure 8 shows the prediction error against the normalized total amount of transmitted data when the block size is $\Delta_B = 1000 \times 1000$. RMSLE is larger as r is smaller for all the methods. This describes the trade-off between the amount of data available for prediction and the accuracy of prediction. RMSLE of the random and uniform methods fluctuates as r changes. This is because only a small number of blocks mainly contribute to the prediction accuracy, and thus RMSLE of the random and uniform methods depends greatly on whether those blocks are selected or not. In contrast, RMSLE of the impurity and perturb methods is stably small for a wide range of r. This is because the proposed system with the impurity or perturb methods always prioritizes the data from probe vehicles in important blocks, which contributes to the prediction accuracy. RMSLE of the impurity and perturb methods is better than that of the volume-based method for a wide range of r. This is because, as observed in Figs. 5 – 7, blocks with high average road-traffic volume do not always have high importance. Prioritizing the transmissions on the basis of the average road-traffic volume of each block leads to redundant data transmission. By using the importance of blocks, the impurity and perturb methods avoid redundant data transmission, and thus the proposed system can achieve better RMSLE than that of the volume-based method.

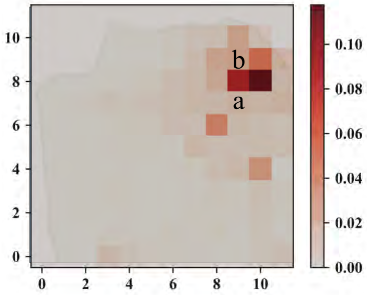


FIGURE 6. Extracted importance of each block for road-traffic volume prediction (impurity).

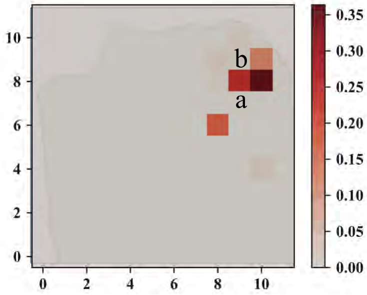


FIGURE 7. Extracted importance of each block for road-traffic volume prediction (perturb).

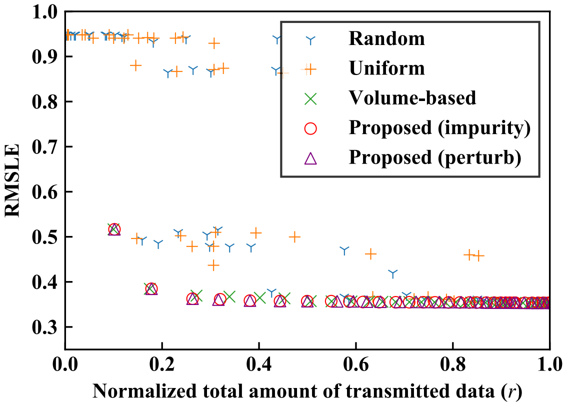


FIGURE 8. Prediction error vs. normalized total amount of transmitted data for road-traffic volume prediction ($\Delta_B = 1000 \times 1000$ (m^2)).

Figure 9 shows the prediction error against the normalized total amount of transmitted data when the block size is $\Delta_B = 500 \times 500$. The same observations can basically be obtained as in Fig. 8, while the scale of RMSLE of Fig. 9 is smaller as a whole than that of Fig. 8. This is because, in general, the prediction error tends to be small when the scale of predicted values is small. The scale of predicted values in this evaluation was smaller when $\Delta_B = 500 \times 500$ than when $\Delta_B = 1000 \times 1000$ because smaller blocks have smaller road-traffic volume.

VI. PERFORMANCE EVALUATION BY MOBILITY DEMAND PREDICTION

A. EVALUATION SCENARIO

This evaluation focuses on a specific application in which the system predicts the number of pickups by taxis on the basis of people detection from sensing data collected by probe

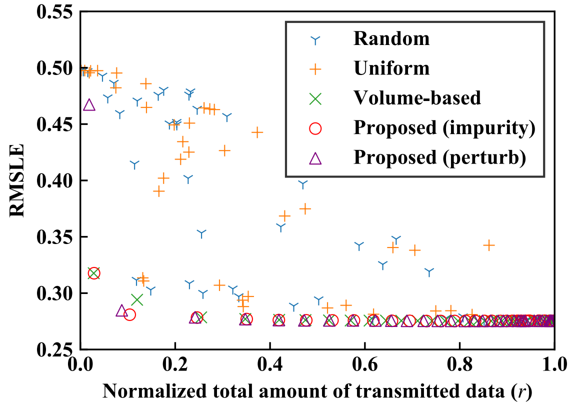


FIGURE 9. Prediction error vs. normalized total amount of transmitted data for road-traffic volume prediction ($\Delta_B = 500 \times 500$ (m^2)).

vehicles. The purpose of this evaluation is the same as that of performance evaluation in Section V. Parameters used are listed in Table 3.

