# Anti Device Detection in IoT Networks

Dixi Yao, Jimmyyao18@sjtu.edu.cn
Department of Computer Science, Shanghai Jiao Tong University, Shanghai, China

***Abstract*—IoT security is becoming more and more important and we find the problem of device detection attack in IoT networks. To solve the anti-device deteciton probelm, we propose the concept of safety index. Traditional method based on graph isomorphism and graph homomorphism cannot well establish the index. As a result, we propose the safety index based on structral entropy of a weighted and directed graph where we extend the structral entropy to IoT networks. Base on that, we propose an efficient deception algorithm. The experiments on different networks show the efficiency and outstanding performace of our deceptor. The project is now open source on https://github.com/daxixi/Extended-Structure-Entropy**

***Index Terms*—Internet of Things, secuity, information entropy, graph algorithm, deception algorithm**

## I. Introduction

With the fast development of internet of things (IoT), the IoT may even represent the next generation of industrial revolution. As the IoT is coming into the life of the public, the security problem of IoT networks is also critical [1], [2]. There exist various IoT cyber attacks , and attack based on device identification is a new one. The attacker can use common identification algorithms to detect particular device and launch high computational resources attack. If core device in the IoT network is attacked, severe consequences could occur and even directly affect physical word.

A simple but helpful way is to let IoT devices do some useless communications. In this case, the network of IoT will be added with some edges, but actaully these connections does not transfer actual instructions. The confused structure can protect the network. However, to achieve this, we need to measure the identification accuracy each time a communication is added, which is time-cost. To solve such problem, we propose to construct safety index similar to the idea of value net in reinforcement learning to indirectly measure the identification accuracy but directly represent the safety level of a system.

There exists various methods to model the safety index of a given network, the method based on Graph Isomorphism and Graph Homomorphism is a popular one [3]. However, such method still suffers from the problem of time-cost and the derived deception method cannot be applied to IoT networks. Information entropy theory is another direction, there exist many different approaches to depict information complexity in a graph [4]–[7]. However, they all fail to build the concept of structral information of a graph where the safety of a network lays. Li. etal. [8] porpose the concept of structral entropy to address such problem. Nevertheless, we cannot directly use structral entropy as safety index of a IoT network which is a directed and weighted graph. As a result, we modify and extend the structral entropy to build the concept of safety index.

After constructing the safety index, we derive a decption algorithm based on it which can construct a confused network in polyomial time cost. The proof shows the time complexity of the algorithm and then we evaluate the performance of such method on various datasets. The results show the method can well solve the problem of anti-device identification, and such method can aslo be extended to other types of networks.

Highlights of our contributions are as follows: we are the first to find the problem of device identification and propose anti device detection algorithms to our knowledge. We extend the structral entropy to weighted and directed graphs and use it to construct the safety index. We derive the deception algorithm based on such index. Then we evaluate the method on various networks.

The rest of this paper is as follows: Chapter 2 will introduce the related works. In Chapter 3, we will show our derivation of saefty index. In Chapter 4, we will introduce the deception algorithm. In Chapter 5, we will present the experimental results and corresponding discussions. Finally, we summarize the whole paper in the last chapter.

## II. Related Work

This section introduces the works most related to our work and some preliminaries for better understanding our work.

### A. IoT Security

IoT security is a family of techniques, strategies and tools used to protect IoT devices from becoming compromised. Ironically, it is the connectivity inherent to IoT that makes these devices increasingly vulnerable to cyberattacks [9]. Because IoT is so broad, IoT security is even broader. There are three layers in the IoT architecture, perception layer, network layer and application layer [10]. All of these layers are very vulnerable to cyber attacks. Especially the perception layer is straightly connected to our physical world. Because of these characters of IoT, the function of a particular device can be very unique. The attacker can reach network traffic information free of authentication. Once, the attacker identifies a core device, he can launch task specific attack. Moreover, because the IoT is connected to the physical world, attacker can not only launch cyber attack, he also can destroy the physical world, which will lead to serious consequence either in large-scale industrial network or in household intelligent network. As figure 1 shows, if the attacker break the firewall of the access point, he will be able to get full routing information and launch next round attack to specific device.

