Network Traffic Attention Features Fusion for Unauthorized IoT Devices Detection

Duong Van Dan, Ta Minh Thanh
Le Quy Don Technical University, 236 Hoang Quoc Viet, Ha Noi, Viet Nam

Correspondence: Ta Minh Thanh, thanhtm@lqdtu.edu.vn
Communication: received 25 Apr 2023, revised 17 May 2023, accepted 29 June 2023
DOI: 10.32913/mic-ict-research.v2024.n1.1205

***Abstract*: The demand for using IoT devices is increasing because of its usefulness. IoT devices are present in all areas of life. They make life easier and more comfortable. Moreover, many companies and households are tending to increase in the use of IoT devices to turn on/off home, school, office into smart home, smart school, smart office and so on. Although IoT devices are very useful, they have major security problems because they are not interested, leading to a large number of security holes and creating opportunities for hackers to attack the systems. In this paper, we develop an attention features fusion (AFF) method for the purpose of classifying and detecting IoT devices that are not in the pre-registered white list. Our recommended technique finds the features with the best classifier ability, then converts them into images and uses ensemble learning based on the baseline models which are deep learning models to detect and classify. Our proposed solution helps to detect strange devices that are exchanging data in the IoT network, reducing the risk of the system being attacked by hackers. According to the experimental results, our method achieves very good results when we detect ten IoT devices with up to 97.85% accuracy.**

Keywords: *Internet of Things (IoT), deep learning, IoT device identification, attention features fusion method (AFF), IoT device detection, feature fusion; image processing.*

I. INTRODUCTION

1. Overview

In the world, more and more IoT systems are deployed. IoT devices are increasingly useful and almost everyone knows about it. It includes devices connected to the internet system such as computers, cameras, sensors, and many other smart devices. The number of IoT devices worldwide by the end of 2022 is about 14 billion devices and this number will increase even more. It is expected to reach threshold of 30 billion devices by the end of 2030 [1]. IoT devices are used in many different fields, with about 60% of devices connected by 2020 and this number is increasing [1]. According to Cisco's prediction [2], this number may increase to 29 billion as soon as 2023.

However, apart from the strong development of IoT systems, the field of information security for IoT networks has received little attention. Leading to the existence of a large number of security holes in the system, creating opportunities for hackers to attack the system and steal important resources. The security vulnerabilities of IoT devices come from the manufacturer's lack of information security knowledge as well as the lack of national common safety standards for IoT devices [3].

In 2016, a renowned attack on the IoT systems called Mirai Botnet [4]. In earlier times, IoT systems as well as IoT devices were set up and deployed with almost default accounts and passwords, leading to systems being quickly attacked by hackers if they had a set of default accounts and passwords of companies. In addition, when IoT devices are connected to the Internet, hackers can easily use search engines like Shodan [5] to find the physical location of the devices and other information of the devices. The top 10 critical security vulnerabilities of IoT devices published by OWASP [6] in 2018, including very basic vulnerabilities but which are weak points of IoT devices such as weak passwords, lack of authentication mechanisms, hierarchical, lack of coding element, and so on.

Collecting and managing data from all devices in a large organization is also a challenge. If there is an error in the system, it is likely that the devices connected to the IoT network will be disconnected and need time to fix. At present, there are no international standards for IoT devices, when participating in data exchange in the Internet, it is easy for bad actors to take advantage of weaknesses to attack these devices.

In order to ensure that the devices in the IoT network are the devices of the organization, have been registered for use in advance and prevent the strange devices of hackers from entering the IoT network, the first step we need to do is to identify/detect which are the organization's IoT devices and which are potentially bad IoT devices. Therefore, we need

to build solutions to classify IoT devices that are exchanging data in the IoT network.

2. Our contributions

In this paper, we propose a method to detect IoT devices that are exchanging data in the network by using network traffic to analyze and detect those IoT devices based on the our Attention Features Fusion method (AFF). The suggested approach achieves wholly high results when we apply it to the test dataset, obtaining up to 97.85% accuracy comparing with another conventional algorithms.

Our contributions of this paper are explained as follows:

- In fact, the dataset collected about the network traffic of IoT devices has many data attributes. This causes noise when training deep learning models, leading to low accuracy. Therefore, our first contribution is to find the 14 best features for each device and generated an image dataset from 14 best features.
- Propose an attention features fusion (AFF) method by using 14 features with the best classification ability then convert them into images and use Deep Learning models for training. Finally, we use ensemble learning with voting type with deep learning models.
- Discover a collection of models that provide the most accurate classification results for each device.

3. Roadmap

In the remainder of this paper, we organize the content as follows: In Section II, we present the relevant knowledge and some methods proposed by previous researchers. Our suggested approach is presented in Section III. The experimental environment and experimental results are explained in Section IV. Section V concludes this paper and shows the future research works.

II. Related Work

Many researchers have done on unauthorized IoT devices detection before and have obtained extremely positive results. The field of information security on IoT networks is increasingly interested and developed.

