PAPER Special Section on Wired-and-Wireless Network System Technologies in Beyond 5G/6G

Device Type Classification Based on Two-Stage Traffic Behavior Analysis*

Chikako TAKASAKI[†a)], **Tomohiro KORIKAWA**[†b)], **Kyota HATTORI**[†c)], *and* **Hidenari OHWADA**[†d)], *Members*

SUMMARY In the beyond 5G and 6G networks, the number of connected devices and their types will greatly increase including not only user devices such as smartphones but also the Internet of Things (IoT). Moreover, Non-terrestrial networks (NTN) introduce dynamic changes in the types of connected devices as base stations or access points are moving objects. Therefore, continuous network capacity design is required to fulfill the network requirements of each device. However, continuous optimization of network capacity design for each device within a short time span becomes difficult because of the heavy calculation amount. We introduce device types as groups of devices whose traffic characteristics resemble and optimize network capacity per device type for efficient network capacity design. This paper proposes a method to classify device types by analyzing only encrypted traffic behavior without using payload and packets of specific protocols. In the first stage, general device types, such as IoT and non-IoT, are classified by analyzing packet header statistics using machine learning. Then, in the second stage, connected devices classified as IoT in the first stage are classified into IoT device types, by analyzing a time series of traffic behavior using deep learning. We demonstrate that the proposed method classifies device types by analyzing traffic datasets and outperforms the existing IoT-only device classification methods in terms of the number of types and the accuracy. In addition, the proposed model performs comparable as a state-of-the-art model of traffic classification, ResNet 1D model. The proposed method is suitable to grasp device types in terms of traffic characteristics toward efficient network capacity design in networks where massive devices for various services are connected and the connected devices continuously change.

key words: *device type classification, traffic behavior analysis, machine learning, deep learning*

1. Introduction

Network operators play a fundamental role in providing network connectivity that fulfills service requirements of user devices. In the beyond 5G and 6G eras, the number of connected Internet of Things (IoT) devices will become much larger. In addition, service types of IoT are becoming more diverse such as health-care, manufacturing, and autonomous vehicles. Connected IoT devices numbered 13.2 billion in 2022 and are estimated to increase to 34.7 billion by the end of 2028 [3]. Legacy mobile networks, such as 4G, have limited network requirements, where voice and data communication using mobile phones is a dominant network service. On the other hand, in 5G and 6G, network requirements become more diverse as heterogeneous services, including massive IoT device-based services, need to be provided on the common network infrastructure. In 5G networks, three service types are specified: enhanced mobile broadband (eMBB), massive machine-type communications (mMTCs), and ultra-reliable low-latency communications (URLLCs) [4]. Finer-grained service types may be specified in future networks such as vehicle-to-everything (V2X) communication for autonomous vehicles and remote surgery, which require networks with lower latency and higher reliability. Therefore,it is a challenge to design network capacity to fulfill all service requirements on a single piece of network infrastructure shared by massive user devices.

In the beyond 5G and 6G eras, network demands change largely and rapidly because devices hand off more frequently in accordance with denser base-station placement. The number of situations where network demands change largely and rapidly from ordinary time is also increasing due to natural disasters and events. Moreover, Non-terrestrial networks (NTN) has been considered in 5G standardization, where flying objects such as unmanned aerial vehicles (UAVs), high altitude platform stations (HAPSs), and satellites are used as base stations and relay nodes. NTN may require frequent and continuous network design because network nodes provide connectivity to devices while moving. As a result, network operators need to design network dynamically to handle the massive and rapid changes in connected devicesand network demands.

Network operators need to design network capacity, in which how many network nodes, including NTN nodes, are required to accommodate the traffic of massive user devices for various services. However, it is difficult to optimize network capacity per device in networks where massive devices are connected and the connected devices continuously change. We introduce device types as groups of devices whose traffic characteristics resemble and optimize network capacity per device type for efficient network capacity design. As a result, it is expected to reduce the calculation amount to optimize network capacity design.

Network operators need to grasp network requirements of connected devices for network capacity design. Network operators may grasp network requirements for network capacity design with a device type classification method by linking requirements models with device types that network

Manuscript received May 9, 2023.
Manuscript revised August 16, 2023.
Manuscript publicized October 17, 2023.

[†]The authors are with Network Service Systems Laboratories, NTT Corporation, Musashino-shi, 180-8585 Japan.

[*]A part of this paper was presented at IEEE ICNC 2023 [1] and IEICE Tech. Rep. [2].

a) E-mail: chikako.takasaki.dp@hco.ntt.co.jp
b) E-mail: tomohiro.koorikawa.xa@hco.ntt.co.jp
c) E-mail: kyota.hattori.vj@hco.ntt.co.jp
d) E-mail: hidenari.oowada.uk@hco.ntt.co.jp

DOI: 10.1587/transcom.2023WWP0004

operators define in advance. For example, a requirement model needs a lot of bandwidth because they continuously send high-resolution video. On the other hand, another requirement model does not need much bandwidth but low latency and high-reliability communication. As a result, a method to classify traffic into device types are required for efficient network capacity design to fulfill network requirements in beyond 5G and 6G eras with massive and diverse connected devices.