TABLE 3. Parameters for performance evaluation by mobility demand prediction.

Parameter	Value
Sampling interval (Δ_T)	1 (minute)
Sampling intervals in one time slot (T)	60
Size of block (Δ_B)	1000 × 1000, 500 × 500 (m^2)
No. of blocks (N_B)	144, 576
No. of estimators in Random Forest (N_E)	100

B. EVALUATION MODEL

1) PRE-TRAINING OF MACHINE LEARNING MODEL

The Random Forest regressor model in the scikit-learn library [28] is used as the machine learning model for prediction for this evaluation. The shape of input X of the model is the same as the input described in Section V, except that each element x_{ij} in the matrix represents the number of pickups in block j at time slot $t - i$, where t is the current sampling time. The output y of the model is the number of pickups of each block at sampling time t.

The Random Forest regressor is trained with the pickup log data of 536 taxies before the evaluation. The data is split into the first 20 days for training and last five days for evaluation. The details of the dataset are described in Section VI-C.

2) CALCULATION OF IMPORTANCE OF BLOCKS

The impurity and perturb methods described in Section V-B are also used as feature selection methods for the proposed system in this evaluation.

3) CONTROL OF DATA TRANSMISSION

The number of pickups by taxis can be detected from sensing data collected by onboard cameras or LiDARs using an object detection algorithm at the pre-processor on the edge server. In this evaluation, we streamlined this process and obtained directly the number of pickups from an existing dataset as in Section V.

The same as in Section V-B, probe vehicles decide whether or not to transmit the collected data in each time slot.

4) AGGREGATION OF TRANSMITTED DATA

It is assumed that an aggregator on the edge server receives raw sensing data and pickup logs from taxies and a pre-processor on the edge server extracts useful information to predict mobility demand. From the extracted information and pickup logs, the aggregator counts the number of pickups in each block at each time slot. If no probe vehicles are in the block at the sampling time, zero-filling is used to complete the missing parts of data.

5) PREDICTION

Prediction is performed in each time slot using the pre-trained model described in Section VI-B.1.

C. DATASET

A trace set of the mobility data of taxi cabs in San Francisco is used in this evaluation, the same as in V. The mobility log includes occupancy data that represents whether or not a taxi has passengers. This evaluation uses the occupancy of taxies as well as the mobility traces of taxies included in the dataset. This evaluation considers that a pickup occurred when the occupancy value of the log changed from 0 (not-occupied) to 1 (occupied).

The evaluation area is split into N_B blocks, as described in Section V, and the location of taxies is interpolated every Δ_T minute. Figure 10 shows the average number of pickups in each block.

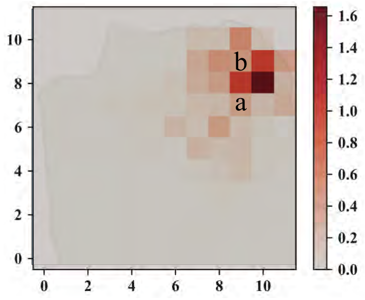


FIGURE 10. Average no. of pickups in each block per minute.

D. METRICS AND BENCHMARKS

This evaluation uses the same metrics and benchmark method as Section V.

E. RESULTS

Figures 11 and 12 show the importance of each block extracted from the Random Forest regressor by using the impurity method and the perturb method, respectively. As observed in Section V-E, there is a difference between the number of pickups in Fig. 10 and the importance of blocks in Figs. 11 and 12.

Figures 13 and 14 show the prediction error against the transmission rate when the block size is $\Delta_B = 1000 \times 1000$ and $\Delta_B = 500 \times 500$, respectively. RMSLE is larger when the transmission rate is smaller for all the methods, which is the same as the trend observed in Figs. 8 and 9.

