Sample Length Determination from Network Traffic based on Periodicity Analysis

Chikako Takasaki, Tomohiro Korikawa, and Kyota Hattori
NTT Network Service Systems Laboratories, Tokyo, Japan
{chikako.takasaki, tomohiro.korikawa, kyota.hattori}@ntt.com

***Abstract*—Machine learning technologies will enable more intelligent network operations in the beyond 5G/6G era. Operators will provide appropriate network operations that meet service requirements. Network traffic is one of the most familiar information that represents user behavior such as calling, streaming videos, and exchanging text messages. The traffic is a time series of the packet captured at a certain collection point, whose inter-arrival times are unevenly spaced in general. The uneven inter-arrival times of packets are one of the useful information representing traffic behavior. However, it is not feasible to collect a large amount of traffic in terms of operation and storage equipment. Therefore, a method to extract traffic features representing user behavior from limited amount of traffic, is necessary. This paper proposes a method to determine sample length, the length of features per sample representing traffic characteristics, based on the periodicity analysis of unevenly-spaced packet series. The periodicity analysis is inapplicable to the unevenly-spaced packet series. Therefore, we introduce a function to convert unevenly-spaced packet series into evenly-spaced time series to enable the periodicity analysis. The proposed method determines the sample lengths, input to machine learning models, based on periodicity analysis of the converted evenly-spaced packet series. We evaluate the accuracy in traffic classification task to compare the proposed method with the existing method. The evaluation results show that the classification accuracy trained with the samples extracted from a part of input traffic is comparable to the accuracy trained with the samples extracted from all input traffic, whereas the accuracy of the existing method decreases as the number of days of input traffic decreases. When training with the samples extracted from the all input traffic, each traffic type is classified with 99% accuracy. The proposed method will reduce the duration of traffic collection for maintaining the accuracy in the traffic classification.**

***Index Terms*—Traffic feature extraction, periodicity analysis, machine learning, deep learning.**

I. Introduction

Machine learning technologies will enable more intelligent network operations in the beyond 5G/6G era. Operators can provide network connectivity considering user behaviors based on machine learning models trained with information collected from networks. For example, in 5G networks, open radio access networks (O-RAN) [1] introduce an architecture to control the networks using machine learning and policies on near-real-time and non-real-time radio access network (RAN) intelligent controller (RIC). In O-RAN architecture, operators collect and store information about communication performance per user and network congestion through base stations. RIC analyze the information using machine learning models and optimize resource allocation such as radio resource and slices to fulfill service requirements per user. Operators require to collect information representing user and network behaviors for appropriate network operations.

For network operators, network traffic is one of the most familiar information that represents user behaviors such as calling, video streaming, and exchanging text messages. The traffic is a time series of packets captured at a certain collection point, such as base stations and gateway routers, whose inter-arrival times are unevenly spaced in general. The uneven inter-arrival times of packets are one of the useful information representing traffic behavior. For example, we can observe the bursty behaviors and the sparsity behaviors from the uneven packet inter-arrival times. The observations of the burstiness and the sparseness help operators to grasp timely behaviors of users. However, it is impossible to collect and store a large amount of raw traffic, which flows in real networks, in terms of operation and storage equipment. Therefore, an efficient feature extraction method, which extracts features representing user behavior from limited amount of traffic, is necessary.

A typical method to prepare the input data of machine learning models is sampling the traffic data by calculating averages or sums of packets sent during a time window with a certain length. This method empirically determines a common length of series, input into machine learning models, in all samples, regardless of traffic characteristics. A method to determine the lengths of series per sample includes periodicity and frequency analysis in signal processing. This method can determine the length of series per sample considering characteristics of the original series; however, the periodicity and frequency analysis methods are inapplicable to the unevenly-spaced packet series. Therefore, efficient feature extraction from traffic data calls for a different approach to adaptively determine the sample length that can be applied to unevenly-spaced time series.