Possible approaches for device type classification can be categorized into two classes: analysis based on logs and management databases; and analysis based on traffic in the network. Typical ways in the former approaches are to use system logs (Syslog) [5] and to use subscriber identity module (SIM) information in mobile networks. However, the standardized formats of these data are not so detailed that can be exploited to classify connected device types including IoT device types. In particular for IoT devices, IoT device types are not specified in SIM and depend on users' inputs. In the latter approach, several works have presented traffic analysis-based ways in IoT-only networks, which can be further categorized into two sub-classes: to analyze traffic contents [6], [7]; and to analyze only traffic behavior such as packet headers and statistics on the basis of encrypted traffic [8]–[10]. However, due to privacy-related laws that network operators must follow, network operators need to take a device classification approach without the contents of traffic. Additionally, it is difficult to apply methods using features extracted from the traffic of specific protocols because protocols of traffic sent from IoT are not standardized and proprietary protocols for some specific IoT services are used. There is no method to classify devices including not only IoT devices but also mobile phones, PCs, and routers only with traffic behavior. Therefore, there is a need for a new device type classification method based only on traffic behavior, such as traffic statistics, whose input data can be collected by a network operator.

In this paper, we propose a method to classify device types by analyzing only traffic behavior, including statistics and their time series extracted from encrypted traffic without using any information of payload and specific protocols in two stages, which can be collected by a network operator at base stations and some access points. The proposed method analyzes statistics and their time series extracted from encrypted traffic without information of payload and specific protocols. The proposed method is applicable to traffic in networks where diverse devices are connected. We analyze packet header statistics and their time series in the first and second stages, respectively. The first stage classifies devices into general types such as IoT, non-IoT, and routers by analyzing packet header statistics. The second stage classifies connected devices that are classified as IoT in the first stage into IoT device types by analyzing traffic waveforms that indicate the number of packets and the sum of packet lengths sent by each device over time. We demonstrate that the proposed method classifies device types only on the basis of traffic behavior in a network, where diverse devices including mobile phones, PCs, and routers as well as IoT devices are connected.

Progress is being made in intelligent networking, which aims to autonomously and automatically design and control networks with artificial intelligence (AI) and machine learning (ML). The 3rd Generation Partnership Project (3GPP) [11] specifies that the network data analytics function (NWDAF) is to be introduced to 5G core network (5GC) functions in release 15 and AI/ML function is to be included in NWDAF in releases 16 and 17. NWDAF is assumed to collect and analyze various data in networks with ML models, and output analyzed results. An intelligent RAN is being promoted by O-RAN Alliance [12] which has been established by fundamental network operators all over the world including AT&T, China Mobile, Deutsche Telekom, NTT DOCOMO, and Orange. They define RAN intelligent controller (RIC) as a control function that plays roles to make RAN intelligent. The design and management of these automated autonomous networks require more various, timely, finer-grained data in terms of traffic, service and application types of connected user devices, and surroundings without manual operation. Therefore, the proposed method to classify traffic into device types and the dynamic optimization based on the device types for network capacity design will fit for automated autonomous networks in beyond 5G and 6G eras.

The rest of this paper is organized as follows. Section 2 describes related works. Section 3 presents the proposed method. Section 4 describes evaluation results using a real traffic dataset. Section 5 indicates directions for future research. Finally, Sect. 6 concludes this paper.

2. Related Work

This work focuses on a method to classify source device types by analyzing traffic characteristics. Hence, we classify the related works into two topics: traffic classification and device type classification.

2.1 Traffic Classification

Methods have been presented to classify applications and services for security and network design and management by analyzing encrypted traffic [13]–[16]. S. Rezaei et al. surveyed encrypted traffic classification approaches using deep learning [13]. They provided general guidelines for classification tasks including data collection and cleaning, feature selection, and model selection. They also raised the problem of collecting and labeling a large amount of data for deep learning model training and proposed a multi-task learning approach for traffic classification [14]. N. Bayat et al. provided a method to classify encrypted server name indication in HTTPS traffic [15]. K. Hatanaka et al. proposed a method to extract and predict both patterns of accessed domains and temporal access patterns from DNS query log data for network monitoring [16]. These methods use traffic information in communication with specific protocols such

as HTTPS and DNS. However, these methods are not applicable to networks where various devices, including IoT, are connected because protocols used in communication with IoT are not standardized. The proposed method is suitable for traffic classification in networks where various devices are connected because it classifies traffic without protocol information. In beyond 5G and 6G eras, massive devices for various services are connected to networks, and the connected devices and the service requirements dynamically change because of the introduction of NTN. Network operators are expected to efficiently design network capacity to accommodate traffic sent from massive devices and deal with dynamic changes in the connected devices and requirements. It is difficult to optimize network capacity per application because of frequent optimization caused by dynamic change and handle all devices as the same because of the variety of devices and services. Hence, device type classification helps efficient network capacity design by handling devices of the same device type together.

Several state-of-the-art approaches in the fields of other deep learning tasks such as image recognition and natural language processing are applied to classification tasks for encrypted traffic [17]–[19]. Y. Zhou et al. apply an improved AlexNet model to encrypted traffic classification [17]. AlexNet [20] is the first deep convolutional neural network (CNN) model proposed in 2012 for image recognition. They shorten the long training time of AlexNet and have higher accuracy than the classical traffic classification model. M. Lotfollahi et al. proposed a CNN model that integrates both feature extraction and classification [18]. They also provide an implementation of ResNet one dimensional (1D) model in addition to the CNN model. Residual neural networks (ResNet) [21] is a standard model for image recognition tasks, which has a very deep convolutional network. The ResNet 1D model is implemented to apply the ResNet model to traffic analysis that has one dimension. X. Lin et al. applied bidirectional encoder representations from transformer (BERT), which achieves state-of-the-art performance on natural language processing tasks, to encrypted traffic classification tasks [19]. They showed that BERT outperformed several state-of-the-art methods on multiple datasets. In beyond 5G and 6G eras, classification models are assumed to require retraining many times to accommodate the frequent changes of connected devices and their requirements. In addition, smaller models are expected because it is supposed to implement the models distributed to each access point such as base station. These state-of-the-art methods take long to retrain in the networks though they are effective for classification tasks. Hence, the proposed method is suitable to apply to the networks in beyond 5G and 6G eras because the proposed method is smaller than the state-of-the-art models in terms of lines of code.

