Hindawi
Security and Communication Networks
Volume 2022, Article ID 1941697, 11 pages
https://doi.org/10.1155/2022/1941697

Research Article

Understanding User-Level IP Blocks on the Internet

Yimo Ren,[1,2] Hong Li,[1,2] Ruinian Li,[3] Hongsong Zhu,[1,2] and Limin Sun[1,2]

[1] *School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China*
[2] *Beijing Key Laboratory of IoT Information Security Technology, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China*
[3] *Department of Computer Science, Bowling Green State University, Bowling Green, OH, USA*

Correspondence should be addressed to Hong Li; lihong@iie.ac.cn

Received 17 June 2022; Revised 12 July 2022; Accepted 21 July 2022; Published 17 August 2022

Academic Editor: Hamad Naeem

Generally, the devices on the Internet are identified by IP addresses. The users of IPs are those who use IPs on the Internet and are always different from their registers and operators. Since IPs are used as unique identifiers of devices, knowing users of IPs according to their multisource data is critical for experts to protect the security of the network. At present, there are only few methods to mine the users of IPs from their public data. To make matters worse, the existing methods do not make effective use of a large amount and multisource data, such as certificates, protocol banner, rDNS, location, topology, etc. As a result, the performances of existing methods are largely limited. To tackle this issue, we proposed ULIB, short for "Understanding User-Level IP Blocks on the Internet." ULIB is based on improved community detection to mine the users for as many blocks of IPs as possible. By analysing comprehensive attributes of IPs, ULIB is able to recognize users effectively. Meanwhile, we evaluated our methodology in the real world and the experiments demonstrated that the accuracy of ULIB is 74.20% and the coverage is 28.90% in a city of China, which outperforms other existing methods.

1. Introduction

Identifying the users of devices can enable numerous network security applications. For example, when a serious vulnerability such as Apache Log4j vulnerability [1] is exposed, the users should be promptly notified by regulators and security researchers, so that the vulnerability can be patched in time. And the wireless sensor networks are facing numerous tribulations regarding network coverage. That is because of uncouth deployment of the sensor nodes [2], which effects the security of users. The devices connecting to the Internet always use IP addresses or domain names as unique identities. Querying public databases such as Whois of IANA [3] and IPIP (https://www.ipip.net/) is a common way to identify the users of large-scale devices, but it has a lot of limitations: (1) many organizations recorded in the databases are the registers of IP addresses or domains, most of which are used by Internet service providers or cloud service providers, rather than the real users. (2) IP addresses can be sublet or sold, as Figure 1 shows, so the registered organizations may not be the real users and it is difficult to know as much as possible who uses the public devices on the Internet.

There is not only little research specifically aimed at the user identification of IPs, but also no analysis on the aggregation of IPs with users, namely, user-level IP blocks. Therefore, we proposed ULIB: Understanding User-Level IP Blocks on the Internet to detect and recognize high-quality user-level IP blocks in a fine-grained way. ULIB is a kind of top-down method based on community detection on knowledge graph to recognize the users of devices. At the same time, according to ULIB proposed in this paper, the network measurement is carried out in the real environment. The evaluation results show that the accuracy of ULIB is 74.20% and the coverage is 28.90%, in a city of China, which outperforms other existing methods. At the same time, we found most users have relatively small number of blocks (less than 10) and the mean size of blocks is small (less than 10, too). Through all the experiments, it is clear that identifying User-Level Blocks can be more helpful to understand and analyse the usage of IPs.

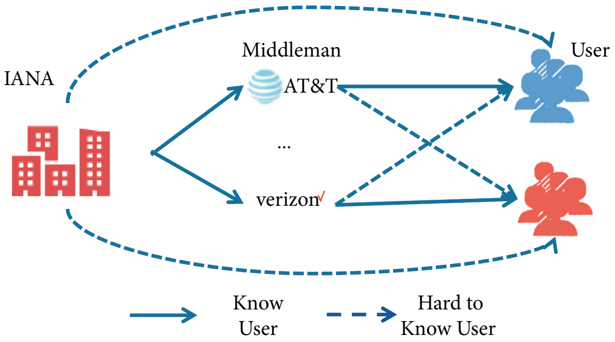


Figure 1: There is a "data fortress" in the allocation mechanism of the IP or domain.

Our contributions are summarized as follows:

(1) This paper defines three types of owners for IPs, namely, register, operator, and user, for network security. The paper also detects and recognizes the user-level IP blocks in real environment and demonstrates the high performance as well.

(2) Louvain algorithm, a traditional community detection method, is improved to realize the clustering on the knowledge graph of IPs, enhanced by a Siamese Deep Neutral Network.

(3) Aiming at the rough results obtained after community detection, a top-down method is adopted to achieve fine-grained recognition of user-level IP blocks.

2. Problem Formulation

2.1. Motivation. Our work is motivated by several famous and influenced cyber security incidents, such as Apache Log4j vulnerability [1]. Log4j is a widely used tool for gathering log information and is widely deployed in websites. The Log4j vulnerability allows attackers to execute code remotely to control the websites. According to media reports, the vulnerability affected many Internets service companies, including Apple, Amazon, IBM, Microsoft, and Twitter. After regulators and security researchers detect the affected websites, the users should be promptly notified to protect potential attacks.

2.2. Statement. In the life cycle of a public IP address, there may be three owners. Originally, all IP address spaces were managed directly by the IANA or ICANN [3]. Public Internet registries, such as Regional Internet registries (RIR), National Internet Registries (NIR), and Local Internet Registries (LIR), may acquire parts of the IP addresses from IANA or ICANN. Finally, Internet Service Providers (ISP) obtain useable addresses from LIR, NIR, or RIR and assign them to end users. The three kinds of owners are characterized as follows:

(1) Register: the register of an IP address refers to the organization that obtains the IP address from IANA. The registers can be RIR, NIR, or LIR.

(2) Operator: the operator of an IP address refers to the organization that obtains the IP address from the registers and assigns it to the end user.

(3) User: the user of an IP address refers to the organization that gets the address from its ISP and uses the IP address for particular purposes. The user can be a government agency, a company, a school, or a person, etc.

While the registers and operators of an IP address can be identified by querying public databases, it is hard for us to recognize the end user of an IP that is hidden in multisource data. Thus, in this work, we only focus on identifying the users of IP addresses. At the same time, in order to facilitate visualization and analysis, we classify the IP addresses of each user in clusters, called user-level IP blocks.

2.3. Challenge. Identification of users of IPs is particularly challenging due to the following reasons.

Nowadays, users may use various Internet services from providers all over the world. Also, wide references of cloud services and other mechanisms, such as dynamic allocation, used by operators, and the fact that a single user can own multiple devices associated with multiple IP addresses have all contributed to the complexity of identifying the users of IPs. As a result, the IP addresses not only have local aggregation, but also have overall dispersion, making it hard to find out all User-Level Blocks precisely. For example, a user may own several IP addresses in p1.0/24 and p2.0/24 at the same time, while p1 and p2 are two different prefixes of IPs. Therefore, it is difficult to directly obtain all IP blocks of users simply based on topology, location, and others.