We propose a method to determine the lengths of series per sample representing traffic characteristics using the periodicity analysis. This paper defines the sample length as the length of series per sample input to machine learning models. The periodicity analysis is inapplicable to the unevenly-spaced packet series. Hence, we present a function that converts unevenly-spaced packet series into evenly-spaced time series to apply the periodicity analysis. The function calculates evenly-spaced time series using a ratio of the inter-arrival time between a packet and the previous packet to the inter-arrival time between the packet and the next packet. We determine the sample lengths, which is input to machine learning models, as

979-8-3315-2096-0/25/$31.00 ©2025 IEEE

the peak value of autocorrelation for the converted evenly-spaced time series. The proposed method is expected to shorten the duration of traffic observation required to maintain the accuracy of machine learning models. We evaluate the relation between the classification accuracy and the duration of traffic collection in the traffic classification task. We train a traffic classification model with the traffic features extracted by the proposed method. The proposed method determines the sample lengths input to the classification model considering the uneven inter-arrival times of packets. The evaluation results show that the classification accuracy trained with the samples extracted from a part of traffic is comparable to the accuracy trained with the samples extracted from the whole traffic. When training with the samples extracted from the whole traffic, each traffic type is classified with 99% accuracy. The experiments show that the proposed method efficiently extracts features even from traffic collected for short duration in the traffic classification.

II. Related Work

We summarize methods to extract features from traffic for analysis using machine learning. H. Zhang et al. studied a traffic classification method using graph neural networks (GNN) by describing a packet as a graph and each raw byte as a node of the graph [2]. L. Yu et al. presented a network intrusion detection method using convolutional neural networks (CNN) by describing a packet with 256-dimensional one-hot encoding for each byte [3]. These methods do not take account of the relation among packets because raw bytes and header information of each packet are analyzed. A. Sivanathan et al. presented a method to identify source devices by analyzing traffic statistics with a random forest classifier [4]. J. Bao et al. studied a hybrid supervised and unsupervised learning method using 297 statistic features extracted from traffic to classify seen and unseen devices [5]. These methods analyze statistics of packets sent for a period, and the methods do not consider the temporal change in the statistics. L. Bai et al. studied a method to classify traffic by analyzing time series of traffic statistics calculated from packets sent for each window with CNN and long short-term memory (LSTM) [6]. I. Akbari et al. studied a traffic classification method using a CNN and LSTM model learned with the combination of the three types of features [7]. These approaches empirically determine a common sample length in all samples based on the fine-tuning of models or the experience regardless of traffic characteristics. The proposed method extracts effective features by determining the sample lengths adaptively to input traffic characteristics considering the uneven inter-arrival times of packets.

Periodicity analysis and frequency analysis are typically used to determine the sample lengths input to machine learning models from time series. Typical methods include Fourier analysis and Wavelet analysis, which are only applicable to evenly-spaced time series. Applying the periodicity and frequency analysis methods to traffic requires converting the unevenly-spaced packet series into evenly-spaced time series. One method for converting unevenly-spaced packet series into evenly-spaced time series is window aggregation. Window aggregation calculates statistics of packets arriving for a window, such as the number of packets and the sum and average of packet lengths. E. Grabs et al. analyzed evenly-spaced time series generated by window aggregation with Fourier analysis for the classification of video resolution and frame rate [8]. T. Zhou et al. studied a traffic feature encoding model using discrete Fourier transform to map the encrypted voice traffic features to the frequency domain space for encrypted voice traffic fingerprinting [9]. It is difficult to determine the sample lengths in methods that generate evenly-spaced time series using window aggregation. J. Koumar et al. presented a method to classify traffic by training periodicity features extracted using the Lomb-Scargle periodogram [10]. The Lomb-Scargle periodogram is a method analyzing periodicity for unevenly-spaced time series, which is used to analyze time series of discrete events such as astronomical events. The method does not consider the temporal change in network traffic behavior because the periodicity feature during a period are analyzed. The proposed method determines the sample lengths based on periodicity analysis of evenly-spaced time series. We address inter-arrival times of packets by introducing a function converting the unevenly-spaced packet series into evenly-spaced time series, which has not been considered in the existing methods that use statistics and periodicity analysis methods for unevenly-spaced time series.

III. Periodicity Analysis Method for Unevenly-spaced Packet Series

We propose a method to determine the sample lengths considering the traffic characteristics. The proposed method introduces a conversion function to enable the periodicity analysis for determining the sample lengths, which is inapplicable to the raw traffic. The proposed function converts the unevenly-spaced packet series into hypothetical evenly spaced packet series by regarding the packet series as a continuous flow of data. Each value of the evenly spaced time series is calculated based on the lengths and the inter-arrival times of a packet, the previous packet, and the next packet. We apply autocorrelation, one of the periodicity analysis methods, to the converted evenly-spaced time series, assuming that the traffic is periodic. The sample length is determined as the peak value of the autocorrelation of the converted evenly spaced time series. This section presents the proposed method in terms of the sample length determination and a function to convert the unevenly-spaced packet series into the evenly spaced time series.