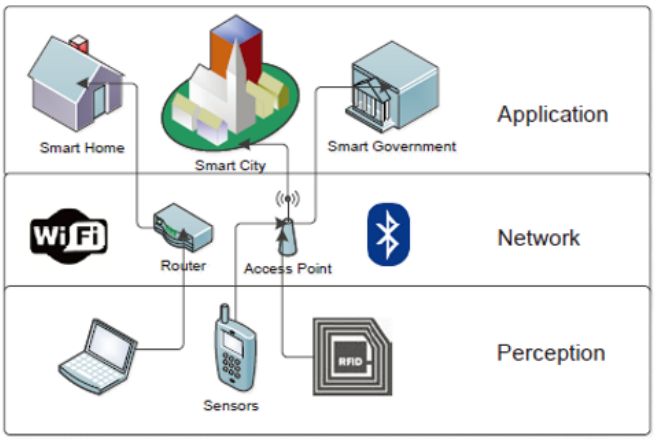


Fig. 1. A simple illustration of houshold IoT Netowrks

### B. IoT device identification

IoT identification (classification) was first introduced in [11] and it was originally designed against cyber attack. Because the attacker will forge IP address or MAC address to deceive the core device into believing the attacker as an authorized user or device, IP, MAC and other explicit identifications are out of utility. The core devices need to detect attacker through some implicit information such as traffic. If the requester's implicit information is not coordinate with real devices' patterns, the requests can be rejected [12].

However, this also provides those attackers an innovative attacking idea. The raw data behind these implicit information are more accessible than those explicit information. Attackers can easily get these raw data and identify those core devices. As a result, I come up with an idea that we can let devices to do some useless communication to fool the attackers. Useless communicaton represents the communication between devices but with no actual message in the packed segments. For example there are threes devices, the temperature monitor and the mobile phone both send a lot of communication requests to the robot. Then the robot can be easily identified. After, we let teperature monitor and the mobile phone do some useless communicatoins, the communicatio graph will be highly symmetric, so that it would be diffcult for attacker to mine information.

### C. Structure Entropy

Li and Pan proposed the structure entropy [8] to describe the complexity of a grpah. Comparing to information entropy, structure entropy can represent the information volume of high dimension data such as graph data while the low dimension information entropy can lose some ability to describe such information.Structure entropy of a graph can be defined by a coding tree with such formulation

$$H^T(G) = -\sum_{\alpha \neq \lambda, \alpha \in T} \frac{g_\alpha}{vol(G)} \cdot \log_2 \frac{vol(\alpha)}{vol(\alpha^-)} \tag{1}$$

where $G$ is a graph $(V, E)$, and $T$ is a coding tree of $G$. $g_\alpha$ is the number of edges from the complemennt of $T_\alpha$. $vol(G)$ is the volume of $G$ and the $\alpha^-$ is the parent node of $\alpha$ in T.

Actually, there exist rich substructures in a complex network. With a patition $\mathcal{P}$, we can dig the structure infromation latent in such partition. Li [8] defines the structual information of a network by a partition.

$$\begin{aligned} H_{\mathcal{P}}(G) &= \sum_{j=1}^{m} \frac{V_j}{2|E|} H\left(\frac{d_1^{(j)}}{2|E|}, \cdots, \frac{d_{n_j}^{(j)}}{V_j}\right) - \sum_{j=1}^{m} \frac{g_j}{2|E|} \log_2 \frac{V_j}{2|E|} \\ &= -\sum_{j=1}^{m} \frac{Vj}{2|E|} \sum_{i=1}^{n_j} \frac{d_{n_j}^{(j)}}{V_j} \log_2 \frac{d_{n_j}^{(j)}}{V_j} - \sum_{j=1}^{m} \frac{g_j}{2|E|} \log_2 \frac{V_j}{2|E|} \end{aligned} \tag{2}$$