3. Related Work

3.1. User Recognition. It seems easier to find the owners of devices, using public databases such as Whois [4, 5] and DNS [6, 7]. But these methods can only get the register of websites. However, the registers are always not their owners. Certainly, these are some other methods to identify owners. AIWEN [8], a commercial company, and Wang [9] extract the owners of domains by regex rules such as "Copyright@ (.+?)." But that kind of methods only uses the websites instead of heterogeneous data of devices, resulting in low accuracy and coverage.

3.2. Clusters and Communities. Clustering is a process of classifying data into different classes or clusters. Objects in the same cluster have great similarity, and objects among different clusters have great dissimilarity [10]. Community detection [11, 12] reveals the relations among the nodes in a network, which is essentially clustering the network. Louvain algorithm [13] is a modularity-based community detection algorithm that can achieve fast clustering of network nodes, especially for networks with lots of nodes but fewer links. And there are lots of community detection methods [14, 15] based on Louvain algorithm. But all the above algorithms could only realize the clusters on graph with its structure, leading to rough aggregation of nodes.

Deep graph neural network has been adopted in community detection [16] in recent research works. However, the deep learning method is usually time consuming, making it difficult to adapt to constructed knowledge graphs based on data in real environment to build user-level IP blocks.

The current IP block detection methods mainly focus on the traditional AS [17, 18] and CIDR [19, 20]. The detection of AS depends on the operator of the autonomous system (AS), and the CIDR method depends on the IP address and prefix length. Generally, an AS is composed of hundreds of thousands of addresses. In CIDR, the/24 network has 256 addresses and the/30 network has only 4 addresses. For the IP blocks used by small companies, the division between AS and CIDR may be too large; for large companies, CIDR may be too partitioned, with uncertain length of CIDR prefix and number of CIDR subnets. So, the existing methods cannot meet the demands of fine-grained user-level IP blocks detection.

4. Methodology

In this paper, we proposed a method, called ULIB, that can identify the users to which the IP addresses are owned. As mentioned before, the public registries and operators can be easily identified by querying public databases such as Whois, AS, etc. Therefore, our method will focus on analysing users of the IP addresses.

ULIB firstly mines the users of each single device through their SSL certificates and protocol banners, which we call **Seed Mining**. Secondly, a knowledge graph, named as **device book**, is constructed to introduce other IP addresses based on the seeds, the relations between seeds and others, and a set of multidimensional attributes, such as DNS, AS, location, devices, etc. Then, **Community Detection** clusters the IPs on the device book with maximizing the improved community fitness. Finally, a top-down method in **Community Recognition** is adopted to achieve fine-grained recognition of user-level IP blocks.

For the identification of users, the summary of our methodology is shown in Figure 2.

4.1. Seed Mining. For a single device, we identify the user from SSL certificates and protocol banners. Figure 3 shows examples of users in SSL certificates and protocol banners, as well as the users of the devices that may be contained in the certificates and banners and circled by red boxes, which we also call IP seeds.

4.1.1. SSL Certificates. SSL, serving as the backbone of Internet security, is a standard security technology that is enabled by digital certificates. Jain [21] introduced how the SSL certificates work and Clark and Van Oorschot [22] discussed the mechanism and security of issuing certificates. The subject in SSL is the owner of the website, which is probably the user of the IP we defined.

For SSL certificates, the regexes and keywords we used are (subject: (. *?)) and (O = (. *?);). But previous researchers have found lots of obvious errors of certificates [23, 24], so filters as follows are necessary:

(1) We dropped the self-signed certificates.

(2) We dropped the expired certificates.

(3) We dropped the certificates with risk issuers or subjects.

4.1.2. Protocol Banner. Protocol banners are the public response data when we send queries to IPs and provide information about services and applications. For example, the banners of a device using HTTP are its HTTP content, and the banners of a device using SSH or FTP are the information returned from the device at login. Through banner grabbing and analysis, devices running on a network can be easily identified [25, 26], as well as the organizations to which the devices are owned, which makes it possible to further identify the real users of those IP addresses.

As aforementioned, protocol banners are the public response data when we send queries to IPs. For example, the protocol banners of devices using HTTP are their HTTP content, and the protocol banners of devices using SSH or FTP are the information returned from devices at login, while we do not need to guess the user names and passwords.

Because of the various formats of the returned information, it is hard to use single regexes or keywords to extract possible users; therefore we used Named Entity Recognition (NER) methods to enhance Seeds Mining. NER is often used to extract some key information from natural language. Goyal et al. [27] present a survey of developments and progresses made in NER and Georgescu et al. [28] enhances the process of diagnosing and detecting possible vulnerabilities within an Internet of Things (IoT) system by using a NER-based solution.

We build a NER model to extract users from banners, coming from response data when we access to the IPs. BiLSTM-CRF is one of common and mature methods of NER [29], and we choose to build a pretrained BiLSTM-CRF based on Wiki to extract users from protocol banners.

4.2. Device Book. From the data of a single device, only the user of the IP can be directly extracted. However, many IPs lack the information directly exposed on the Internet, which makes it difficult to mine their users, resulting in the low coverage of the existing methods. Therefore, this paper constructs the device book of IPs, introducing the relations between IPs.

The Centre for Applied Internet Data Analysis (CAIDA) [30] builds research infrastructures for large-scale Internet data collection and provides such services to scientific research communities. CAIDA collects data by sending scamper probes continuously to destination IP addresses that are randomly selected from each routed IPv4/24 prefix