A. Proposed sample lengths determination

This section describes the overview of the proposed method to determine the sample lengths based on periodicity analysis. The input traffic, unevenly-spaced packet series, is represented by a set of packets

$$\{p_0, p_1, \cdots, p_{i-1}, p_i, p_{i+1}, \cdots, p_T\},$$

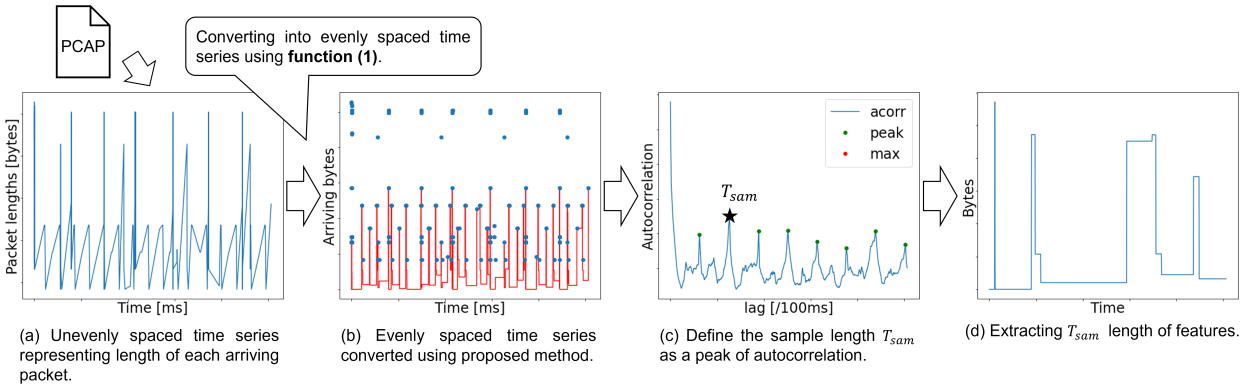

Fig. 1. Overview of the proposed method to determine the sample lengths.

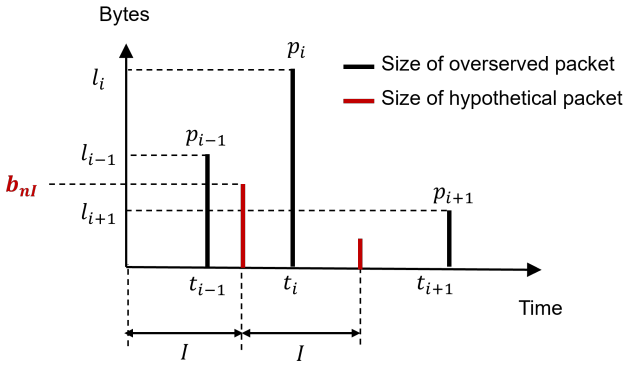

Fig. 2. Example of the observed packets and the hypothetical packet.

where p_i and T are a observed packet at time t_i and traffic collection duration, respectively. The size of packet p_i is denoted by l_i. The proposed method to determine the sample length are outlined as follows. First, the unevenly-spaced packet series are converted into hypothetical evenly-spaced packet series by regarding the packet series as a continuous flow of data. The converting function is described in Section III-B. Next, the sample length are determined as the peak values of the autocorrelation for the converted evenly-spaced packet series. Figure 1 shows the overview of the proposed method to determine the sample length. Figure 1(a) shows an example of a packet series, which is the unevenly-spaced time series. Each plot represents one packet, whose length and arrival time are shown in the vertical axis and the horizontal axis, respectively. The proposed method converts the unevenly-spaced packet series (Figure 1(a)) into the evenly-spaced time series (Figure 1(b)) following the proposed function. The sample length T_{sam} is determined as the peak value of autocorrelation of the evenly-spaced time series (Figure 1(c)). In Figure 1(c)), the lags represent the number of intervals shifted to calculate the autocorrelation. The input features into machine learning models are extracts based on the determined sample lengths (Figure 1(d)).