2.2 Device Type Classification

Network operators need to classify traffic into device types to efficiently optimize network capacity of user devices. In beyond 5G and 6G eras, various devices such as mobile phones and tablets as well as IoT devices are connected to networks. Therefore, network operators need a method to classify traffic into device types. Table 1 shows the summary of the related works regarding device classification by analyzing traffic on the axes of data types and classification objectives.

Table 1 Summary of related works for device type classification.

		Data types in analysis	
		Contents	Behavior
Classification objectives	Source device	[6]	[8], [9]
	Device type	[7]	[10], Proposed

There are methods to extract device information from packets of the Universal Plug and Play (UPnP), SNMP, and NETCONF protocols. However, several challenges remain in the methods using the IoT management protocols: standardization of protocols and data models, and security and privacy [22]. Therefore, a secure and efficient method is expected to extract device information for heterogeneous IoTs. A method has been studied to identify devices connected to the network by analyzing the communication traffic with the server (device server) operated by the manufacturer [6]. It is not applicable to device type classification because device types include several devices whose manufacturers are different, and their manufacturers produce many types of devices. J. Bao et al. proposed a hybrid supervised and unsupervised learning method to classify seen and unseen devices on the basis of service types [7]. For network capacity design, unknown devices are expected to be classified as types with similar network requirements. These two studies are analyzed traffic including contents. However, network operators generally cannot see packet payloads. H. Noguchi et al. studied a method of device identification that identifies the model of devices by analyzing traffic feature similarities [8]. This method may not identify devices in networks where the connected devices and their traffic continuously change because the rules of the similarity threshold to identify devices as the same device defined in the database change. A. Sivanathan et al. presented a method to identify source devices by analyzing traffic with random forest classifier [9]. The method uses header information of specific proxies such as DNS and NTP. The protocols may not be used in all IoT communication because the standardization of IoT communication is premature. In addition, devices whose traffic features change over time may not be classified into correct types without time series analysis. These studies presented methods to identify source devices by analyzing packet header statistics. However, only the lower-right category in Table 1 is suitable for classifying device types by analyzing only traffic behavior, and these methods are not applicable.

L. Bai et al. studied classification of four IoT device types (cameras, switches & triggers, electronics, and hubs) by analyzing traffic output from IoT devices with deep learning [10], which are categorized as the same category with the proposed method in Table 1. Their study was presented for

device management in networks where many IoT devices are connected, assuming that they can extract traffic sent by IoT devices. However, network operators may not extract traffic sent by IoT devices from traffic including other devices: mobile phones and tablets when they cannot access SIM information. As a result, there is no applicable method to classify device types by analyzing information in networks, where mobile phones and routers as well as IoT devices are connected.

3. Device Type Classification Method

We propose a method to classify device types only on the basis of traffic behavior, including statistics and their time series extracted from encrypted traffic without using any information of payload and specific protocols in networks where various devices include not only IoT but also mobile phones and routers. Figure 1 shows an overview of the proposed method. The proposed method classifies device types on the basis of the two-stage traffic analysis. In the first stage, we analyze packet header statistics and classify devices into three general device types: A. IoT, B. Non-IoT, and C. Router. A. IoT consists of IoT devices such as cameras and hubs, B. Non-IoT consists of devices such as PCs, mobile phones, and C. Router consists of network routers. In the second stage, we analyze traffic waveforms that represent the number of packets and the sum of packet lengths sent by each device over time and classify IoT device types of devices classified as A. IoT in the first stage. The proposed method does not specify device types of any kind.

In beyond 5G and 6G eras, massive devices for various services are connected to networks, and the connected devices dynamically change because of the introduction of NTN. Network operators are expected to efficiently design network capacity to accommodate traffic sent from massive devices and deal with dynamic changes in the connected devices. It is difficult to optimize network capacity per service or device within a short time span because of the heavy calculation amount. Hence, a method is required to classify traffic into device types introduced for efficient network capacity design. The device type classification method is expected to help optimize network capacity with a light calculation amount by defining the device types and linking the device types with the network requirement models by network operators in advance. In this paper, we propose a method to classify traffic into device types by analyzing only traffic behavior toward efficient network capacity design.

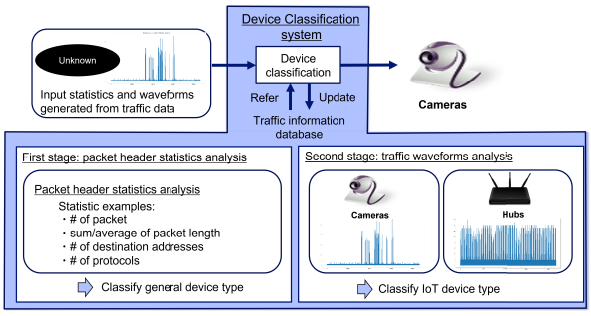


Fig. 1 Overview of the device classification method.

3.1 First Stage: Classification of Device Categories by Analyzing Packet Header Statistics

In the first stage, we analyze packet header statistics and classify the general device types (A. IoT, B. Non-IoT (other than IoT devices: mobile phones, tablets, and PCs), and C. Router). We analyze statistics such as traffic send rate and the number of destination addresses, extracted from packet headers with ML models and classify device categories. We use (1a) logistic regression, (1b) random forest, and (1c) support vector machine (SVM) as ML models.