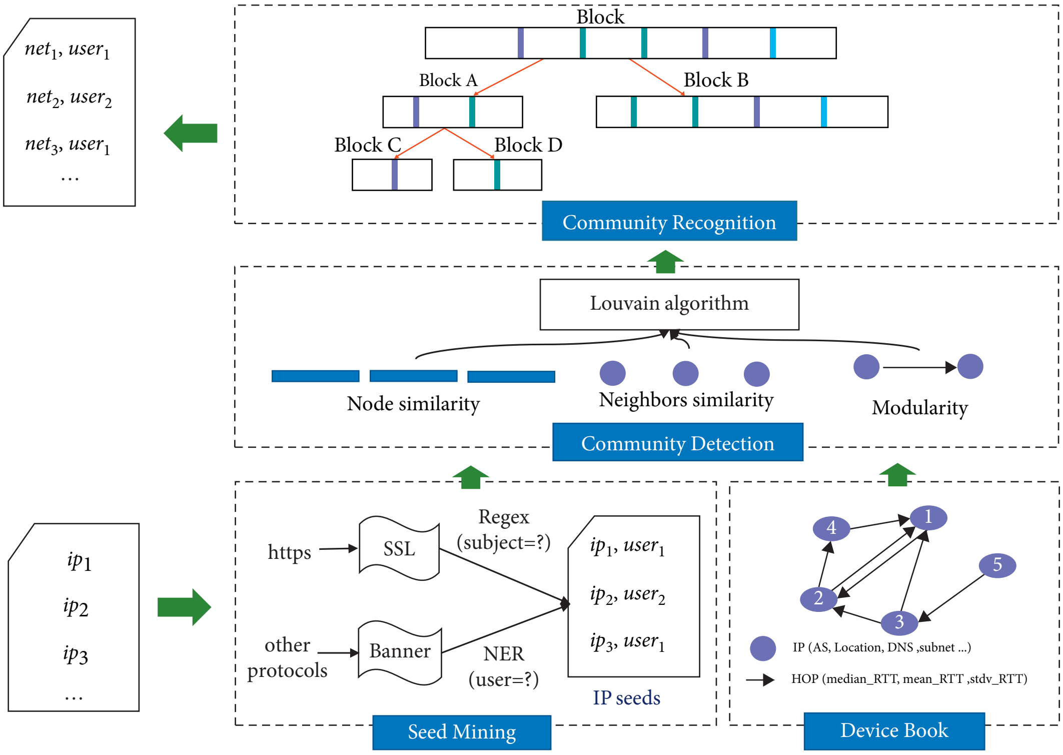
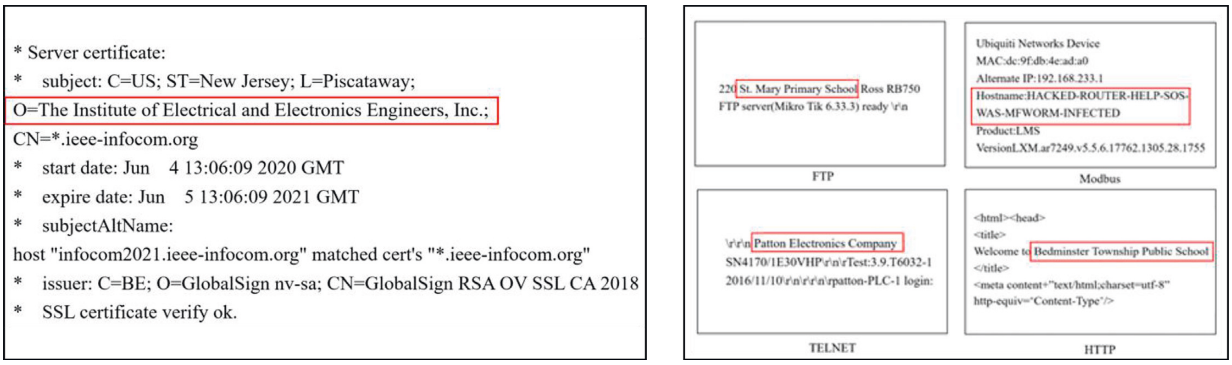


FIGURE 2: The structure of ULIB.



```
* Server certificate:
*   subject: C=US; ST=New Jersey; L=Piscataway;
O=The Institute of Electrical and Electronics Engineers, Inc.;
CN=*.ieee-infocom.org
*   start date: Jun   4 13:06:09 2020 GMT
*   expire date: Jun   5 13:06:09 2021 GMT
*   subjectAltName:
host "infocom2021.ieee-infocom.org" matched cert's "*.ieee-infocom.org"
*   issuer: C=BE; O=GlobalSign nv-sa; CN=GlobalSign RSA OV SSL CA 2018
*   SSL certificate verify ok.
```

(a)

```
220 St. Mary Primary School Ross RB750
FTP server(Mikro Tik 6.33.3) ready \r\n
```

FTP

```
Ubiquiti Networks Device
MAC:dc:9f:db:4e:ad:a0
Alternate IP:192.168.233.1
Hostname:HACKED-ROUTER-HELP-SOS-
WAS-MFWORM-INFECTED
Product:LMS
VersionLXM.ar7249.v5.5.6.17762.1305.28.1755
```

Modbus

```
\r\r\n Patton Electronics Company
SN4170/1E30VHP\r\n\rTest:3.9.T6032-1
2016/11/10\r\n\r\r\n\rpatton-PLC-1 login:
```

TELNET

```
<html><head>
<title>
Welcome to Bedminster Township Public School
</title>
<meta content+"text/html;charset=utf-8"
http-equiv="Content-Type"/>
```

HTTP

(b)

FIGURE 3: Examples of users in SSL certificates and protocol banners. (a) Users in SSL certificates of IPs. (b) Users in protocol banners of IPs.

on the Internet. In this way, a random address in each prefix is probed approximately every 24 hours by one probing cycle (https://www.caida.org).

In order to integrate the heterogeneous data as much as possible and effectively build the device book of IPs, we select AS, CIDR Subnets, location, and DNS as the attributes of nodes, CAIDA topology as the links, and the median, mean, and stand deviation of RTTs as the properties of the links.

When building the device book, we need to represent the knowledge of the IPs. IP represents the attributes of IPs, respectively. Hop represents the attributes of links, which can represent several groups of triples (src_IP, Hop, dst_IP). In triples, src_IP represents the attribute of start device, dst_IP is the attribute of target device, and Hop is the attributes of links between two devices. For the device book, V represents all devices of the graph, and E represents all links of the graph.

4.3. Detection. Because the objective of Louvain algorithm is only modularity, it cannot effectively use the attributes of IP to detect the user-level IP blocks. Therefore, ULIB unites nodes similarity, neighbour nodes similarity, and modularity as the optimization objective. Therefore, the community detection in this paper can better integrate attributes and topology of IPs to realize the detection and recognition of highly precise user-level IP blocks.

4.3.1. Node Similarity Model. Node Similarity Model relies on the attributes of nodes as features and user seeds of IPs as labels. Node Similarity Model adopts pairwise way to construct $(\mathrm{IP}_1, \mathrm{IP}_2, \text{label})$, in which the label is 1 when the users of IP_1 and IP_2 are the same and 0 when the users are different. On the basis of ensuring the effect and efficiency, the Siamese DNN [31] is trained to learn to judge whether two IPs have the same user based on the attributes of nodes, namely, oriented similarity model.

Among the four kinds of selected attributes, IPs have certain aggregation obviously. Using AS, location, and Subnet/24, the aggregation of IPs is more compact to cluster users of IPs; in DNS, the aggregation is sparser, while some DNS can directly show the related user information, such as SOHO, ICBC, and so on. Therefore, the paper used the Bow model to convert DNS to numbers, and the rest of the attributes can be directly vectored.