**Definition 1** *Structure entropy of a network by a partition. Let $G = (V, E)$ be a network with $M$ partitions ($\{X_1, \cdots, X_m\}$). The structural information of $G$ by partition $\mathcal{P}$is as follows:*

*where $V_j$ is the volume of $X_J$, $2|E|$ is the volume of $G$, $g_j$ is the number of inter-edges. $d_{n_j}^{(j)}$ is the degrees of the $n_j$th node in $X_j$.*

In IoT device identification, it is critical to descsribe the structure information in different clusters and the clustering can be viewed as a kind of a partition. As a result, the structure entropy can have a good representation of a graph with a clustering partition though it cannot be straightly applied to IoT networks.

## III. Problem Formulation

### A. IoT Networks

The structural entropy defined in [8] can be applied to undirected and unweighted graphs. However, in IoT networks, the graphs have different topology structures. In IoT networks, we have such differences.

1) **Direction.** In IoT networks, the direction has different meaning. Some devices can only send requests to other devices while not requesting feedback. For example, the monitor can send several instructions to the robot while robot may only need to send few TCP replies back to the monitor only for confirming existence. However, a formal definition of structural entropy of directed graph has not been given.
2) **Edge Weights.** Another charateristics in IoT networks is that the edge weights represent the communication time. In social networks, edges may mean the relationships between different people. In quote networks, an edge means the quotation between two papers or two authors. In these networks, weights of edges are meaningless. However, in IoT networks the edge weights can represent the communication frequencies or communication times which may indicate much latent information. However, a formal definition of structural entropy of weighted graph has not been given.
3) **Clusters.** In IoT networks, different devices may belong to different categories or device types. Our objective is to hide the category information of these devices. As a result, we need to represent the structural entropy of

a network with a given clustering method. Fortunately, we can apply structural entropy with a partition into the clusters to represent local structural information of a cluster. Though these nodes may belong to a same cluster, they may not be locally nearby.

### B. Structral Entropy of weighted and undirected graphs

To solve the problem of direction, we can first calculate the structural entropy only consdering the edges out of nodes when calculate the weight sum of a node. Then we get a out-structural entropy. Similarly, we can calculate the in-structural entropy and their sum will be the structural entropy.

To solve the problem of edge weights, in 1-dimension information entropy of a graphs, a classic way is to convert degrees of each node into the weights of each node [13], [14]. As a result, we can adopt similar policy here.

**Definition 2** *Structure entropy of a weighted and directed graph by clusters. Let $G = (V, E, W, N)$ be a weighted and directed graph with $M$ clusters ($\{X_1, \cdots, X_m\}$). $N$ denotes the cluster of each node where $n_i \in M$. The structural information of $G$ by $M$ under out-edges is as follows:*

$$\begin{aligned} H_M(G) &= \sum_{j=1}^{m} \frac{V_j}{\Gamma} H\left(\frac{w_1^{(j)}}{V_j}, \cdots, \frac{w_{n_j}^{(j)}}{V_j}\right) - \sum_{j=1}^{m} \frac{g_j}{\Gamma} \log_2 \frac{V_j}{\Gamma} \\ &= -\sum_{j=1}^{m} \frac{Vj}{\Gamma} \sum_{i=1}^{n_j} \frac{w_{n_j}^{(j)}}{V_j} \log_2 \frac{w_{n_j}^{(j)}}{V_j} - \sum_{j=1}^{m} \frac{g_j}{\Gamma} \log_2 \frac{V_j}{\Gamma} \end{aligned} \tag{3}$$

*where $V_j$ is the sum of $\{\forall w_{ij} | i \in X_j\}$, $\Gamma$ is the sum of $W$, $w_i^{(j)}$ is the sum of weights of out edges of the $i$th node in $X_j$, $g_j$ is the weights of inter edges between $X_j$ and other clusters where $g_j$ is the sum of $\{\forall w_{ij} | i \in X_j \text{ and } j \notin X_j\}$*

### C. Safety Index

To bridge the gap between the network and the safety analysis of the network, we can use safety index. Similar to the idea of adopting value net in reinforcement learning which allows the system to evaluate the performance of current action but not run until the final reward is generated, we can use a safety index to reflect the safety of the networks. In REM [15], the authors use structal entropy to express the amount of information revealed by a community structure. In Resistor Graphs [16], the authors use structure entropy to measure the force of the graph to resist cascading failure of strategic virus attacks. As a result, it is a straight idea to apply structure entropy to evluate the ability of a network to defend identification attacks.