The proposed method calculates the n-th value of evenly-spaced time series, b_{nI}, focusing on the last packet p_i of arriving packets by the time nI following the proposed function. I is the interval of the evenly-spaced time series, determined in manual. We define the size and the arrival time of the packet p_i as l_i and t_i, respectively. The proposed function calculates a value with the sizes of the packet p_i and the next packet p_{i+1}, l_i and l_{i+1}, and the inter-arrival times among the previous packet p_{i-1}, the packet p_i, and the next packet p_{i+1}, $t_i - t_{i-1}$ and $t_{i+1} - t_i$. Figure 2 shows an example of the observed packets and the hypothetical packets.

B. Proposed function converting unevenly-spaced packet series into evenly-spaced time series

This section introduces the function converting unevenly-spaced packet series into evenly-spaced time series. The evenly-spaced time series are calculated by assuming that the same amount of data arrives continuously during the time between the arrival of a packet and the arrival of the next packet. The evenly-spaced time series represent the hypothetical packets capturing the continuous arriving data at even interval.

We formulate the proposed function converting the unevenly-spaced packet series into the evenly-spaced time series as follows:

$$b_{nI} = \min\left(l_i, l_{i+1}, \frac{l_i + l_{i+1}}{2}\frac{t_i - t_{t-1}}{t_{i+1} - t_i}\right). \quad (1)$$

In the function, I represents the interval of evenly-spaced time series, and b_{nI} is the n-th value of the time series. l_i and t_i represent the size and the arrival time of the packet p_i, respectively. The packet p_i is the last packet in $\{p_i|(n-1)I < t_i \leq nI\}$. We assume that the mean of l_i and l_{i+1} of signals arrives during the time $t_{i+1} - t_i$. We multiply the ratio of the packet inter-arrival times $(t_i - t_{i-1})/(t_{i+1} - t_i)$, assuming that the arriving signals is in proportion to the inter-arrival times t_{i-1}, t_i, t_{i+1}. In addition, the minimal value is calculated considering that the calculated value become bigger than the sizes of the observed packets when the packet arrival time $t_i - t_{i-1}$ is longer than $t_{i+1} - t_i$.

TABLE I
NUMBERS OF DEVICES AND SAMPLES EXTRACTED BY THE PROPOSED METHOD PER TRAFFIC TYPE.

Traffic type	# of devices	# of samples
Type 1	2	278,600
Type 2	7	392,295
Type 3	4	81,724
Type 4	2	19,859
Type 5	3	36,868
Type 6	1	22,067
Type 7	3	88,083
Total	22	919,496

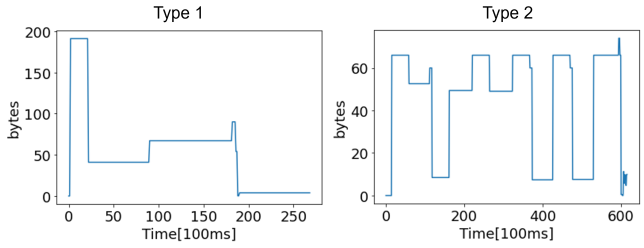

Fig. 3. Examples of samples extracted by the proposed method.

IV. EVALUATION

We evaluate the proposed method to determine the sample lengths in a traffic classification task. We measure the accuracy in traffic classification of seven types using an open IoT traffic dataset. In addition, we evaluate the classification accuracy when training samples extracted from different days of input traffic to compare the proposed method with the existing method [11]. The existing method extracts statistical time series by calculating the sums of packet lengths send per 100 ms for ten minutes. We select the existing method, which differs from the proposed method in a point using a common sample length in all samples.

A. Dataset and classification model

We use a dataset presented in the work [12], which consists of packet captures of 22 devices in seven IoT categories for 60 days. Table I shows the numbers of devices and the number of samples extracted by the proposed method per type. Each sample extracted by the proposed method consists of the converted evenly-spaced time series divided into the sample length. The all samples extracted by the proposed method has the 3,000 length, which is the maximum value in the sample lengths determined with the proposed method. We set the interval of time series I to 100 ms. We make a sample by repeating the series of the sample length T_{sam} into the 3,000 length because all samples need to have the same length, which is input into the machine learning model. On the other hand, each sample extracted with the existing method has the same 6,000 length consisting of the sums of packet lengths send per 100 ms for ten minutes. The examples of samples extracted by the proposed method and the examples of samples extracted by the existing method are shown in Figure 3 and Figure 4,

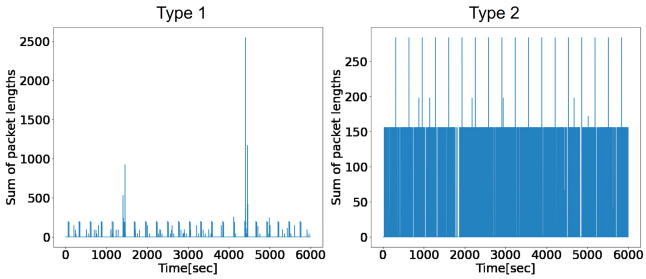

Fig. 4. Examples of samples extracted by the existing method.