Generally speaking, unsupervised measures such as cosine similarity can be used to calculate similarity between two IPs by their attributes. However, cosine similarity considers attributes equivalently and does not consider users of IPs. At the same time, cosine similarity also does not consider the relationship between the IPs with users obtained in advance. To overcome the shortcomings of unsupervised measures, ULIB constructs a Siamese DNN to supervised learning of the similarity of IPs, and the model is shown in Figure 4.

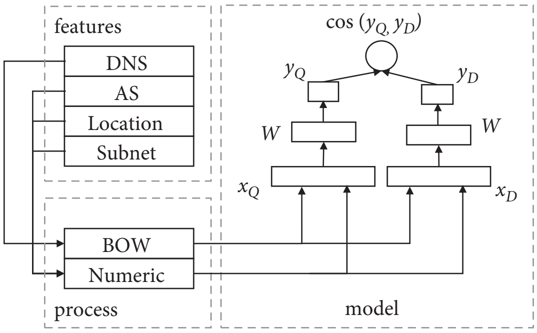


FIGURE 4: Structure of Node Similarity Model.

The similarity of IP u and v is as follows:

$$\mathrm{sim}_1(u, v) = \cos(y_Q, y_D) = \cos(W \cdot x_Q, W \cdot x_D), \quad (1)$$

where x represents the attributes of IPs. W is the main parameter of Siamese DNN, which is learned by labelled data constructed from IPs seeds. Siamese DNN calculates such similarities and builds a space in which IPs belonging to the same users are closer while the IPs belonging to different users are more discrete in the owner space. By reducing the model parameters, the amount of calculation is greatly reduced.

4.3.2. Neighbours' Similarity Model. Intuitively, IPs sharing the same neighbouring routers are considered to belong to the same user. For two IPs u and v, the more similar their neighbouring routers are, the closer the similarity is to 1. Therefore, the neighbours' similarity is defined as

$$\mathrm{sim}_2(u, v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{\sqrt{|\Gamma(u)||\Gamma(v)|}}, \quad (2)$$

where $\Gamma(u) = \{u \epsilon V | (u, v) \epsilon E\} \cup \{u\}$.

4.3.3. Modularity Model. Modularity [14] is a commonly used property to measure the division of network communities. The value of modularity mainly depends on the distribution of nodes in the communities of the network, namely, the community division of the network. The closer the value of modularity to 1, the stronger the community structure divided in the network. Therefore, the goal of network community division could be set to maximize the modularity Q.

$$Q = \frac{1}{2m} \sum_{vw} \left(A_{vw} - \frac{k_v k_w}{2m} \right) \delta_{vw}, Q \in [-0.5, 1), \quad (3)$$

where A is the adjacency matrix of the graph, k is the degree of the node, m is the total number of edges, and δ_{vw} quantifies whether v and w are in the same community.

4.3.4. Community Fitness. The optimization goal of community detection consists of three parts, which combines the node similarity, neighbour similarity, and modularity.

For node similarity and neighbour similarity, Silhouette Coefficient is a way to evaluate the whole performance of nodes cluster. For a community i , Silhouette Coefficient is defined as

$$S_i = \frac{1}{n} \sum_i \frac{b_i - a_i}{\max(a_i, b_i)}, S_i \epsilon [-1, 1], \quad (4)$$

where

$$a_i = \frac{1}{n} \sum_{p \epsilon C_i} \mathrm{distance}(p, i) (p \neq i),$$

$$b_i = \min \left(\frac{1}{n} \sum_{p \notin C_i} \mathrm{distance}(p, i) \right), \text{and} \quad (5)$$

$$\mathrm{distance}(p, i) = \frac{1}{\mathrm{sim}(p, i)} - 1.$$

Then the optimization of community detection is defined as

$$\begin{aligned} FC &= FC_1 + FC_2 + FC_3 \\ &= \alpha \times \frac{1}{n} \sum_i \frac{b_i - a_i}{\max(a_i, b_i)} |\mathrm{sim}_1 + \beta \\ &\quad \times \frac{1}{n} \sum_i \frac{b_i - a_i}{\max(a_i, b_i)} |\mathrm{sim}_2 + \gamma \\ &\quad \times \frac{1}{2m} \sum_{vw} \left(A_{vw} - \frac{k_v k_w}{2m} \right) \delta_{vw}. \end{aligned} \quad (6)$$

Among them, $\alpha = 0.6, \beta = 0.2, \gamma = 0.2$ are weight parameters based on prior knowledge and experience.

After defining *FC*, this paper used an improved Louvain algorithm to maximize the optimization goal to get the temporary communities.

In a word, as in **Community Detection** of Figure 2, ULIB first calculates the **node similarity, neighbours similarity of IPs,** and **modularity of the whole device book**. Then ULIB calculates the total fitness as the target for Louvain algorithm to get the final communities for IPs until reaching the best performance or max iterations **Epoch**.

4.4. Recognition. The user-level IP blocks got from community detection are relatively rough. For an IP block, there may be different IPs seeds with their users, making it hard to find out which user of IP seed is the user of the IP block. Therefore, ULIB uses the top-down method to divide rough IP blocks into fine-grained IP blocks until each IP block has a unique user.

For example, in **Community Recognition** of Figure 2, **Block** is the IP block obtained from the first community detection. Different colours in the block represent seeds with different users. **Block A** and **Block B** are the IP blocks obtained from the second community detection. **Block A** does not meet the recognition rules, so it is necessary to conduct the third community detection to obtain **Block C** and **Block D**. Therefore, **Blocks B, C, D** are final user-level IP blocks we could get using the methodology.

5. Experiment

5.1. Motivation. The data used for experiments in this paper is as follows in Table 1. The Topology Link represents the number of triples after processing the path provided by CAIDA. SSL Certificates and Protocol Banner represent the number of IPs with users identified from the corresponding data, which are the existing basis of work. Total Seeds represent the result of removed duplication of SSL Certificates and Protocol Banner, and Internet Edges represent the number of goals identified by function of IPs.

5.2. Evaluation

5.2.1. Detection Metric. The Adjusted Rand Index [32] is used to measure the performance of user-level IP blocks detection. ARI is used to measure whether the algorithm can divide IPs with the same user into a cluster. The closer the ARI is to 1, the stronger the performance of detection is.

$$\mathrm{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i (a_i/2)\sum_j (b_j/2)\right]/\binom{n}{2}}{1/2\left[\sum_i\binom{a_i}{2} + \sum_i\binom{b_j}{2}\right] - \left[\sum_i (a_i/2)\sum_j (b_j/2)\right]/\binom{n}{2}}, \tag{7}$$

where n_{ij} is the number of samples that should be recognized into class i while being recognized into class j actually, and the sum $\sum_{ij}\binom{n_{ij}}{2}$ is the correct number of results. $1/2\left[\sum_i\binom{a_i}{2} + \sum_i\binom{b_j}{2}\right]$ indicates that the classification is all correct, and $\left(\left[\sum_i\binom{a_i}{2}\sum_j\binom{b_j}{2}\right]/\binom{n}{2}\right)$ indicates the expectation of the number of correct results.