The research problems of identifying IoT devices also have different specialized ways. In the paper [7], the author has synthesized and analyzed how to use machine learning in detecting and classifying IoT devices in a very detailed and relatively complete manner. In the paper [7], more specialized, the authors have divided the problem into small aspects such as device-specific pattern recognition, deep learning-enabled device identification, unsupervised device identification, and abnormal device detection. Some

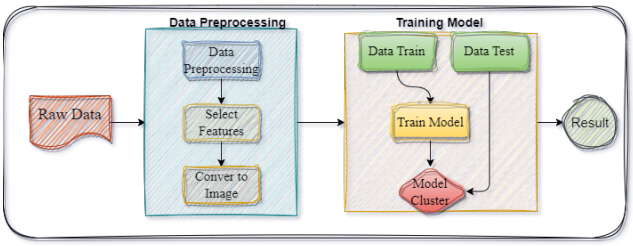

Figure 1. Pipeline for Solving Problem

others, they can use GAN [8] algorithms to extract features, then they use Machine Learning algorithms like Decision Tree, Support Vector Machine (SVM) classification [9, 10]. The results they obtained were relatively good, reaching 95% [11] with the test data set.

In another approach, Jmila *et al*. [12] provides an overview of the use of machine learning-based network traffic analysis for the classification of smart home IoT devices. The paper discusses the challenges associated with smart home IoT device classification and presents various machine learning techniques that have been used for this purpose, including decision trees, support vector machines, and deep learning. The authors also discuss the different types of network traffic that can be used for classification, such as packet header information, packet payload, and flow-based information. The paper concludes with a discussion of future research directions in the area of smart home IoT device classification using machine learning-based network traffic analysis.

Some other studies in transforming data into images and using powerful models of deep learning give better results. This type is widely used in both medical and digital signal processing. Bashivan *et al*. [13] performed the experiment on the data set of electroencephalograms (EEG). They perform transformations to convert the signal to an image form and use Deep Learning for classification. In the article titled:"From ECG signals to images: a transformation based approach for deep learning" [14], the authors have converted the electrocardiogram data to binary data and used classical deep learning networks for classification such as VGG16 [15], AlexNet [16].

With the results of previous researchers, we also expect IoT data after converting into images and using deep learning models to give better results for IoT devices classification.

III. Attention Features Fusion Method (AFF)

In this section, we focus on finding important attributes and converting them to images for unauthorized IoT devices

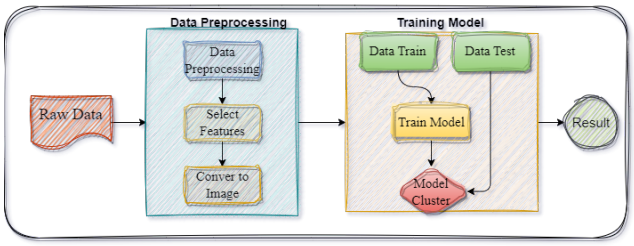

Figure 2. Pipeline for Solving Problem

detection. We collect data in pcap form (pcap file) from IoT devices then pre-process the data to convert the data into csv file format with each record being a data exchange session of each device, including data fields such as ack, ack_A,ack_B, bytes, bytes_ A, bytes_A_B_ratio, bytes_B, ds_field_A, ds_field_B, duration, and many more data fields.

For other methods, researchers use the approach of selecting highly categorical attributes, then feeding into machine learning algorithms or simple Deep Learning models for classification. In this paper, we use all the attributes of the dataset because each attribute has its own classification and will all be effective when we perform the classification based on several machine learning methods.

Our problem-solving approach is to find the best categorical attributes, then use those attributes to convert to images and use deep learning models for classification.

The method we use to find the best attributes through finding the most important attributes for machine learning algorithms like XGBoost, Random Forest, LightGBM [17]. By surveying of the best feature for each algorithm, we combine them to extract the best features for classification.

IV. Attention Features Fusion Method (AFF)

In this section, we focus on finding important attributes and converting them to images for unauthorized IoT devices detection. We collect data in pcap form (pcap file) from IoT devices then pre-process the data to convert the data into csv file format with each record being a data exchange session of each device, including data fields such as ack, ack_A,ack_B, bytes, bytes_ A, bytes_A_B_ratio, bytes_B, ds_field_A, ds_field_B, duration, and many more data fields.

For other methods, researchers use the approach of selecting highly categorical attributes, then feeding into machine learning algorithms or simple Deep Learning models for classification. In this paper, we use all the attributes of the dataset because each attribute has its own classification and will all be effective when we perform the classification based on several machine learning methods.

Our problem-solving approach is to find the best categorical attributes, then use those attributes to convert to images and use deep learning models for classification.

The method we use to find the best attributes through finding the most important attributes for machine learning algorithms like XGBoost, Random Forest, LightGBM [17]. By surveying of the best feature for each algorithm, we combine them to extract the best features for classification.

1. Approach

Well-known algorithms like Random Forest, XGBoost are all based on the classic famous algorithm, Decision Tree as the foundation. To calculate which attribute has the most influence on the final decision, we just need to look at the tree and the weight values in that tree, the larger the value and the closer the node is to the root, the better the attribute, important, the more categorical.