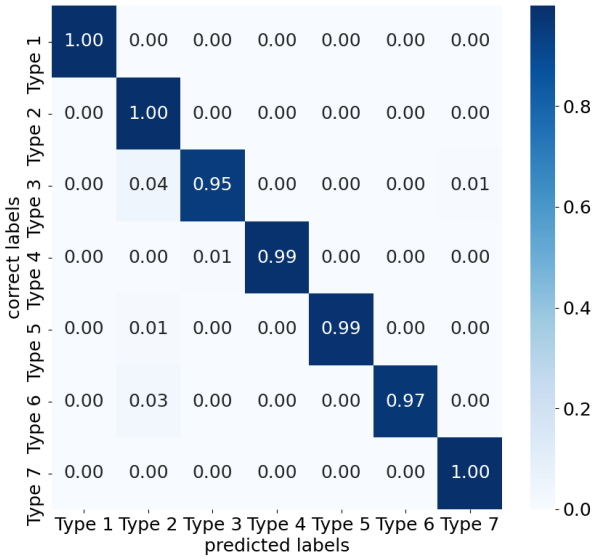

Fig. 5. Confusion matrix of traffic classification trained with samples extracted by the proposed method.

respectively. The examples of the proposed method and the examples of the existing method are included in the same set of types, Type A and Type B. The example of Type A represents a sparsity behavior because a few data points of large amount are temporally captured at the beginning of the sample. In contrast, the example of Type B represents a bursty behavior because multiple data points of similar amount are continuously captured. The series of the values shown in the vertical axis are input into the machine learning model. We use an LSTM and 1D-CNN model consisting of one layer of LSTM, one layer of 1D-CNN, and one layer of MaxPooling as the classification model.

B. Evaluation results

First, we evaluate the proposed method in terms of the accuracy using samples consisting of training samples extracted from 45 days of packet captures and test samples extracted from 15 days of packet captures. The numbers of training samples and test samples are 700,442 and 219,054, respectively. Figure 5 shows the confusion matrix. The results show that the test accuracy of all test samples is 99% and the test accuracy of each traffic type is more than 95%. The loss

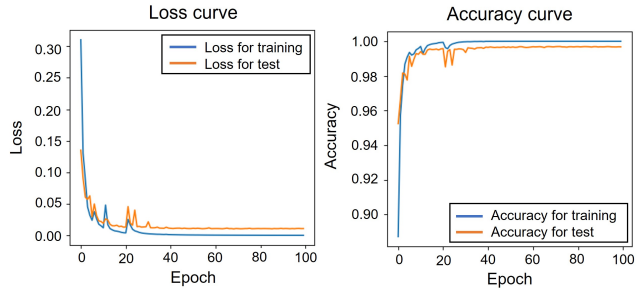

Fig. 6. Loss and accuracy curves of traffic type classification trained with samples extracted by the proposed method.

TABLE II
CLASSIFICATION ACCURACY OF THE PROPOSED METHOD PER TRAFFIC DATA DAYS USED FOR TRAINING.

# of days	Proposed method		Existing method	
	# of samples	Test accuracy	#of samples	Test accuracy
5 days	107,104	96.2%	9,530	86.3%
10 days	211,776	97.8%	15,449	91.3%
15 days	317,568	98.3%	24,001	92.7%
30 days	640,032	99.1%	50,195	97.1%
45 days	1,019,977	99.2%	76,062	97.6%

and accuracy curves converge at about 30 epochs as shown in Figure 6.

Second, we compare the test accuracy of the proposed method with the test accuracy of the existing method when training the traffic features extracted from packet captures for 5, 10, 15, 30, and 45 days. Table II shows the accuracy and the number of training samples extracted by the the proposed method and the accuracy and the number of training samples extracted by the existing method for different days of input traffic. The results show that the proposed method outperforms the existing method, where the proposed method maintains more than 95% accuracy when training the samples extracted from packet captures for five days. On the other hand, the accuracy of the existing method decreases as the number of days of input traffic decreases, where it is 10% below the proposed method when training the samples extracted from packet captures for five days. In addition, the number of samples of the proposed method is more than ten times as many as the number of samples of the statistical method.