Logistic regression classifies data by applying a logistic function to a linear polynomial equation. Random Forest classifies data by generating a large number of decision trees with repeated conditional branches in a tree structure and taking majority votes or averages. SVM classifies data by determining class boundaries so that the distance between the nearest data in each class is maximized.

3.2 Second Stage: Classification of IoT Function Categories by Analyzing Traffic Waveforms

In the second stage, traffic waveforms that represent time variation of packet statistics are analyzed with deep learning to classify IoT device types of devices that were classified as A. IoT in the first stage. We construct three types of deep learning models using multi-layer perceptron (MLP), long short-term memory (LSTM), and CNN. A neural network (NN) is a model that mimics human neural circuits. MLP is a fully-connected NN that uses the output of all nodes in each layer to compute nodes in the next layer. Our MLP model is configured with four MLP layers and the softmax layer used for classification. The LSTM is an improvement on the recurrent neural network (RNN), which can learn temporal dependencies, and enables learning of long-term dependencies. The CNN learns by convolving surrounding information and is good at image identification. In this paper, we construct three models: (2a) a MLP model with four layers (MLP), (2b) a LSTM model with two layers of MLP on each time step and one layer of LSTM (LSTM), and (2c) a LSTM and 1D CNN model with one layer of LSTM and one layer of 1D CNN (LSTM+CNN). Figure 2 shows the structure of (2c) LSTM+CNN.

3.3 Target Network Models

Figure 3 shows the target network models of the proposed device type classification method. We assume that the device type classification method is applied to two types of networks: cellular and home networks. Network operators capture raw traffic without decryption outside base stations or access points, and the traffic is input to the device type classifier in both types of networks. The network operators

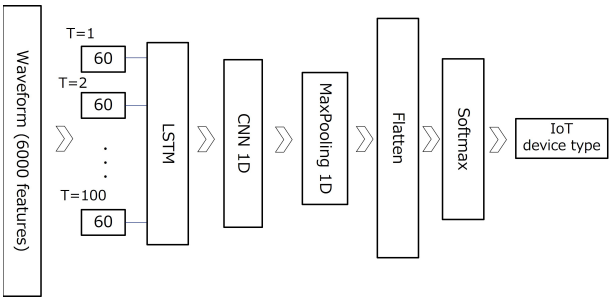


Fig. 2 Structure of (2c) LSTM+CNN model.

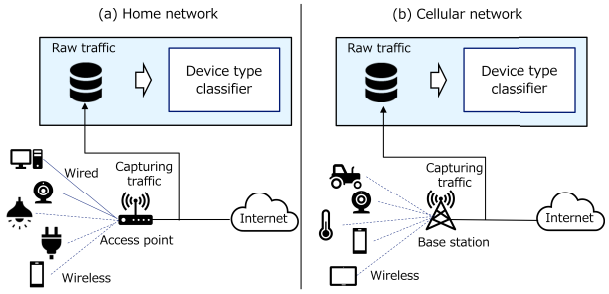


Fig. 3 Target network models.

may capture traffic with other packet capturing equipment in cellular and home networks or access points where packet dump software is installed. For example, tcpdump is installed on OpenWrt [23], which is a Linux operating system targeting embedded devices such as routers, used in the related work providing the dataset [24].

4. Evaluation

We evaluate the device type classification accuracy of the proposed method, including comparison with the state-of-the-art classification models. A. Sivanathan et al. provided a dataset that captures sending and receiving packets from 31 devices in nine categories connected to the network of the University of New South Wales [24]. In addition to the dataset, we used our own dataset consisting of traffic sent and received from 12 devices in seven categories. Table 2 shows the number of devices per function category in the two datasets.

4.1 First Stage: Classification of General Device Types by Analyzing Packet Header Statistics

In the first stage, we analyze packet header statistics with ML models, and classify the three general device types: A. IoT, B. Non-IoT, and C. Router. We generate a feature dataset by extracting the number of packets, the sum of packet lengths, the average of packet lengths, and the number of destination addresses, which each device sends for 10 minutes. Table 3 shows the number of data per general device type: 75% of the data is used for training and the rest for test. In addition, 30% of training data is used for validation data in hyper-parameter optimization with Optuna [25].

Table 2 Number of devices per device category.

Device type		# of device
General device type	IoT device type	
A. IoT	1. Hubs	4
	2. Cameras	14
	3. Switches & Triggers	5
	4. Air quality sensors	2
	5. Healthcare devices	4
	6. Light bulbs	1
	7. Electronics	6
	8. Smart watches	3
B. Non-IoT		3
C. Router		1

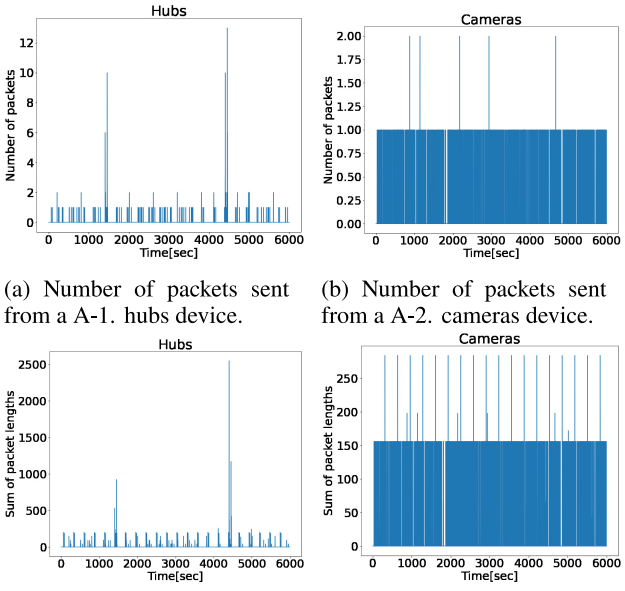
Table 3 Number of data per general device type used for analysis of packet header statistics.