In this paper, we use Scikit-learn[1] library which is a powerful support library for Machine Learning (ML) algorithms. This library supports the Random Forest algorithm and can display the important properties of this algorithm. Similar to the algorithms XGBoost and LightGBM [17] have open source libraries supported, we use *xgboost* and *lightgbm* libraries, then, find the best categorical importance attributes for each algorithm.

To find those important features we used a function in the libraries called feature_importances_. This function is created by the library based on the mathematical formula of the authors Trevor Hastie, Robert Tibshirani, and Jerome Friedman. They presented it in the book **The Elements of Statistical Learning: Data Mining, Inference, and Prediction**. They define the calculation of feature importance using the following equation:

$$I_l^2(T) = \Sigma_{t=1}^{J-1} i_t^2 \mathbb{I}[v(t) = l]$$

In there, T is the whole decision tree, l feature in question, J number of internal nodes in the decision tree, i squared the reduction in the metric used for splitting, II indicator function, v(t) a feature used in splitting of the node t used in splitting of the node. This math formula will sum up all the decreases in the metric for all the features across the tree (Random Forest, Xgboost, LightGBM algorithms are all based on Decision Tree algorithm).

We take the overlap within the 50 best attributes of all 3 algorithms to find the 14 features best classifiers.

[1]https://scikit-learn.org/stable/

Converting those 14 attributes to the image data from 0 to 255, we get the new form data. Such generated data can be fed into classifier algorithms to recognize the unauthorized IoT devices.

We use that image data to train basic deep learning models such as Multilayer Perceptron (MLP), LeNet [18, 19], AlexNet [20], to find the model that best fits the dataset and gives the most accurate prediction results. Then, the traffic will go through a list of models, and return it belongs to which device in the network. If not it belongs to the device outside, then system and we need to issue a warning to the administrator.

2. Data Pre-processing

The dataset that we have is a relatively large dataset, and there are many data fields (features) in it, leading to the training process of many data fields that will cause noise, making the efficiency of the obtained model not high.

Firstly, in order to use this dataset, we need to clean the data, because in the data collection process, it is likely that there will be many missing data columns, we need to remove those missing data. Therefore, we use marking it in this dataset as "?" and during data pre-processing we convert "?" then to "zero" to facilitate data processing, and the model can learn from that data.

As described above, we select the best attributes that each algorithm considers important and highly categorical, convert to images, and use deep learning models for classification.

We proceed to create sub-datasets to train each device by taking all the data of that device and assigning it to "1" at random and randomly assigning the data in the remaining devices to "0" so that the number of data of the device and not of the device are equal.

Data after each train dataset for each device is created, we divide the dataset into train dataset and the dataset to test the model is generated at the rate of 80 to 20, $i.e$, 80% data for train model and 20% data for model testing.

To better understand the whole process we analyze and approach the problem, we can see Figure 2, which describes our steps and solution, from data preprocessing, to data transformation, model training and testing with test dataset.

3. Data Transformation

The first task after we have selected the important attributes with the data set is that we convert the data into images. Some of the attributes we found include ttl_min, B_is_system_port, ttl_A_sum, ttl_max, ssl_dom_server_name_alexaRank and some other attributes, are suitable for image-based classification.

After selecting the important attributes by taking overlap within the 50 most important attributes of machine learning algorithms such as XGBoost, Random Forest, LightGBM. We concatenate the selected features as new features, then convert it to between 0 and 255.

Our purpose is to use AI models that can run similar to the Mnist dataset [21], we try to convert the selected data to the same form as the Mnist dataset which is $28 \times 28 \times 1$. We have tested on two methods. Firstly, we take those 14 important features and padding add zero to the dataset so that there are 784 columns. However, we see that in terms of images, the images representing the data exchange sessions of different devices do not have a certain difference, so we made the next test. In this next test, we do not add zero padding to the dataset, but instead, we padding the dataset itself, ie. 14 existing features, into the dataset until 784 columns are reached. After converting it to an image, the difference between sessions of different devices can be seen.

As seen on Figure 3, we can easily see the difference between the IoT devices that are exchanging data in the network. The results show that the devices produce relatively different images, and the device sessions are relatively similar. Because the sessions of the same device have negligible difference, we cannot see the difference of those sessions with the naked eye. With the above results, we expect the AI models to be able to distinguish well between devices, which is the basis for detecting unauthorized devices in the IoT network.

4. Model Architecture

After converting the data into images, we use well-known deep learning models to train and predict the classification of IoT devices. The models that we use are Multilayer Perceptron(MLP) [22], LeNet, AlexNet, these are models with very simple and primitive architecture of deep learning models.

We use Ensemble Learning [23, 24] with voting type. After converting all 14 features into images, we feed them to the training with a Voting model. The data is put through training the MLP, LeNet, AlexNet models. After training, we get the prediction models with different accuracy. Based on our experience, we weight the MLP, LeNet, and AlexNet models at 1, 2, and 2 respectively.