The problem of anti-device identification is actually very similar to de-anonymization problem in social networks. Method based on graph homomorphism and isomoprphsim is a more common way in such problems. Basic idea behind such methods and structure entropy is that, a more symmetric network is more robust to cyber attacks. However, the anoymization problem based on features suffers the same problem as directly using identification methods. While adopting the unseeded method, to minimize the adacency disagrrement $\Delta_\pi$ is $\Omega\left(\frac{\log n}{n}\right)$. This is an efficient method, but suffers from another problem. In IoT networks, the only modification allowed is adding communications, while deleting or changin edges are not allowded. Because once two devices have established communication, the record shall already be captured. As a result, we can only add edges on current graph, which makes such method unfeasible.

Nevertheless, the safety index based on structural entropy can give a quick result in $O(1)$ cost each time an edge is added. So we give the definition of the safety index of a IoT network.

**Definition 3** *Structure entropy of a weighted and directed graph by clusters. Let $G = (V, E, W, N)$ be a weighted and directed graph with $M$ clusters*

$$S(G) = H_M(G)^{in} + H_M(G)^{out} \tag{4}$$

With the definition of entropy, we know the higher the entropy, the more uncertrainty. As a result, **higher** safety index reflects higher uncertrainty on the view of the attacker, which means the system is **safer**.

### D. Safety Index Analysis

Definition three gives out the formulation of unnormalized safety index of a IoT network, and here we give some analysises over such index.

If the network has only one class, then the second part of the directed and weighted structural entropy (DWSE) becomes 0, and the first part will be $-\sum_{i=1}^{|V|} \frac{w_i}{\Gamma} \log_2 \frac{w_i}{\Gamma}$ and this is the same as 1-dimensional entropy of a weighted graph in [14].

If each node reprsents one class, then the first part of the DWSE becomes 0, and the second part will be $-\sum_{i=1}^{|V|} \frac{w_i}{\Gamma} \log_2 \frac{w_i}{\Gamma}$. The same as only one class. As a result, in two extreme cases, they point to the same explanation of a network.

Then we use a toy example to give deeper insights how DWSE reflects the safety of a network against identifications. As shown in figure 2. The DWSE of graph a) and graph b)

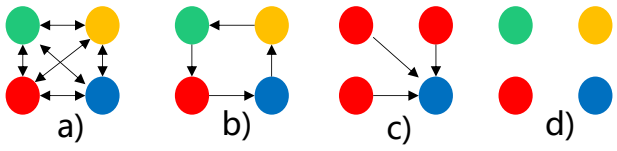


Fig. 2. A top example for better understanding DWSE

is 2+2 and this is the maximum of DWSE. Such network structure is highly symmetric, if the attacker only knows the connections, it cannot read out any information. While for the strcuture in c), the DWSE is 0+1.58, which is smaller and graph structure is very easy to be digged out. Since all three red nodes point to the blue nodes.

## IV. IoT Deception

### A. Deception Method

After the safety index is constructed, we will solve the problem of device deception. The deception means with a given cost $C$, we will use less than $C$ cost to minimize the identification accuracy of the attacker. The formulation is as follows

$$\begin{aligned} &\min_{E'-E} \sum_{vi \in V} \log P_\theta(y_{v_i}|M_{v_i}) \\ &s.t.\ cost(G'-G) \leq C \end{aligned} \tag{5}$$

where $m_{v_i}$ is the ground truth class of nodes and the $y_{v_i}$ is the predicted class by attacker. The $G' = (V, E', W')$ is the confused graph and $G = (V, E, W, N)$ is the original graph. $C$ denotes the given costs.