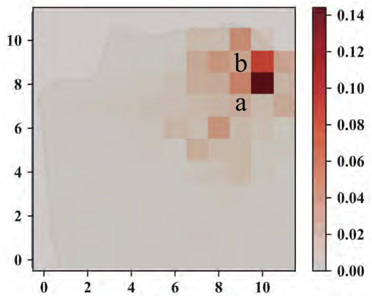


FIGURE 11. Extracted importance of each block for mobility demand prediction (impurity).

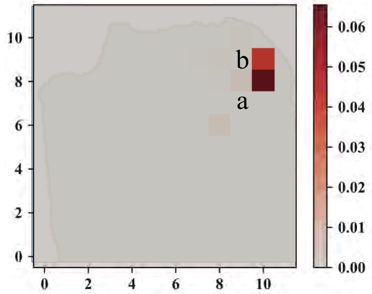


FIGURE 12. Extracted importance of each block for mobility demand prediction (perturb).

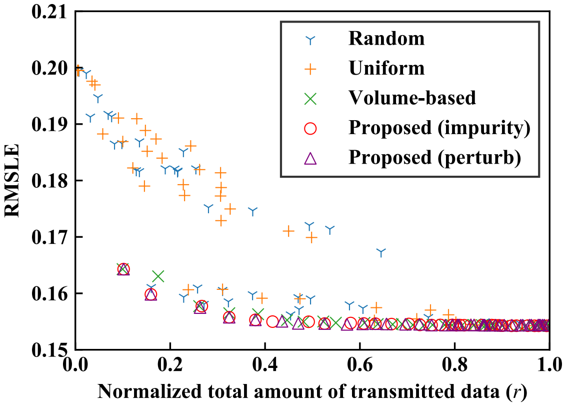


FIGURE 13. Prediction error vs. transmission rate for mobility demand prediction ($\Delta_B = 1000 \times 1000$ (m^2)).

In Figs. 13 and 14, when $r < 0.2$, RMSLE of the proposed system has a relatively larger value than RMSLE in other ranges of the transmission rate r. In contrast, in Figs. 8 and 9, when $r < 0.1$, RMSLE of the proposed system has a relatively larger value than RMSLE in other ranges of r. This is presumably because the number of relatively important blocks is smaller in the road-traffic volume dataset compared to the mobility demand dataset. In the road-traffic dataset in Section V, only a small number of blocks have high importance and many other blocks have low importance. In the road-traffic dataset, RMSLE has a large value especially when $r < 0.1$ because those small numbers of important blocks are dropped when $r < 0.1$. In contrast, in the mobility demand dataset, since the number of relatively important blocks is larger, RMSLE of the proposed system has a relatively large value when $r < 0.2$.

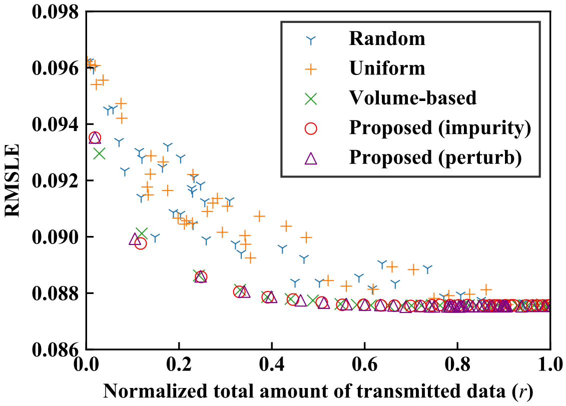


FIGURE 14. Prediction error vs. transmission rate for mobility demand prediction ($\Delta_B = 500 \times 500$ (m^2)).

VII. CONCLUSION AND FUTURE WORK

To reduce the volume of transmitted data used as input for real-time spatial information prediction while maintaining the prediction accuracy, this paper proposed an IoT device control system that uses the importance of data extracted from the machine learning model used for prediction. Importance of data is obtained by measuring how much the collected data by mobile sensors will contribute to the prediction accuracy of real-time spatial information. The proposed system extracts the importance of data by applying feature selection methods. This enables the mobile IoT devices in the proposed system to avoid transmitting less important data (in terms of how much the data contributes to the prediction accuracy) to an edge server. Performance evaluations with road-traffic and mobility-demand prediction scenarios demonstrated that the proposed system reduces the volume of data transmission for real-time prediction while achieving the same level of prediction accuracy as the benchmark methods.