5.2.2. Recognition Metric. In this part, we evaluated our methodology by calculating the precision, recall, and F1. We calculated the precision and recall [33] of our methodology based the evaluation dataset. The definitions are as follows:

$$\begin{aligned} \text{Precision} &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \\ \text{Recall} &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \\ F1 &= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \end{aligned} \tag{8}$$

where TP presents the number of the true positive, FP denotes the number of the false positive, and FN is the number of the false negative. For those IPs that could not be identified, we treated the organization results as none.

5.3. Results. In this section, we analyse the details of the proposed framework ULIB in three aspects: (1) Accuracy of Node Similarity Calculated by ULIBl (2) ARI of Community Detection; (3) Accuracy of Community Recognition.

5.3.1. Node Similarity Model. During training the Node Similarity Model, the extracted seeds construct $(\mathrm{IP}_1, \mathrm{IP}_2, \text{label})$ using pairwise way, in which the label is 1 when the users are the same and 0 when the users are different. 10000 samples were randomly selected from the constructed data and divided into 70% train set and 30% test set. The parameters of the model are shown in Table 2.

The process of the model training is in Figure 5, where accuracy represents the accuracy performance of the model in the training set, and val_accuracy is the accuracy performance of the model on the test set.

The results show that the training of Node Similarity Model tends to be stable and the final accuracy is 0.62. However, it is unsatisfied for the user identification of IPs.

Table 1: The description of the dataset.

Time	Type	Number
20190101–20190601	Topology Link	13.17 million
	SSL Certificates	56K
	Protocol Banner	114K
	Total Seeds	158K
	Internet Edges	2.53 million

Table 2: Parameters of Node Similarity Model.

Parameter	Value	Parameter	Value
Train set number	7000	Test set number	3000
Epoch	30	Layers	1
Drop out	0.1	Layer unit	128

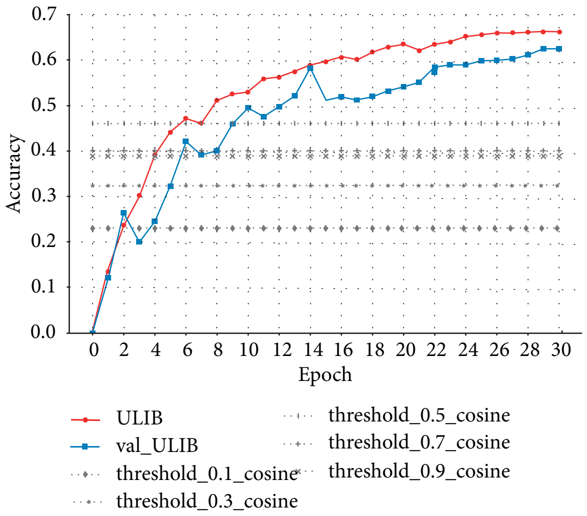


Figure 5: Training process of Node Similarity Model in ULIB, compared with cosine similarity.

Therefore, based on Node Similarity Model, the community detection is realized by combining Neighbours Similarity Model and Modularity Model to improve the performance of user identification.

At the same time, this paper evaluates performance with cosine similarity and sets different thresholds to calculate the accuracy on the test set in Figure 5. It can be seen that when the threshold of cosine similarity is about 0.5, the max accuracy is 0.46 and the performance is the best on the test set. Obviously, cosine similarity is weaker than the supervised Node Similarity Model proposed in this paper.

5.3.2. Community Detection. After calculating community fitness *FC*, the communities, namely, IP blocks, without users could be achieved, by maximizing community fitness. The performance of ULIB on Community Detection is shown in Figure 6, compared with traditional method AS and CIDR. We choose typical prefixes 24 and 28 for CIDR, namely, Subnet/24 and Subnet/28.

As can be seen from the Fitness-ULIB in Figure 6, the process of community detection tends to be stable, which means ULIB could divide the IPs into different communities successfully. The larger the ARI is, the better the corresponding method can divide the IPs with the same user into a community. We can see that the ARI of ULIB can reach 0.87, which is higher than those of AS, Subnet/24, and Subnet/28.

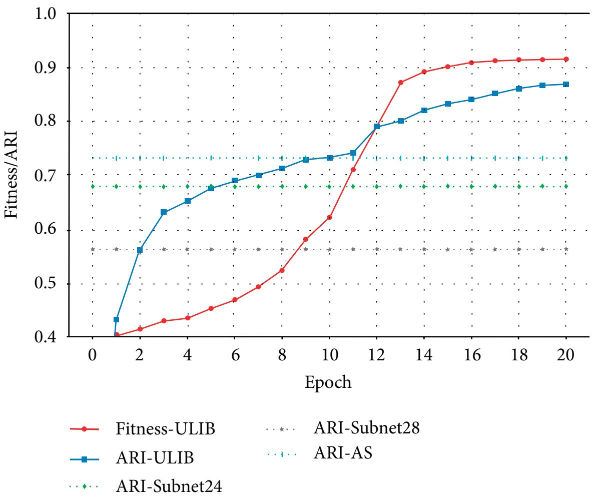


Figure 6: Performance of community detection.

5.3.3. Community Recognition. For recognition of the user of IP blocks, we measure it from two aspects: one is the performance in domain: only the performance on IP seeds is evaluated, and ablation experiments were performed at the same time. The other is the performance out of domain, that is, the Internet measurement in real environment; the model is evaluated by randomly selecting samples.

(1) Ablation. The IP seeds are randomly divided into training set and test set, and the accuracy and coverage are calculated. At the same time, in order to compare the final results of different parameters of community fitness, we set up multiple groups of parameters for experiments. Among them, $\alpha = 1, \beta = 0, \gamma = 0$ represent only using Node Similarity Model; namely, ULIB uses standard Louvain algorithm to detect communities, $\alpha = 0, \beta = 1, \gamma = 0$ represent only using Neighbours Similarity Model, and $\alpha = 0, \beta = 0, \gamma = 1$ represent Modularity Model.

It can be seen from Table 3 that, due to the lack of multisource data fusion, Node Similarity Model, Neighbours Similarity Model, and Modularity Model are not as good as ULIB. ULIB outperforms them because ULIB has an overall consideration on attributes, neighbours of IPs, and structure of whole network, which are related to users of IPs.

At the same time, we calculate the performance of our method under different seeds numbers, and the results are shown in Figure 7. With the increase of seeds number, more IPs with accurate users would pass more accurate labels to other IPs by community detection and recognition; thus the performance of ULIB would has a significant improvement.

Table 3: Performance of ablation.