When a session of a certain IoT device passes, the data is passed through the prediction model. The model will then give that session data to each submodels one by one, and the submodels will predict different values. Based on the predicted value of each model, the voting model multiplies the weighted values and averages them all. The label that

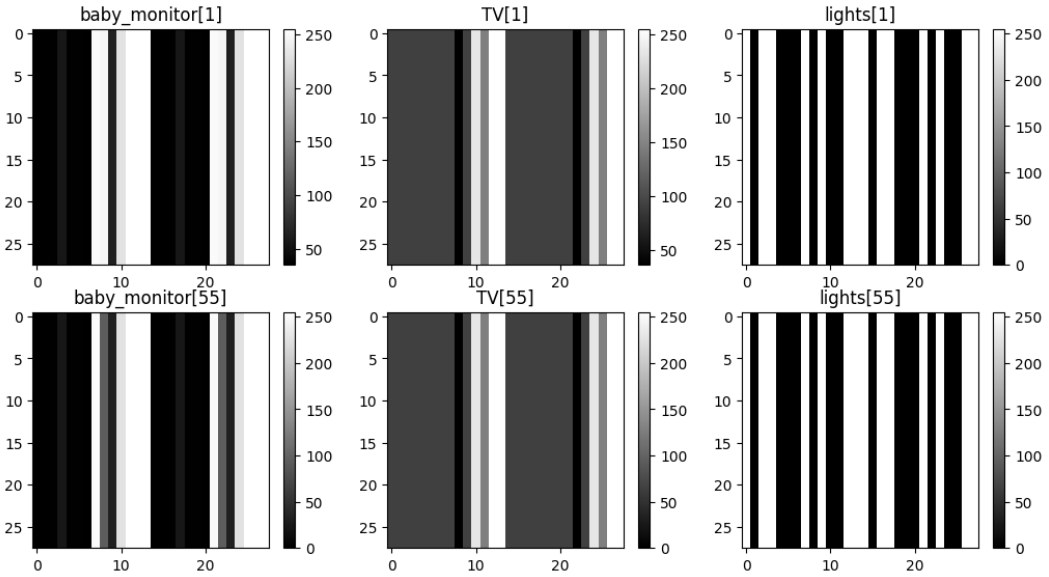

Figure 3. Several Sessions of the IoT Devices after Converting to Images

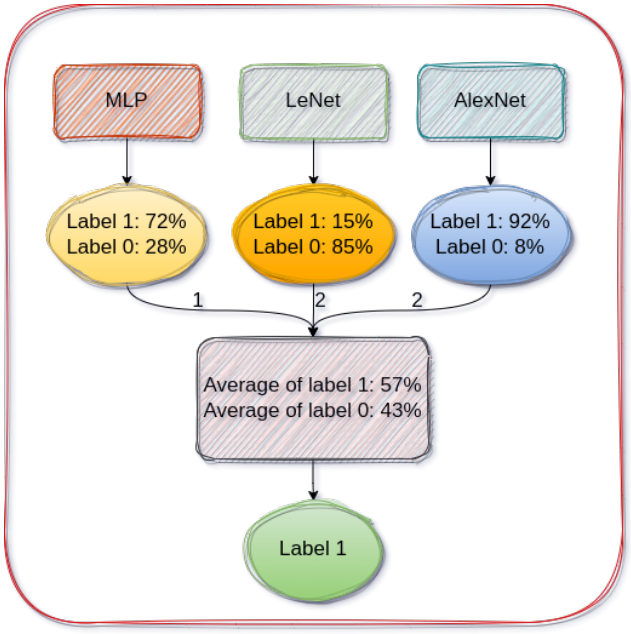

Figure 4. Predictive Voting Model

gives the highest prediction result will be taken as the output of the Voting model. We describe the operation and prediction of the voting model in Figure 4.

V. Experimental Results

1. Experimental environment and Parameter setting

Our experimental environment is on a PC with the following configuration: Intel i7-10700f CPU, 32 GB RAM, and 16GB VRAM RTX A4000 VGA card. We choose python language and jupyter notebook platform to perform the experiments. When we train the model, we use python version 3.6.13, scikit learn 0.24.2, tensorflow 2.6.2, numpy 1.19.2, pandas 1.1.5 and some other tools to analyze and train the model in the experiment.

In this paper, our proposed solution uses basic deep learning models such as MLP, LeNet, and AlexNet. These models have been around for a long time and are relatively suitable for the data set we have. Based on our experience in training each submodel, we found that LeNet and AlexNet models have relatively better accuracy than MLP models. The accuracy of the MLP is about 15% to 20% lower than the rest of the models.So we have set the parameters for model voting with weights for submodels(MLP, LeNet, AlexNet) of 1, 2, 2 respectively.

After we apply that weight to the voting model we get the final model which gives better accuracy in most devices with prediction accuracy increasing from 3% to 12%.

2. Datasets

The dataset we use consists of ten IoT devices such as: security camera, TV, smoke detector, thermostat, water sensor, watch, baby monitor, motion sensor, lights, socket, which are common devices commonly used to build smart homes. Our dataset consists of 298 columns and 399544 rows. This is a relatively large data set and has many different attributes, in which there are also many data attributes that do not have much classification effect.