For a more practical evaluation, in future work, other machine learning models along with suitable feature selection methods for the models should be considered. Future work will also include an evaluation with other applications than the ones discussed in this paper.

REFERENCES

[1] *Global Status Report on Road Safety*, World Health Organization, Geneva, Switzerland, 2015.

[2] C. F. Economics and B. Research, "The future economic and environmental costs of gridlock in 2030," INRIX, Kirkland, WA, USA, Tech. Rep., 2014.

[3] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generat. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[4] X. Wang, W. Wu, and D. Qi, "Mobility-aware participant recruitment for vehicle-based mobile crowdsensing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4415–4426, May 2018.

[5] Z. He, J. Cao, and X. Liu, "High quality participant recruitment in vehicle-based crowdsourcing using predictable mobility," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2015, pp. 2542–2550.

[6] H. G. Seif and X. Hu, "Autonomous driving in the iCity—HD maps as a key challenge of the automotive industry," *Engineering*, vol. 2, no. 2, pp. 159–162, Jun. 2016.

[7] *Cyber-Physical System Market: Will China be Able to Surpass Western Europe in Terms of Growth in the Coming Years: Global Industry Analysis (2013—2017) & Opportunity Assessment (2018—2028)*, Future Market Insights, Pune, India, Apr. 2018.

[8] V.-H. Cao, K.-X. Chu, N.-A. Le-Khac, M.-T. Kechadi, D. Laefer, and L. Truong-Hong, "Toward a new approach for massive lidar data processing," in *Proc. IEEE 2nd Int. Conf. Spatial Data Mining Geograph. Knowl. Services (ICSDM)*, Jul. 2015, pp. 135–140.

[9] A. Manjeshwar and D. P. Agrawal, "TEEN: A routing protocol for enhanced efficiency in wireless sensor networks," in *Proc. Parallel Distrib. Process. Symp.*, Apr. 2001, pp. 2009–2015.

[10] Y. Ma, Y. Guo, X. Tian, and M. Ghanem, "Distributed clustering-based aggregation algorithm for spatial correlated sensor networks," *IEEE Sensors J.*, vol. 11, no. 3, pp. 641–648, Mar. 2011.

[11] J. Yuea, W. Zhang, W. Xiao, D. Tang, and J. Tang, "Energy efficient and balanced cluster-based data aggregation algorithm for wireless sensor networks," *Procedia Eng.*, vol. 29, pp. 2009–2015, Feb. 2012.

[12] A. Sinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Hum.-Centric Comput. Inf. Sci.*, vol. 3, no. 1, p. 13, 2013.

[13] I. Czarnowski, "Distributed learning with data reduction," in *Transactions on Computational Collective Intelligence IV*, N. T. Nguyen, Ed. Berlin, Germany: Springer, 2011, pp. 3–121.

[14] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[15] B. H. Menze, B. M. Kelm, R. Masuch, U. Himmelreich, P. Bachert, W. Petrich, and F. A. Hamprecht, "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data," *BMC Bioinf.*, vol. 10, no. 1, p. 213, Jul. 2009.

[16] M. Gevrey, L. Dimopoulos, and S. Lek, "Review and comparison of methods to study the contribution of variables in artificial neural network models," *Ecol. Model.*, vol. 160, no. 3, pp. 249–264, Feb. 2003.

[17] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Comput. Netw.*, vol. 67, pp. 104–122, Jul. 2014.

[18] R. Rajagopalan and P. K. Varshney, "Data aggregation techniques in sensor networks: A survey," *Elect. Eng. Comput. Sci.*, vol. 22, pp. 48–63, Jan. 2006.

[19] M. A. Azim, S. Moad, and N. Bouabdallah, "SAG: Smart aggregation technique for continuous-monitoring in wireless sensor networks," in *Proc. IEEE Int. Conf. Commun.*, May 2010, pp. 1–6.

[20] H. Jiang, S. Jin, and C. Wang, "Parameter-based data aggregation for statistical information extraction in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 8, pp. 3992–4001, Oct. 2010.

[21] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 7, no. 3, pp. 537–568, May 2009.