Model	Parameter	Dataset	Precision (%)	Recall (%)	F1 (%)
Node Similarity Model	$\alpha=1,\ \beta=0,\ \gamma=0$	Train	69.78	54.62	61.28
		Test	66.42	52.32	58.53
Neighbours Similarity Model	$\alpha=0,\ \beta=1,\ \gamma=0$	Train	73.45	50.24	59.67
		Test	**70.89**	49.80	58.50
Modularity Model	$\alpha=0,\ \beta=0,\ \gamma=1$	Train	69.57	**59.78**	64.30
		Test	65.59	**56.54**	60.73
Node + Neighbours Similarity Model	$\alpha=0.5,\ \beta=0.5,\ \gamma=0$	Train	**74.89**	**59.87**	**66.54**
		Test	70.76	55.42	**62.16**
Node Similarity + Modularity Model	$\alpha=0.5,\ \beta=0,\ \gamma=0.5$	Train	**73.98**	59.45	**65.92**
		Test	**70.82**	**55.67**	**62.34**
Neighbours Similarity + Modularity Model	$\alpha=0,\ \beta=0.5,\ \gamma=0.5$	Train	67.67	53.45	59.73
		Test	67.19	52.44	58.91
ULIB	$\alpha=0.6,\ \beta=0.2,\ \gamma=0.2$	Train	**74.22**	**59.67**	**66.15**
		Test	**71.23**	**62.23**	**66.43**

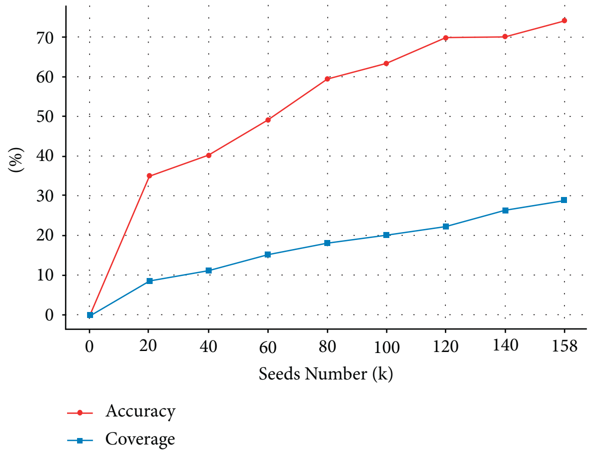


Figure 7: Seeds number and performance.

(2) Contrasts. Using IP seeds, the algorithm is trained and tested in real environment. In this paper, we choose the real environment in a city of China for measurement. Limited to conditions to get the results of other methods (Zoomeye, a mature business system, https://www.zoomeye.org), we randomly select 1000 samples in the results to calculate the performance. Moreover, for a more comprehensive comparison, ULIB also has a comparison with seeds in Table 4, which represents the results from Section 4.1 of Seed Mining. Seeds share the same idea as researches [8, 9] in **Related Work** section.

It can be concluded from Table 4 that the method used in this paper can achieve a relatively high coverage rate of 28.90% under the premise of ensuring a certain accuracy of 74.20% with the randomly selected 1000 samples in the real environment.

Compared with seeds and Whois, ULIB achieves a competitive performance at accuracy but outperforms them much at the coverage. That is because seeds only recognize users from SSL Certificates and Protocol Banner, but without similarities between them. And most organizations of IPs recorded in Whois are their registers or operators, not the users. For example, ChinaNet is a common organization recorded in Whois, but at many times, it is an operator but not user of IPs.

Table 4: Contrasts between ULIB and other methods.

Dataset	Model	Coverage (%)	Accuracy (%)
A city of China	Whois	3.10	87.10
	Seeds	6.20	83.87
	Zoomeye	21.50	71.16
	ULIB	28.90	74.20

Compared with Zoomeye, which mainly uses the domains to mine the associated organization, ULIB outperforms both at coverage and accuracy. That is mainly because Zoomeye focuses on those IPs whose location is relatively obvious and clear. So, some of the results from Zoomeye include locations such as buildings, parks, roads, etc. Therefore, Zoomeye has a poorer performance in the issue about user recognition of IPs.

5.4. Measurement. In this section, we conduct Internet Measurement by ULIB for user-level IP blocks, with analysing and visualizing the blocks simultaneously.

As can be seen from Figure 8, the number of blocks divided by ULIB is less than that divided by Subnet28, but far more than those divided by Subnet24 and AS. Also, the average IP number of each block is more than that of Subnet28, but less than that of AS and Subnet24. Combined with the results in Results section, it is clear that the User-Level Blocks of ULIB are more practical than those of other methods.

Figure 9 shows the distributions of Blocks Number and Mean Block Size from ULIB. The results of Network Measurement show most of users have relatively few blocks (<10) and mean size of blocks is small (<10), which means users in the city of China possibly use Network Address Translation (NAT) to fully use limited public IPs. Also, we analysed the 21 users having most blocks (>500) and the 15 largest blocks (>60), and we found that they are related to cloud service providers, such as Alibaba Cloud and Tencent cloud.

In order to verify the effect of ULIB on identifying User-Level Blocks of IPs, we selected a typical block for visualization, as shown in the Figure 10. In the figure, white

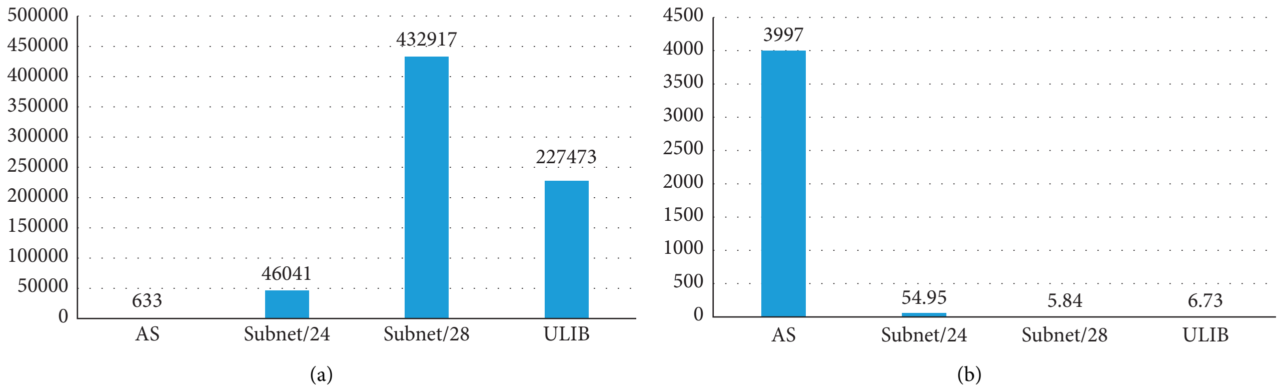


FIGURE 8: Total block number and average IP number of each block from ULIB and other methods. (a) Total block numbers. (b) Average IP number of each block.