TABLE I
RESULT USING ML WITH ORIGINAL DATASET

Device \ Algor	DT	RF	XGB	LoRC	LGBM
baby monitor	**97.70**	96.65	94.95	49.21	91.88
lights	92.45	94.80	**97.52**	50.21	93.40
motion sensor	93.75	91.90	89.95	49.58	**96.95**
security camera	94.65	94.91	**97.95**	49.75	90.52
smoke detector	**98.52**	96.01	93.42	89.75	94.53
socket	91.15	92.57	**93.76**	51.09	91.01
thermostat	91.08	91.68	92.52	50.85	**97.56**
TV	**97.65**	95.83	94.86	49.96	93.85
watch	94.62	97.66	**98.64**	50.88	96.64
water sensor	82.99	86.64	**86.86**	51.33	86.61

During operation, the system has collected session data of IoT devices, but this raw data set is difficult for models to learn and classify between devices. The reason is that the excessive amount of noise exists in the data set, exists in attributes that have no categorical value, leading to wrong learning models and inaccurate predictions.

To solve that problem, we propose the AFF method to select the features with the best classifier ability, convert it into an image and pass it through deep learning models for prediction.

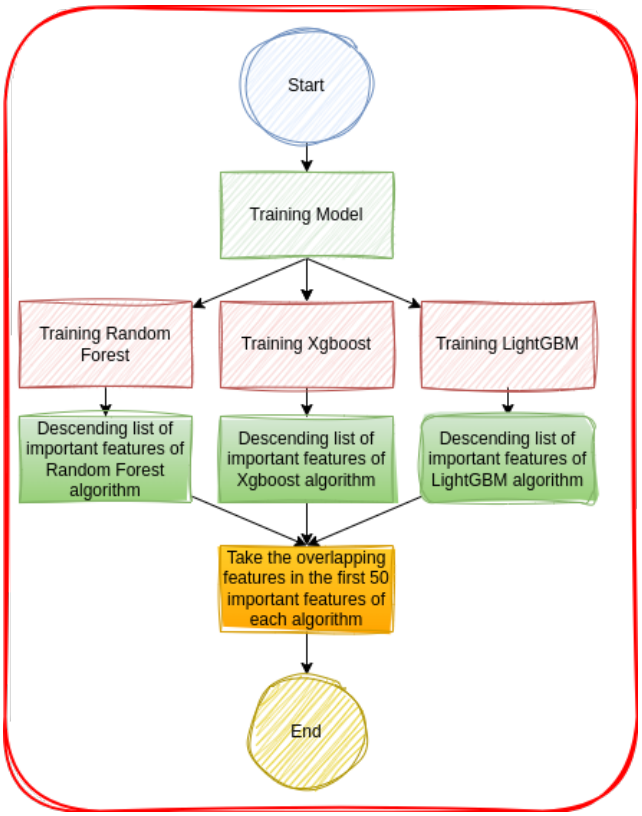

Figure 5. The process to take important features

3. Results of Each Device Classification with The Original Dataset

In this part of the experiment, we tried training the original dataset with machine learning algorithms like Decision Tree (DT), Random Forest (RF), XGBoost (XGB), Logistic (LoRC), LightGBM (LGBM). The obtained results are the basis for us to perform the steps of selecting features and comparing with our method later.

For this method, we simply put all the features into the algorithm and expect that the model will be able to filter out the noise and give the best results. Based on Table I we see, Logistic Regression algorithm gives the lowest accuracy with 49.21% in baby monitor devices, and Xgboost algorithm has the highest accuracy in many devices with 98.64% in watch devices. The table above shows that using the original features is also a relatively good problem-solving approach, but it is not the optimal solution.

4. Results of Each Device Classification with AFF

In this part of the experiment, we find the features with the best classifier, then do experiment with basic Machine Learning algorithms. Then, we continue the experiment by converting the 14 features with the best classifier into an image and using Deep Learning models to classify the generated image.

Finally, we experiment with the voting model belonging to ensemble learning. With the weights of basic deep learning models such as MLP [22], LeNet[2], AlexNet[3] are 1, 2, 2, respectively.

a) The best features of the classification model

After training the machine learning model with all the features, we get the features that are the most important for each algorithm. These features have the best classifier ability with each of those algorithms.

There are many ways to find the most important features of each algorithm. This can include methods such as based on mean decrease in impurity or feature permutation. In this article, we used based on mean decrease in impurity to perform the experiment. This function is used in scikit-learn, xgboost, lightgbm libraries under the function name feature_importances_. It will calculate and rate the importance of each feature to the algorithm. Based on that we find the most important features for each algorithm.

[2]https://paperswithcode.com/method/lenet
[3]https://paperswithcode.com/method/alexnet

TABLE II
THE MOST IMPORTANT FEATURES FOR EACH ALGORITHM

Feature \ Score	XGB	LGBM	RD
ttl_min	**0.57899**	**147**	**0.09298**
B_is_system_port	0.01973	61	0.06732
ttl_A_sum	0.00751	82	0.00327
ttl_max	0.00673	32	0.00554
ssl_dom_server _name_alexaRank	0.00415	78	0.00215
ttl_thirdQ	0.00210	73	0.00728
bytes_A_B_ratio	0.00201	87	0.03479
ttl_B_stdev	0.00070	62	0.00825
reset	0.00065	42	0.03416
ttl_firstQ	0.00064	49	0.00728
ttl_stdev	0.00056	94	0.06376
packet_inter_arrivel_B_min	0.00055	74	0.00914
ttl_B_firstQ	0.00049	30	0.00727
ttl_B_avg	0.00048	39	0.00772

Within the 50 most important features for each of those algorithms, we take overlap and get the important features with all 3 algorithms: Random Forest, XGBoost, and LightGBM. We obtained the 14 most important features with all 3 algorithms within the 50 most important features of the algorithms.