General device type	# of data
A. IoT	95,364
B. Non-IoT	6,801
C. Router	6,592
Total	108,757

Table 4 Accuracies and F-measures of general device type classification per ML model.

ML model	Training accuracy	Test accuracy	F1-score
(1a) Logistic Regression	92.5%	92.5%	60.5%
(1b) Random Forest	97.8%	97.6%	92.4%
(1c) SVM	98.9%	90.7%	58.1%

We use training accuracies, test accuracies, and F1-scores for evaluation metrics of each model. Training and test accuracies indicate the percentage of prediction results that are consistent with the correct answer. An evaluation only based on accuracies may not correctly classify devices of B. Non-IoT and C. Router, which have fewer data than A. IoT, because the number of data in each category is non-uniform. Therefore, we use the F1-score as a metric to accurately evaluate when using non-uniform dataset. The F1-score is a metric of performance with a non-uniform dataset that takes the harmonic mean of the recall (the percentage of data that is predicted to be positive out of those that are actually positive) and the precision (the percentage of data that is actually positive out those that are predicted to be positive). Table 4 shows the training accuracies, the test accuracies, and the F1-scores of each ML model. While (1a) logistic regression and (1c) SVM classify the general device types with test accuracies of more than 90%, the F1-scores drop significantly and fail to classify the devices in the B. Non-IoT and C. Router types. On the other hand, (1b) Random Forest shows a test accuracy of 97.6% and an F1-score of 92.4%, indicating that it classifies the general device types more accurately than other ML models.



(a) Number of packets sent from a A-1. hubs device.

(b) Number of packets sent from a A-2. cameras device.

(c) Sum of packet lengths sent from a A-1. hubs device.

(d) Sum of packet lengths sent from a A-2. cameras device.

Fig. 4 Examples of traffic waveforms.

4.2 Second Stage: Classification of IoT Device Types by Analyzing Traffic Waveforms

In the second stage, we classify the IoT device types of the devices that are classified as A. IoT in the first stage by analyzing traffic waveforms with deep learning. In addition, our proposed deep leaning model is compared with the ResNet 1D model [18] which is a state-of-the-art method for encrypted traffic classification. ResNet [21] is a standard model for image recognition tasks, which has a very deep convolutional network. The ResNet 1D model is implemented to apply the ResNet model to traffic analysis that has one dimension.

Traffic waveforms represent time variations of packet statistics such as the number of packets and the sum of packet lengths. We generate two types of datasets with 6000 features: the number of packets that each device sends per 100 ms for 10 minutes and the sum of packet lengths that each device sends per 100 ms for 10 minutes. Figure 4 shows examples of traffic waveforms of a A-1. hubs device and a A-2. cameras device. The left waveforms, (a) and (c), in Fig. 4 show the number of packets and the sum of packet lengths sent from a hub device, respectively. The right waveforms, (b) and (d), in Fig. 4 show the number of packets and the sum of packet lengths sent from a camera device, respectively. In terms of device types, outlines of waveform and packet sent frequencies are totally different in a hub device ((a) and (c)) and a camera device ((b) and (d)). In terms of types of waveforms, waveforms of a hub device represent similar outlines. On the other hand, waveforms of a camera device represent that the sum of packet lengths is different although the number of packets is the same at a time slot.

Table 5 Number of data per IoT device type.

IoT device type	# of data
A-1. Hubs	19,436
A-2. Cameras	45,491
A-3. Switches & Triggers	20,836
A-4. Air quality sensors	15,984
A-5. Healthcare devices	13,248
A-6. Light bulbs	7,632
A-7. Electronics	17,354
A-8. Smart watches	1,673
Total	141,654

Table 6 Accuracies of IoT device type classification with dataset 1.

Deep learning model	Training accuracy	Test accuracy
(2a) MLP	85.2%	75.1%
(2b) LSTM	80.2%	80.3%
(2c) LSTM+CNN	83.0%	82.6%
Previous method [10]	NA	74.8%

Table 7 Accuracies of IoT device type classification with dataset 2.

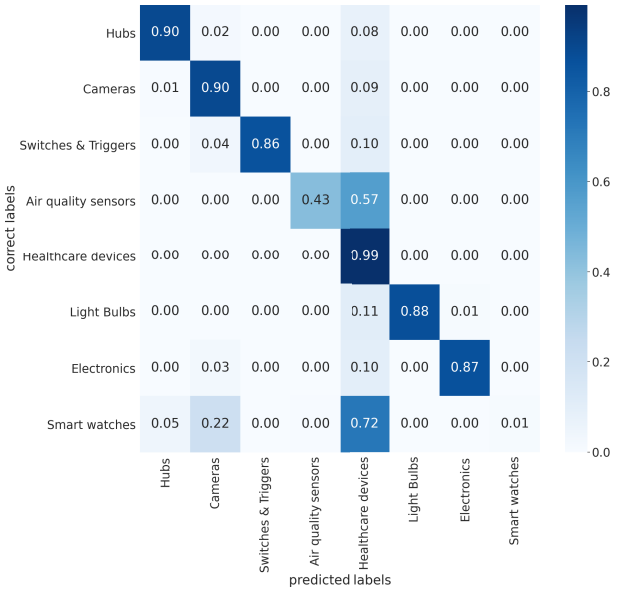
Deep learning model	Training accuracy	Test accuracy
(2a) MLP	75.8%	68.3%
(2b) LSTM	84.6%	84.5%
(2c) LSTM+CNN	90.2%	86.8%
Previous method [10]	NA	74.8%

Table 5 shows the number of data per general device type: 75% of the data is used for training and the rest for test.