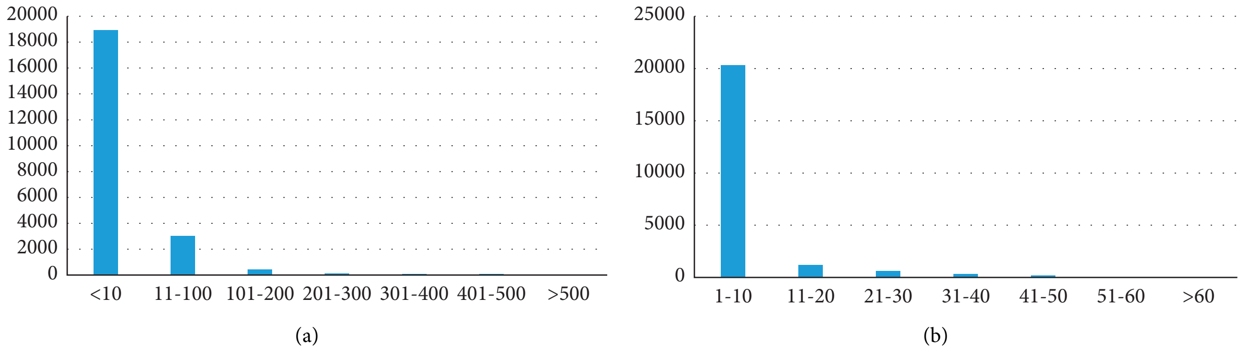
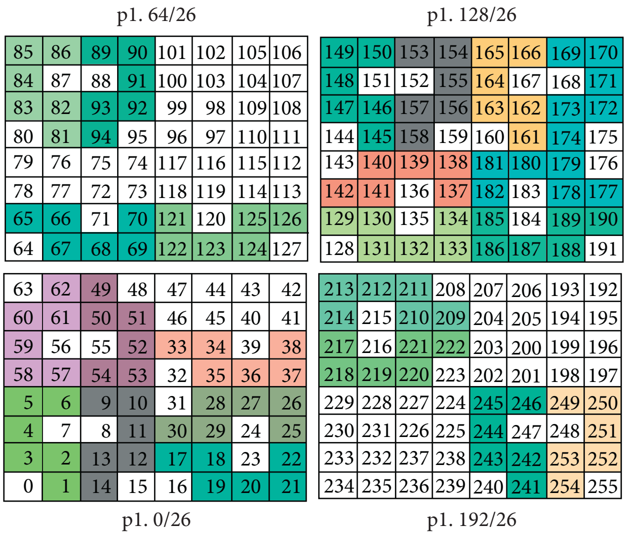


FIGURE 9: The distribution of IP blocks from ULIB. (a) The distribution of the blocks number. (b) The distribution of the mean block size.



85	86	89	90	101	102	105	106
84	87	88	91	100	103	104	107
83	82	93	92	99	98	109	108
80	81	94	95	96	97	110	111
79	76	75	74	117	116	115	112
78	77	72	73	118	119	114	113
65	66	71	70	121	120	125	126
64	67	68	69	122	123	124	127

149	150	153	154	165	166	169	170
148	151	152	155	164	167	168	171
147	146	157	156	163	162	173	172
144	145	158	159	160	161	174	175
143	140	139	138	181	180	179	176
142	141	136	137	182	183	178	177
129	130	135	134	185	184	189	190
128	131	132	133	186	187	188	191

63	62	49	48	47	44	43	42
60	61	50	51	46	45	40	41
59	56	55	52	33	34	39	38
58	57	54	53	32	35	36	37
5	6	9	10	31	28	27	26
4	7	8	11	30	29	24	25
3	2	13	12	17	18	23	22
0	1	14	15	16	19	20	21

213	212	211	208	207	206	193	192
214	215	210	209	204	205	194	195
217	216	221	222	203	200	199	196
218	219	220	223	202	201	198	197
229	228	227	224	245	246	249	250
230	231	226	225	244	247	248	251
233	232	237	238	243	242	253	252
234	235	236	239	240	241	254	255

FIGURE 10: A/24 block and its prefix are anonymized. Addresses of IPs are on a Hilbert curve.

means the users have not been identified yet, and other colours indicate different results of recognized users.

As can be seen from Figure 10, though some of User-Level Blocks have certain aggregation on the addresses, more are intermittently distributed in the/24 block. In this case, it is obviously difficult to identify all User-Level Blocks using AS and CIDR. Therefore, identifying User-Level Blocks can be more helpful to understand and analyse the usage of IPs.

6. Conclusions

With the increasingly wide applications of networking devices in industry and life, how to effectively operate and protect the devices will become a top priority for the countries and enterprises. Knowing the users of IPs can make it easier for operators to manage and protect the network. In this paper, the user-level IP block is divided by an improved community detection and recognition method on the device book base on Louvain algorithm and Deep Neutral Network. Compared with the existing methods on the detection of IP blocks, we further improve the performance of community detection by using the heterogeneous data of IPs, and a top-down method is adopted to achieve fine-grained recognition of user-level IP blocks. We evaluate the performance of our proposed method in real-world networks. The evaluation results show that the accuracy of ULIB is 74.20% and the coverage is 28.90% in a city of China, which outperforms other existing methods. At the same time, we found most users have relatively small number of blocks (less than 10) and the mean size of blocks is small (less than 10, too). Through all the experiments, it is clear that identifying User-Level Blocks can be more helpful to understand and analyse the usage of IPs.

The detection and recognition of user-level IP blocks are drawing more and more attentions to researchers from multiple disciplines. In the future, we will apply this method in more networks to achieve finer results. In the meantime, we will try to improve the accuracy and coverage of ULIB based on more multisource data of devices.

In addition, the popular deep neural network may also be able to effectively integrate attributes and topology of IPs. However, due to the complex structure and long running time of deep learning models, this paper does not consider it temporarily. But in future, the deep learning models may enhance the performance of ULIB.

Data Availability

The data and materials of this study are available from the corresponding author or first author upon reasonable request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The research was completed by the first author as a Ph.D. student in the Institute of Information Engineering, Chinese Academy of Sciences.

References

[1] Nvd - cve-2021-44228, “Nvd - cve-2021-44228,” 2021, https://nvd.nist.gov/561 vuln/detail/CVE-2021-44228.2021.

[2] S. Shahzad, “Culminate Coverage for Sensor Network through Bodacious-Instance Mechanism,” *i-manager’s Journal on Wireless Communication Networks*, vol. 8, no. 3, pp. 1–9, 2019.

[3] P. Palladino Nicola and M. Santaniello, *IANA Functions, ICANN, and the DNS War*, Springer, Berlin Germany, 2021.

[4] S. Liu, I. Foster, and S. Savage, “Who is. com? Learning to Parse WHOIS records,” in *Proceedings of the 2015 Internet Measurement Conference*, pp. 369–380, Tokyo, Japan, October 2015.