Table II gives us the weight scores of the most important attributes for each algorithm. In all 3 algorithms ttl_min(time to live min) feature is always the most important feature.

b) Results using Machine Learning with 14 selected features

We use classic machine learning algorithms such as Decision Tree (DT), Random Forest (RF), XGBoost (XGB), Logistic (LoRC), LightGBM (LGBM) [17]. These algorithms give relatively good results for the individual training of each device.

After finding the 14 most important features, we proceed to select those 14 features out of a total of 297 features of the original dataset to obtain a new dataset. In this part of training each device, once we have training data for each device, we divide it according to the ratio 80 and 20 to generate training data and testing data to calculate the quality of the model after being trained for each device.

As we can see in Table III, after using the most important features and retraining with machine learning models, significantly better accuracy was obtained in some algorithms for some devices.

c) Results Using Attention Features Fusion Method (AFF)

We use basic deep learning models such as MLP [22], LeNet, AlexNet and Attention Features Fusion Method (AFF) proposed by us to solve the problem. This is the next step after we have selected the best attributes to classify IoT devices. We convert those 14 features into grayscale images with range from 0 to 255, then, train with MLP, LeNet, AlexNet models.

TABLE III
RESULT USING ML WITH 14 SELECTED FEATURES

Device \ Algor	DT	RF	XGB	LoRC	LGBM
baby monitor	95.70	95.65	**97.95**	50.21	96.88
lights	93.46	96.80	**97.80**	49.51	97.40
motion sensor	91.75	95.90	**96.95**	47.58	**96.95**
security camera	95.65	96.81	**97.95**	49.75	97.52
smoke detector	93.52	95.01	95.42	44.75	**96.53**
socket	93.25	**95.87**	95.76	50.99	95.01
thermostat	93.38	95.38	96.52	51.35	**96.56**
TV	93.65	**96.83**	95.86	49.96	94.85
watch	97.62	**97.66**	98.64	50.88	96.64
water sensor	83.07	85.40	86.62	47.97	**87.11**

TABLE IV
RESULT USING DL WITH IMAGE DATASET

Device \ Model	MLP	LeNet	AlexNet	AFF
baby monitor	98.70	99.03	99.15	**99.19**
lights	65.89	86.80	87.80	**88.09**
motion sensor	96.75	98.90	98.95	**99.01**
security camera	97.65	98.15	99.05	**99.22**
smoke detector	89.52	94.01	93.42	**97.32**
socket	66.25	**87.87**	81.76	83.51
thermostat	95.38	98.38	**99.02**	99.12
TV	94.15	96.30	**97.62**	97.53
watch	90.62	95.60	97.64	**98.02**
water sensor	81.09	65.64	91.86	**95.06**

For the AFF method, we conduct training with the voting model belonging to the ensemble learning. We use the basic models are MLP, LeNet, AlexNet and the weights are 1, 2, 2 respectively.

Based on our experimental results in Table IV. We find a set of models that have the best classification for each type of device. Based on the algorithms and methods that we have tested, we can easily choose that model set.

Looking at the chart (Figure 6), we can see that the AlexNet model trained by the image converted from 14 features gives very good results with the baby monitor.

Table IV shows that, our proposed method gives relatively good results and gives better accuracy than using all the features of the dataset. Models like MLP [22], LeNet, AlexNet are also very suitable for classifying the images we provide.

After performing the experiments, we found that some models give the highest accuracy with each device as shown