[22] Z. Yan, V. Subbaraju, D. Chakraborty, A. Misra, and K. Aberer, "Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach," in *Proc. 16th Int. Symp. Wearable Comput.*, Jun. 2012, pp. 17–24.

[23] S. Wang, A. Vasilakos, H. Jiang, X. Ma, W. Liu, K. Peng, B. Liu, and Y. Dong, "Energy efficient broadcasting using network coding aware protocol in wireless ad hoc network," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2011, pp. 1–5.

[24] I.-H. Hou, Y.-E. Tsai, T. F. Abdelzaher, and I. Gupta, "AdapCode: Adaptive network coding for code updates in wireless sensor networks," in *Proc. IEEE INFOCOM*, Apr. 2008, pp. 1517–1525.

[25] N. Kimura and S. Latifi, "A survey on data compression in wireless sensor networks," in *Proc. Int. Conf. Inf. Technol., Coding Comput. (ITCC)*, vol. 2, Apr. 2005, pp. 8–13.

[26] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Elect. Eng.*, vol. 40, no. 1, pp. 16–28, Jan. 2014.

[27] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

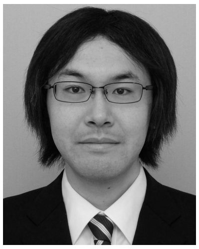
[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[29] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser. (Feb. 2009). *CRAWDAD Dataset EPFL/Mobility (Version 2009-02-24)*. [Online]. Available: https://crawdad.org/epfl/mobility/20090224/cab

[30] M. Zeng, T. Yu, X. Wang, V. Su, L. T. Nguyen, and O. J. Mengshoel, "Improving demand prediction in bike sharing system by learning global features," in *Proc. KDD*, Aug. 2016.

YUICHI INAGAKI received the B.E. degree in electrical and electronic engineering from Kyoto University, in 2017, and the M.E. degree from the Graduate School of Informatics, Kyoto University, in 2018, where he is currently pursuing the Ph.D. degree. He is a member of the IEICE.

RYOICHI SHINKUMA received the B.E., M.E., and Ph.D. degrees in communications engineering from Osaka University, Japan, in 2000, 2001, and 2003, respectively. In 2003, he joined the Faculty of Communications and Computer Engineering, Graduate School of Informatics, Kyoto University, Japan, where he is currently an Associate Professor. He was a Visiting Scholar with the Wireless Information Network Laboratory (WINLAB), Rutgers, the State University of New Jersey, USA, from 2008 to 2009. His research interests include network design and control criteria, particularly inspired by economic and social aspects. He received the Young Researchers' Award from the IEICE, in 2006, the Young Scientist Award from Ericsson Japan, in 2007, the TELECOM System Technology Award from the Telecommunications Advancement Foundation, in 2016, and the Best Tutorial Paper Award from the IEICE Communications Society, in 2019. He was the Chairperson of the Mobile Network and Applications Technical Committee of the IEICE Communications Society, from 2017 to 2019.



TAKEHIRO SATO received the B.E., M.E., and Ph.D. degrees in engineering from Keio University, Japan, in 2010, 2011, and 2016, respectively. From 2016 to 2017, he was a Research Associate with the Graduate School of Science and Technology, Keio University. He is currently an Assistant Professor with the Graduate School of Informatics, Kyoto University, Japan. His research interests include communication protocols and network architectures for the next-generation optical networks. From 2012 to 2015, he was a Research Fellow of the Japan Society for the Promotion of Science. He is a member of the IEICE. From 2011 to 2012, he was a Research Assistant in the Keio University Global COE Program, "High-level Global Cooperation for Leading-edge Platform on Access Spaces," sponsored by the Ministry of Education, Culture, Sports, Science and Technology, Japan.

EIJI OKI received the B.E. and M.E. degrees in instrumentation engineering and the Ph.D. degree in electrical engineering from Keio University, Yokohama, Japan, in 1991, 1993, and 1999, respectively. He was with Nippon Telegraph and Telephone Corporation (NTT) Laboratories, Tokyo, Japan, from 1993 to 2008. From 2000 to 2001, he was a Visiting Scholar with the Polytechnic Institute of New York University, Brooklyn, NY, USA. He was with the University of Electro Communications, Tokyo, from 2008 to 2017. He is currently a Professor with Kyoto University, Japan. His research interests include routing, switching, protocols, optimization, traffic engineering, and system design in communication and information networks.