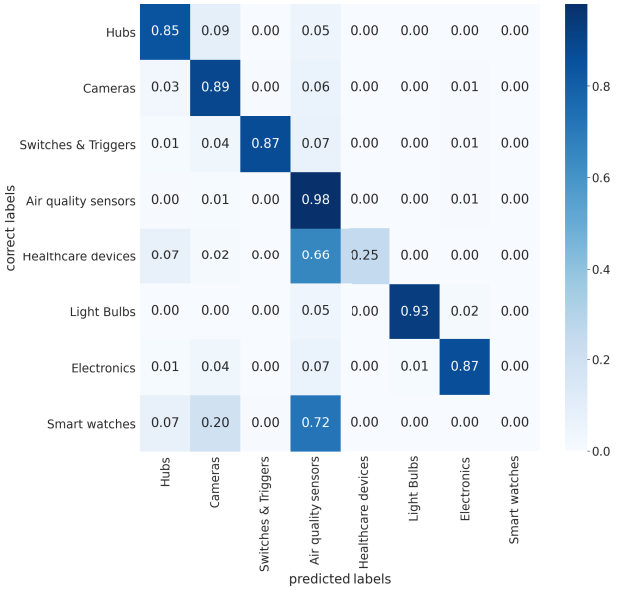
Table 6 shows the training accuracies and the test accuracies of each deep learning model using dataset 1, which is the number of packets that each device sends per 100 ms for 10 minutes. The (2c) LSTM+CNN model has the best accuracy in this evaluation. We consider that this is because the model was able to successfully learn both long-term and short-term time series features by using LSTM and 1D CNN. In addition, Table 6 shows that the proposed method classified device types with about 8% higher accuracy in a network, where not only IoT devices but also mobile phones, PCs, and routers are connected, than the related method [10].

Table 7 shows the training accuracies and the test accuracies of each deep learning model using dataset 2, which is the sum of packet lengths that each device sends per 100 ms for 10 minutes. The (2c) LSTM+CNN model also has the best accuracy in the evaluation. Compared with the results with dataset 1, the results with dataset 2 show higher accuracies to classify IoT device types with the (2b) LSTM and (2c) LSTM+CNN models.

Figure 5 shows the confusion matrices with (2c) LSTM+CNN trained with dataset 1 and 2, which indicates the classification accuracies per IoT device type. The characteristics of classification results are different depending on the input datasets: the mainly misclassified categories using dataset 1 are A-4. Air quality sensors and A-8. Smart watches and the misclassified categories using datset 2 are A-5. Healthcare sensors and A-8. Smart watches. We consider that the misclassifications happen because traffic is sent based on user activity from devices belonging to



(a) Using dataset 1.



(b) Using dataset 2.

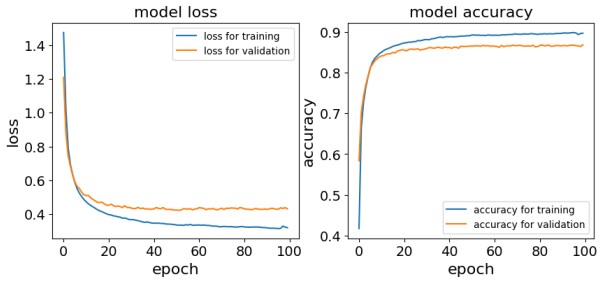
Fig. 5 Confusion matrix of IoT device type classification with (2c) LSTM+CNN model.

A-4. Air quality sensors, A-5. Healthcare sensors, and A-8. Smart watches. The frequency and length of actions are different per user. As a result, it is difficult to extract and train common features of devices of the same type.

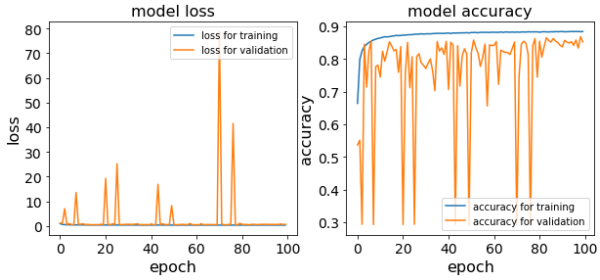
Table 8 shows the training accuracies, the test accuracies, the F1-scores, and the training times per epoch using dataset 2 trained by the proposed (2c) LSTM+CNN model and the ResNet 1D model. Figure 6 shows the learning curves of training and validation loss and accuracy. We used a machine with one CPU [11th Gen Intel(R) Core(TM) i7-11700KF @ 3.60 GHz] and one GPU [NVIDIA GeForce RTX 3090] in the evaluation. The proposed (2c) LSTM+CNN model using dataset 2 that has the best accuracies in the IoT device type classification is compared with ResNet 1D model [18]. We observe that the time consumed for training per epoch with the proposed (2c) LSTM+CNN model is about 40 times shorter than the time with the ResNet 1D model. The proposed (2c) LSTM+CNN model has almost the same accuracy as the ResNet 1D model, which is the state-of-the-art model of traffic classification tasks, even though the (2c) LSTM+CNN model has the simple structure shown in Fig. 2: the proposed (2c) LSTM+CNN model is implemented in 30 times fewer lines than the ResNet 1D model.

Table 8 Accuracy comparison with dataset 2 using (2c) LSTN+CNN and ResNet model.