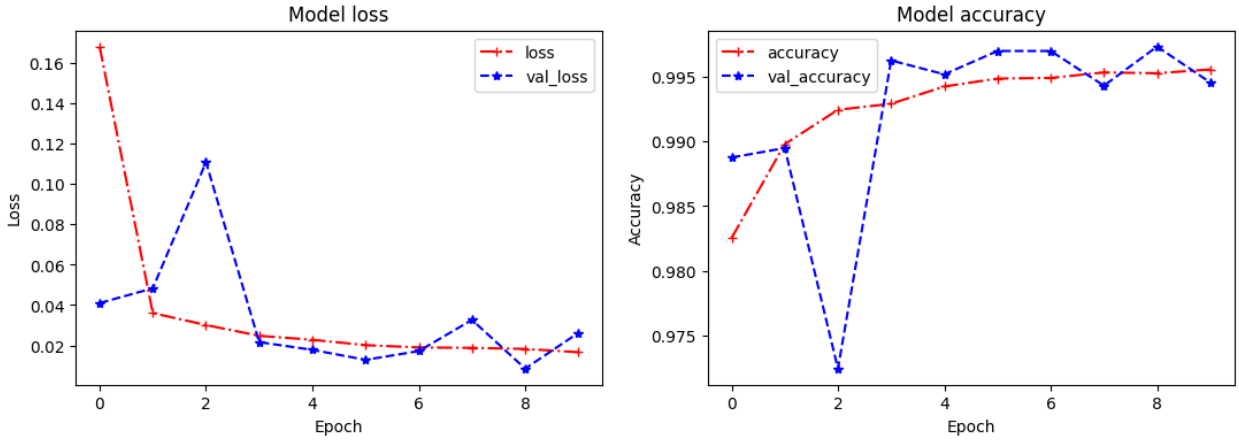

Figure 6. Baby Monitor Device Accuracy and Loss

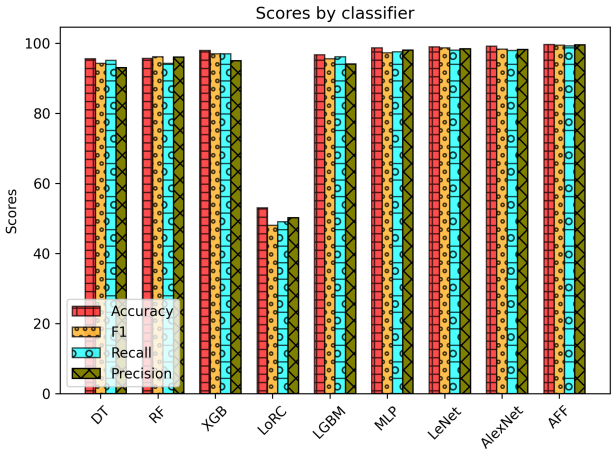

Figure 7. Scores of baby monitor's Classification Models

in Table V.

TABLE V
BEST MODELS FOR EACH DEVICE

Device	Model	Accuracy
baby monitor	AFF	99.19
lights	AFF	88.09
motion sensor	AFF	99.01
security camera	AFF	99.22
smoke detector	AFF	97.32
socket	LeNet	87.87
thermostat	AlexNet	99.02
TV	AlexNet	97.62
watch	AFF	98.02
water sensor	AFF	95.06

From here, we get the model set that gives the best prediction results for each device as listed in the table above.

Based on the Table V we can see that the AFF model runs the best results in most devices, and runs relatively well (but not the best) in a few remaining devices.

The graph of Accuracy, F1, Recall, Precision (Figure 7) shows that the ML algorithms and the DL model both give relatively good results when running with 14 important features selected, except for the Logistic algorithm. Model AFF proved to be superior to other DL models when giving Accuracy, F1, Recall, Precision results a bit higher and more balanced than other models.

5. IoT Device Detection System Results

When we get the models that give the best classification results, we use it to test with the original test dataset. We pass each record through the models in turn. The models will check and return the results that the record belongs to that device or not. If yes, end the test process, otherwise the data will be pushed to the next model in the system for testing. When we have checked all the trained models in the system, but still can not find any device in the system, we conclude that this is a strange device that is not in the pre-registered whitelist of the system, then alert the system administrator.

Performing that test method, we obtained a very high accuracy for the system to detect strange IoT devices accessing the system, the detection results were up to 97.85%. We tested it with a separate dataset that was not seen in any individual device training, and the results were very encouraging as the system reached a good alert level for the system. This can help system administrators better detect strange IoT devices of objects that attack the IoT system.

Our unregistered device detection system works according to Figure 8: Conduct a test of the network traffic of the devices, we will put that network traffic through

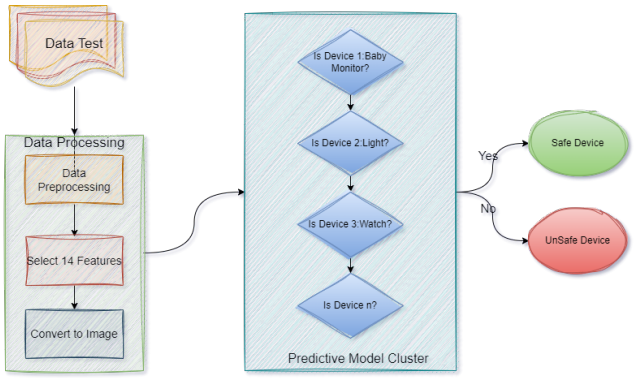

Figure 8. Predictive Model of The System

each device classification model, if it is an organization's device and has been registered, it will return a safe device result. On the contrary, if you have gone through all the classification models and still do not recognize the device of the system, return the device as unsafe and alert the system administrator.

VI. Conclusion

We have proposed and proven that it is possible to build a system to classify IoT devices based on the use of network traffic, and propose a plan to be able to classify that IoT device based on the application of technology deep learning technique.

Our recommended technique (AFF) achieves very good results when conducting tests on our test augmented dataset. The obtained results are the basis for practical application of IoT device classification methods.

In the coming time, we will conduct in-depth research into ways to extract information from network traffic data, such as classifying and detecting the operating systems of IoT devices that are exchanging data in the network system.

References

[1] L. S. Vailshery, "Number of internet of things (iot) connected devices worldwide from 2019 to 2021, with forecasts from 2022 to 2030," Available at https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/ (2022/27/12).