[5] C. Lu, B. Liu, Y. Zhang et al., *From WHOIS to WHOWAS: A Large-Scale Measurement Study of Domain Registration Privacy under the GDPR*, NDSS, Australia, 2021.

[6] R. Romero-Gomez, Y. Nadji, and M. Antonakakis, “Towards Designing Effective Visualizations for DNS-Based Network Threat analysis,” in *Proceedings of the 2017 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pp. 1–8, IEEE, Phoenix, AZ, USA, October 2017.

[7] O. van der Toorn, M. Müller, S. Dickinson, C. Hesselman, A. Sperotto, and R van Rijswijk-Deij, “Addressing the challenges of modern DNS a comprehensive tutorial,” *Computer Science Review*, vol. 45, Article ID 100469, 2022.

[8] Y. Wang, D. Burgener, M. Flores, K. Aleksandar, and H. Cheng, “Towards Street-Level Client-independent IP Geolocation,” in *Proceedings of the 2011 8th USENIX Symposium on Networked Systems Design and Implementation (NSDI 11)*, pp. 365–379, Boston, MA, USA, March 2011.

[9] Y. Wang, X. Wang, H. Zhu, and Z. Hai, “ONE-geo: Client-independent IP Geolocation Based on Owner Name Extraction,” *International Conference on Wireless Algorithms, Systems, and Applications*, Springer, Cham Switzerland, pp. 346–357, 2019.

[10] K. R. Prasad, B. E. Reddy, and M. Mohammed, *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–14, Journal of Ambient Intelligence and Humanized Computing, 2021.

[11] S. Gupta and S. Deodhar, “Understanding Digitally Enabled Complex Networks: A Plural Granulation-Based Hybrid Community Detection approach,” *Information Technology & People*, 2021, (ahead-of-print).

[12] M. Magnani, O. Hanteer, R. Interdonato, L. Rossi, and A. Tagarelli, “Community detection in multiplex networks,” *ACM Computing Surveys*, vol. 54, no. 3, pp. 1–35, 2022.

[13] S. J. Xiong, S. B. Chen, C. H. Q. Ding, and L. Bin, “Large-Scale Network Representation Learning Based on Improved Louvain Algorithm and Deep Autoencoder,” in *Proceedings of the Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, pp. 446–459, Springer, Nanjing China, October 2020.

[14] J. Zhang, J. Fei, X. Song, and J. Feng, “An improved Louvain algorithm for community detection,” *Mathematical Problems in Engineering*, vol. 2021, Article ID 1485592, 14 pages, 2021.

[15] D. Liu, K. Huang, C. Zhang, W. Danling, and W. Shan, “Study on Discovery Method of Cooperative Research Team Based on Improved Louvain Algorithm,” *Scientific Programming*, vol. 2021, Article ID 3234280, 13 pages, 2021.

[16] R. Mittal and M. P. S. Bhatia, “Classification and comparative evaluation of community detection algorithms,” *Archives of Computational Methods in Engineering*, vol. 28, no. 3, pp. 1417–1428, 2021.

[17] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, and KG. claffy, “AS relationships: inference and validation,” *ACM SIGCOMM - Computer Communication Review*, vol. 37, no. 1, pp. 29–40, 2007.

[18] T. B. Paiva, Y. Siqueira, D. M. Batista, R. Hirata, and R. Terada, “BGP anomalies classification using features based on AS relationship graphs,” in *Proceedings of the 2021 IEEE Latin-American conference on communications (LATINCOM)*, pp. 1–6, Santo Domingo, November 2021.

[19] V. Fuller, T. Li, J. Yu, and K. Varadhan, *Rfc1519: Classless Inter-domain Routing (Cidr): An Address Assignment and Aggregation strategy*, RFC Editor, California, CA, USA, 1993.

[20] J. Jerrim, “Methods and systems for network flow analysis,” vol. 10, pp. 547–674, U.S. Patent, Virginia, VA, USA, 2020, 2020-1-28.

[21] R. Jain, *Secure Socket Layer (SSL) and Transport Layer Security (TLS)*, Washington University, Missouri, MO, USA, 2007.

[22] J. Clark and P. C. Van Oorschot, “SoK: SSL and HTTPS: Revisiting past challenges and evaluating certificate trust model enhancements,” in *Proceedings of the 2013 IEEE Symposium on Security and Privacy*, pp. 511–525, IEEE, Berkeley, CA, USA, May 2013.

[23] P. Szalachowski, S. Matsumoto, and A. Perrig, “PoliCert: Secure and Flexible TLS Certificate management,” in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 406–417, Scottsdale, AZ, USA, November 2014.

[24] X. Shi, S. Shi, M. Wang, K. Jonne, and Q. Chen, “On-device IoT certificate revocation checking with small memory and low latency,” in *Proceedings of the 2021 ACM SIGSAC*

Conference on Computer and Communications Security, pp. 1118–1134, Republic of Korea, November 2021.

[25] X. Feng, Q. Li, H. Wang, and L. Sun, “Acquisitional Rule-based Engine for Discovering {Internet-of-Things} Devices,” in *Proceedings of the 2018 27th USENIX Security Symposium (USENIX Security 18)*, pp. 327–341, Maryland, MD, USA, August 2018.

[26] T. Javed, M. Haseeb, M. Abdullah, and M Javed, “Using application layer banner data to automatically identify IoT devices,” *ACM SIGCOMM - Computer Communication Review*, vol. 50, no. 3, pp. 23–29, 2020.

[27] A. Goyal, V. Gupta, and M. Kumar, “Recent named entity recognition and classification techniques: a systematic review,” *Computer Science Review*, vol. 29, pp. 21–43, 2018.

[28] T. M. Georgescu, B. Iancu, and M. Zurini, “Named-Entity-recognition-based automated system for diagnosing cybersecurity situations in IoT networks,” *Sensors*, vol. 19, no. 15, p. 3380, 2019.

[29] K. Xu, Z. Yang, P. Kang, Q. Wang, and W Liu, “Document-level attention-based BiLSTM-CRF incorporating disease dictionary for disease named entity recognition,” *Computers in Biology and Medicine*, vol. 108, pp. 122–132, 2019.

[30] K. C. Claffy, “CAIDA: Visualizing the Internet,” *IEEE internet computing*, vol. 5, no. 1, 2001.

[31] P. S. Huang, X. He, J. Gao, D. Li, A. Alex, and L. Heck, “Learning Deep Structured Semantic Models for Web Search Using Clickthrough data,” in *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, pp. 2333–2338, San Francisco, SA, USA, October 2013.

[32] D. Steinley, “Properties of the haa,” *Psychological Methods*, vol. 9, no. 3, pp. 386–396, 2004.

[33] M. T. Kai, *Precision and Recall*, Springer US, Pennsylvania, PA, USA, 2011.