Model	Training accuracy	Test accuracy	F1-score	Training time per epoch
(2c) LSTM +CNN	90.2%	86.8%	86.6%	3.6 s
ResNet 1D	88.3%	88.0%	87.7%	131.6 s



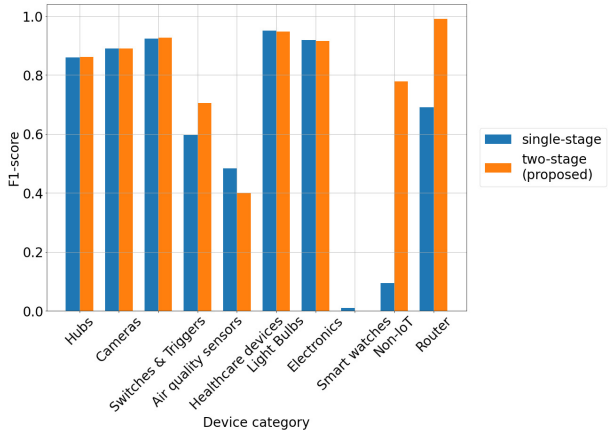
(a) (2c) LSTM+CNN.



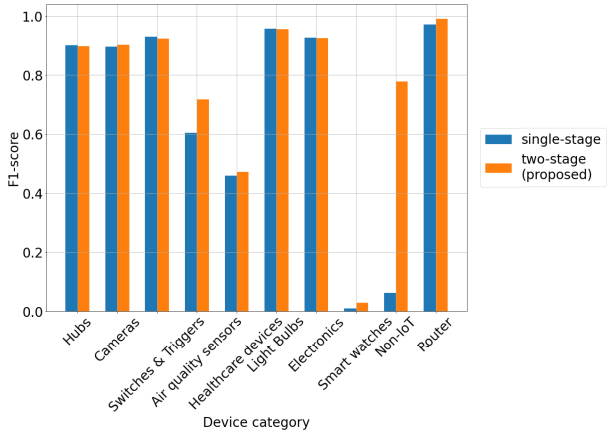
(b) ResNet 1D

Fig. 6 Learning curves on training.

We compare classification results with the proposed two-stage device classification method and a single-stage method to evaluate effects of the proposed method. The single-stage method classifies ten categories including eight IoT device types, Non-IoT, and Router shown in Table 2 by analyzing dataset 2 which represents time variations of the sum of packet length with the (2c) LSTM+CNN and ResNet 1D models only on one stage. Figure 7 indicates F1-scores of each device category classified by the proposed method and the single-stage method. The devices in the Non-IoT category are not classified accurately by the single-stage method. On the other hand, the proposed method classifies the devices in the Non-IoT category more accurately. As a



(a) (2c) LSTM+CNN.



(b) ResNet 1D

Fig. 7 F1-scores of each device category.

result, the two-stage traffic behavior analysis will be more suitable than the single-stage one when diverse devices are connected to the network.

5. Directions for Future Research

There can be three directions for future research in this field. The first direction is to classify device types without packet header information if packets, including headers, are fully encrypted. The proposed device classification method is applicable to encrypted traffic if traffic is divided per source devices. In case it is impossible to divide traffic per source device because of header encryption, a method to classify traffic into device types using only fully encrypted traffic, for example, analysis of the length, inter-arrival time, and byte segment of each packet is possible. J. Zhang et al. proposed an application classification method by training byte segments of encrypted packets with CNN [26].

The second direction is to deal with misclassification of traffic sent based on user activity from devices. The frequency and length of actions are different per device user. It is difficult to extract and train common features of devices of the same type. For example, A-5 healthcare devices generate temporal and burst data traffic triggered by user activities such as measurement of body weight and blood pressure. A method is required to extract common features of control packets, which do not depend on user activity, from traffic, for example, by frequency analysis such as Fourier and Wavelet analysis.

The third direction is to evaluate the performance of service-aware scenarios, such as a scenario considering URLLC services. The evaluation depends on the optimization in network capacity design rather than on the proposed method itself. The target scenarios affect the accuracy, training and inference time, and the number of device types of the device type classification method required to fulfill the network requirement. Hence, the requirements and the performance should be evaluated with the optimization in network capacity design.

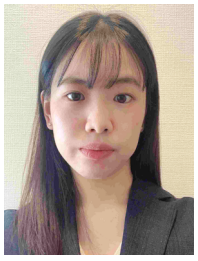
6. Conclusion

We introduced device types as groups of devices whose traffic characteristics resemble for efficient network capacity design in beyond 5G and 6G eras with massive and diverse connected devices. In this paper, we proposed a method to classify device types by two-stage analysis of only traffic behavior collected in networks where various devices are connected. We demonstrated that the proposed method classifies IoT devices by analyzing packet header statistics in the first stage. In the second stage, we also demonstrated that our system classifies more IoT device types by analyzing traffic waveforms using the (2c) LSTM+CNN model with about 12% higher accuracy than the existing method. In addition, the proposed model in the second stage performs almost the same accuracy with 40 times shorter training time as a state-of-the-art model of traffic classification, ResNet 1D model. The proposed method is expected to classify traffic into device types for capacity design in networks where massive devices for various services are connected and the connections continuously change.

References

[1] C. Takasaki, T. Korikawa, K. Hattori, and H. Ohwada, "Traffic behavior-based device type classification," 2023 International Conference on Computing, Networking and Communications (ICNC), pp.353–357, 2023.
[2] C. Takasaki, T. Korikawa, K. Hattori, H. Ohwada, and M. Shimizu, "IoT device identification based on two-stage traffic analysis (encouragement talk)," IEICE Technical Report, NS2022-8, May 2022.
[3] Ericsson, "Ericsson mobility report," Nov. 2022.
[4] P. Popovski, K.F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," IEEE Access, vol.6, pp.55765–55779, 2018.
[5] E. Baseman, S. Blanchard, Z. Li, and S. Fu, "Relational synthesis of text and numeric data for anomaly detection on computing system logs," 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), pp.882–885, 2016.
[6] H. Guo and J. Heidemann, "IP-based IoT device detection," IoT S&P '18, New York, NY, USA, pp.36–42, Association for Computing Machinery, 2018.