[2] Cisco, "Cisco annual internet report (2018–2023)," Available at https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf (2022/12/06).

[3] T. Robinson, "Interpol warns iot devices at risk," Available at https://www.scmagazine.com/news/architecture/interpol-warns-iot-devices-at-risk (2022/27/12).

[4] J. Margolis, T. T. Oh, S. Jadhav, Y. H. Kim, and J. N. Kim, "An in-depth analysis of the mirai botnet," in *2017 International Conference on Software Security and Assurance (ICSSA)*, 2017, pp. 6–12.

[5] "Shodan," Available at https://www.shodan.io/ (2022/27/12).

[6] "Owasp iot top 10 2018," Available at https://owasp.org/www-pdf-archive/OWASP-IoT-Top-10-2018-final.pdf (2022/27/12).

[7] Y. Liu, J. Wang, J. Li, S. Niu, and H. Song, "Machine learning for the detection and identification of internet of things devices: A survey," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 298–320, 2022.

[8] S. Sivanandam and S. Deepa, *Genetic Algorithms*. Springer Berlin Heidelberg, 2008, pp. 15–37.

[9] F. Nie, W. Zhu, and X. Li, "Decision tree svm: An extension of linear svm for non-linear classification," *Neurocomputing*, vol. 401, pp. 153–159, 2020.

[10] M. Arun Kumar and M. Gopal, "A hybrid svm based decision tree," *Pattern Recognition*, vol. 43, no. 12, pp. 3977–3987, 2010.

[11] A. Aksoy and M. H. Gunes, "Automated iot device identification using network traffic," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.

[12] H. Jmila, G. Blanc, M. R. Shahid, and M. Lazrag, "A survey of smart home iot device classification using machine learning-based network traffic analysis," *IEEE Access*, vol. 10, pp. 97 117–97 141, 2022.

[13] P. Bashivan, I. Rish, M. Yeasin, and N. Codella, "Learning representations from eeg with deep recurrent-convolutional neural networks," 11 2015.

[14] M. Naz, J. H. Shah, M. A. Khan, M. Sharif, M. Raza, and R. Damaševičius, "From ECG signals to images: a transformation based approach for deep learning," *PeerJ Comput Sci*, vol. 7, p. e386, Feb. 2021.

[15] H. Qassim, A. Verma, and D. Feinzimer, "Compressed residual-vgg16 cnn model for big data places image recognition," in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, 2018, pp. 169–175.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, vol. 25. Curran Associates, Inc., 2012.

[17] S. Ray, "A quick review of machine learning algorithms," in *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, 2019, pp. 35–39.

[18] L. Bottou, C. Cortes, J. Denker, H. Drucker, I. Guyon, L. Jackel, Y. LeCun, U. Muller, E. Sackinger, P. Simard, and V. Vapnik, "Comparison of classifier methods: a case study in handwritten digit recognition," in *Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. 3 - Conference C: Signal Processing (Cat. No.94CH3440-5)*, vol. 2, 1994, pp. 77–82 vol.2.

[19] T. Li, D. Jin, C. Du, X. Cao, H. Chen, J. Yan, N. Chen, Z. Chen, Z. Feng, and S. Liu, "The image-based analysis and classification of urine sediments using a lenet-5 neural network," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 8, no. 1, pp. 109–114, 2020.

[20] S. Lu, Z. Lu, and Y.-D. Zhang, "Pathological brain detection based on alexnet and transfer learning," *Journal of Computational Science*, vol. 30, pp. 41–47, 2019.

[21] A. Baldominos, Y. Saez, and P. Isasi, "A survey of handwritten character recognition with mnist and emnist," *Applied Sciences*, vol. 9, no. 15, 2019.

[22] I. García-Magariño, R. Muttukrishnan, and J. Lloret, "Human-centric ai for trustworthy iot systems with explainable multilayer perceptrons," *IEEE Access*, vol. 7, pp. 125 562–125 574, 2019.

[23] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on

ensemble learning," *Frontiers of Computer Science*, vol. 14, no. 2, pp. 241–258, Apr 2020.

[24] O. Sagi and L. Rokach, "Ensemble learning: A survey," *WIREs Data Mining and Knowledge Discovery*, vol. 8, no. 4, p. e1249, 2018.

Duong Van Dan received a Bachelor's degree from Le Quy Don technical university, Hanoi, Vietnam, in 2023. His research interests lie in the area of computer science, network security.
Email: duongvandan2806@gmail.com

Ta Minh Thanh is currently an associate professor and vice dean of Institute of information technology and Communication (IITC), Le Quy Don Technical University, Vietnam. He is also a Postdoctoral Fellow of the Department of Mathematical and Computing Sciences at Tokyo Institute of Technology. He received his B.S. and M.S in Computer Science from National Defense Academy, Japan, in 2005 and 2008 and his Ph.D. from Tokyo Institute of Technology, Japan, in 2015, respectively. He is a member of IPSJ Japan and IEEE. His research interests lie in the area of watermarking, network security, and computer vision.
Email: thanhtm@lqdtu.edu.vn