The evaluation results shows that the proposed method is effective to reduce the duration of traffic collection for maintaining the accuracy in the traffic classification. In addition, the proposed method increases the samples extracted from the packet capture comparing to the existing method. This is caused by determining the sample lengths representing traffic characteristics with the proposed method. Note that the samples extracted with the proposed method will become less than the samples extracted with the statistical method when the sample lengths are not determined due to the low frequency of captured packets.

V. CONCLUSION

This paper proposed a method to determine the sample lengths considering traffic characteristics based on the periodicity analysis. The periodicity analysis is inapplicable to unevenly-spaced packet series. Therefore, the proposed method converts unevenly-spaced packet series into hypothetical evenly-spaced packet series by regarding the packet series as a continuous flow of data. The sample length is determined as the peak of autocorrelation, one of the periodicity analysis methods, for the converted evenly-spaced time series. We evaluated the accuracy when training samples extracted from different days of input traffic to compare the proposed method with the existing method in traffic classification. The evaluation results show that the accuracy when training samples extracted from a part of input traffic with the proposed method is comparable to the accuracy when training samples extracted from all input traffic. The proposed method is effective to reduce the duration of traffic collection for maintaining the accuracy in the traffic classification. A method is left for future work to improve the function converting the unevenly-spaced packet series into the evenly-spaced time series.

REFERENCES

[1] O-ran alliance. https://www.o-ran.org/.
[2] H. Zhang, L. Yu, X. Xiao, Q. Li, F. Mercaldo, X. Luo, and Q. Liu. Tfe-gnn: A temporal fusion encoder using graph neural networks for fine-grained encrypted traffic classification. In *Proc. of the ACM Web Conf. 2023*, WWW '23, page 2066–2075, New York, NY, USA, 2023. Assoc. for Comput. Machinery.
[3] L. Yu, J. Dong, L. Chen, M. Li, B. Xu, Z. Li, L. Qiao, L. Liu, B. Zhao, and C. Zhang. Pbcnn: Packet bytes-based convolutional neural network for network intrusion detection. *Comput. Networks*, 194:108117, 2021.
[4] 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. In *2017 IEEE Conf. on Comput. Commun. Workshops (INFOCOM WKSHPS)*, pages 559–564, 2017.
[5] J. Bao, B. Hamdaoui, and W. Wong. Iot device type identification using hybrid deep learning approach for increased iot security. In *2020 Int. Wireless Commun. and Mobile Comput. (IWCMC)*, pages 565–570, 2020.
[6] L. Bai, L. Yao, S. S. Kanhere, X. Wang, and Z. Yang. Automatic device classification from network traffic streams of internet of things. In *2018 IEEE 43rd Conf. on Local Comput. Netw. (LCN)*, pages 1–9, 2018.
[7] I. Akbari, M. A. Salahuddin, L. Ven, N. Limam, R. Boutaba, B. Mathieu, S. Moteau, and S. Tuffin. A look behind the curtain: Traffic classification in an increasingly encrypted web. *Proc. ACM Meas. Anal. Comput. Syst.*, 5(1), feb 2021.
[8] E. Grabs, T. Chen, E. Petersons, D. Efrosinin, A. Ipatovs, J. Kluga, and D. Culkovs. Features extraction for live streaming video classification with deep and convolutional neural networks. In *2021 IEEE Microwave Theory and Techn. in Wireless Commun. (MTTW)*, pages 58–63, 2021.
[9] T. Zhou, Y. Zeng, Y. Chen, Z. Liu, and J. Ma. Encrypted voice traffic fingerprinting: An adaptive network traffic feature encoding model. In *ICC 2023 - IEEE Int. Conf. on Commun.*, pages 3768–3773, 2023.
[10] J. Koumar and T. Čejka. Network traffic classification based on periodic behavior detection. In *2022 18th Int. Conf. on Netw. and Service Manage. (CNSM)*, pages 359–363, 2022.
[11] C. Takasaki, T. Korikawa, K. Hattori, and H. Ohwada. Device type classification based on two-stage traffic behavior analysis. *IEICE Trans. on Commun.*, E107.B(1):117–125, 2024.
[12] 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. on Mobile Comput.*, 18(8):1745–1759, 2019.