[7] J. Bao, B. Hamdaoui, and W.K. Wong, "IoT device type identification using hybrid deep learning approach for increased IoT security," 2020 International Wireless Communications and Mobile Computing (IWCMC), pp.565–570, 2020.
[8] H. Noguchi, M. Kataoka, and Y. Yamato, "Device identification based on communication analysis for the internet of things," IEEE Access, vol.7, pp.52903–52912, 2019.
[9] A. Sivanathan, D. Sherratt, H.H. Gharakheili, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman, "Characterizing and classifying iot traffic in smart cities and campuses," 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp.559–564, 2017.
[10] L. Bai, L. Yao, S.S. Kanhere, X. Wang, and Z. Yang, "Automatic device classification from network traffic streams of internet of things," 2018 IEEE 43rd Conference on Local Computer Networks (LCN), pp.1–9, 2018.
[11] "The 3rd generation partnership project," https://www.3gpp.org/
[12] "O-ran alliance," https://www.o-ran.org/
[13] S. Rezaei and X. Liu, "Deep learning for encrypted traffic classification: An overview," IEEE Commun. Mag., vol.57, no.5, pp.76–81, 2019.
[14] S. Rezaei and X. Liu, "Multitask learning for network traffic classification," 2020 29th International Conference on Computer Communications and Networks (ICCCN), IEEE, Aug. 2020.
[15] N. Bayat, W. Jackson, and D. Liu, "Deep learning for network traffic classification," arXiv, arXiv:2106.12693, 2021.
[16] K. Hatanaka, T. Kimura, Y. Komai, K. Ishibashi, M. Kobayashi, and S. Harada, "Extraction and prediction of user communication behaviors from dns query logs based on nonnegative tensor factorization," IEEE Trans. Netw. Service Manag., vol.20, no.3, pp.2611–2624, 2023.
[17] Y. Zhou and J. Cui, "Research and improvement of encrypted traffic classification based on convolutional neural network," 2020 IEEE 8th International Conference on Computer Science and Network Technology (ICCSNT), pp.150–154, 2020.
[18] M. Lotfollahi, R.S.H. Zade, M.J. Siavoshani, and M. Saberian, "Deep packet: A novel approach for encrypted traffic classification using deep learning," arXiv, arXiv:1709.02656, 2018.
[19] X. Lin, G. Xiong, G. Gou, Z. Li, J. Shi, and J. Yu, "ET-BERT: A contextualized datagram representation with pre-training transformers for encrypted traffic classification," Proc. ACM Web Conference 2022, ACM, April 2022.
[20] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," Proc. 25th International Conference on Neural Information Processing Systems - Volume 1, NIPS'12, Red Hook, NY, USA, p.1097–1105, 2012.
[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," arXiv, arXiv:1512.03385, 2015.
[22] S. Sinche, D. Raposo, N. Armando, A. Rodrigues, F. Boavida, V. Pereira, and J.S. Silva, "A survey of IoT management protocols and frameworks," IEEE Commun. Surveys Tuts., vol.22, no.2, pp.1168–1190, 2020.
[23] "Openwrt," https://openwrt.org/
[24] A. Sivanathan, H.H. Gharakheili, F. Loi, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman, "Classifying IoT devices in smart environments using network traffic characteristics," IEEE Trans. Mobile Comput., vol.18, no.8, pp.1745–1759, 2019.
[25] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," Proc. 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.2623–2631, 2019.
[26] J. Zhang, F. Li, F. Ye, and H. Wu, "Autonomous unknown-application filtering and labeling for DL-based traffic classifier update," IEEE INFOCOM 2020 - IEEE Conference on Computer Communications, pp.397–405, 2020.



Chikako Takasaki received B.S. and M.S. degrees in informatics from Ochanomizu University, Tokyo, Japan, in 2019 and 2021. In 2021, she joined Nippon Telegraph and Telephone Corporation (NTT), Tokyo. She is currently with NTT Network Service Systems Laboratories. Her research interests include traffic analysis, network architecture, network digital twins, and machine learning for networks.

Tomohiro Korikawa received the B.S. and M.S. degrees in physics from Waseda University, Tokyo, Japan, in 2012 and 2014, respectively, and the Ph.D. degree in informatics from Kyoto University, Kyoto, Japan, in 2021. In 2014, he joined Nippon Telegraph and Telephone Corporation (NTT), Tokyo. He is with Network Service Systems Laboratories, where he is currently a Researcher. His research interests include network architecture, network design, and network digital twin.

Kyota Hattori is a Senior Research Engineer of Network Service Systems Laboratories in Nippon Telegraph and Telephone Corporation (NTT), Tokyo, Japan. He received the B.E. in applied physics and the M.E. in computational science and engineering from Nagoya University and a Ph.D. degree in information science and technology from Hokkaido University in 2004, 2006, and 2019, respectively. In 2006, he joined NTT Network Service Systems Laboratories, where he has been engaging to research traffic flow control and optical network architecture. He received the Award for SC4 Best Paper from Photonics in Switching 2012 (PS2012) and the Young Researcher's Award from IEICE in 2013.

Hidenari Ohwada is a Senior Research Engineer of Network Service Systems Laboratories in Nippon Telegraph and Telephone Corporation (NTT), Tokyo, Japan. He received a M.E. from Tohoku University, Japan, in 2002.