Actually, if $C > 1$ ,then this is an NP problem. As a result, we use greedy method to approximate the result. With a given $C$, we add one communication to let the DWSE becomes the minimal and repeat such step $C$ times. Here, we suppose the cost of adding any one communication is 1 unit.

In this method, we can save the time cost of identification algorithm. But the time cost is still $O(C|V|^2)$ , not efficient enough. Following the similar idea in REM [15], we can deliberately reduce the cost to $O(C|V|m)$. Alg 1 implements the IoT deceptor. Each time in adding a communication, we first find the start and end point in two arbitrary clusters, then find out the critical edge, which means either the in weight of its start point or out weight of its end point is the smallest. Then, we calculate the DWSE with adding the critical edge in $O(1)$ and we add the critical edge which can make DWSE smallest into the graph and construct new confused graph.

**Algorithm 1** IoT devices' deception over undirected and weighted networks

**Input:** Graph $G(V, E, W, N)$ with $M$ clusters
**Output:** Confusion Graph $G'(V, E, W, N)$

```
G* ← G;
for c ← 1 to C do
    S* ← S(G*);
    for s ← 1 to m and t ← s to m do
        e'_ij ← with least w_i^out or w_j^in in X_s × X_t
        calculate S(G ⊕ e')
        if S(G ⊕ e') > S* then
            S* ← S(G ⊕ e');
            e* ← e'
        end if
    end for
    G* ← G ⊕ e*;
end for
RETURN G*
```

### B. Complexity Analysis

**Theorem 1.** *Alg 1 implements IoT deceptor in $O(Cm|V|)$*

*Proof.* The Alg 1 goes over all $s, t \in M$ clusters and find critical edges. When finding an edges whose start point has the smallest out weights, we only need to calculate the nodes in $X_s$ because the node lays in $X_t$ only affects the in-weights of the nodes. This takes $O(|X_s|)$. Symmetriclly, when finding an edge whose end point has the smallest in weights, it tasks $O(|X_t|)$. For one pair of $s$ and $t$, this takes $O(|X_s + X_t|)$. Thus, for any $X_s$, the algorithm will take $O(m|X_s|+|X_1|+\cdots+|X_m|) = O(m|X_s|+|V|)$. The overall time takes $O(m|X_1| + \cdots + m|X_m| + m|V|) = O(m|V|)$ for adding one communication. So, it takes $O(Cm|V|)$ in all.

## V. Evaluation

### A. Implementation Detail

*1) Dataset:* In this project, three datasets are selected. The detailed graph information is listed in table I. UNSW [17] [1] is a dataset collecting data communications from 28 real household IoT devices. The dataset depicts the networks almost the same as the IoT networks proposed in this project. These devices are parted into three categories: Computer, Monitor and others. The connection incudes wireless connectoin and wired connection.

Apart from evaluating the performance of proposed method on IoT netoworks, we also launch evaluation on other networks. Email-Eu-core network [18], [19] [2] was generated using email data from a large European research institution. The network is a direction graph but unweighted. The weight of each edge is set to 1 initially.

Facebook [20] [3] was collected from survey participants using Facebook app and consists of circles which means communities in the project. The network is an undirected and unweighted graph. So, for each edge, we separate it into one in-edge and one out-edge. The weight of each edge is set to 1 initially.

*2) Identification:* We choose the infomap algorithm [21], [22] as the identification algorithm for it is also an algorithm based on the theory of entropy. The infomap minimize the average number of bits per step by random walks.

$$L(M) = qH(Q) + \sum_{i=1}^{m} p^i H(P^i)$$

to partite the graph into several communities without the label. Since it is also based on the theory of entropy, it can be a good evaluation of how well the deceptor works

*3) Metrics:* Considering the contents represent different networks, we desgin different metrics to better evaluate the deceptor. For UNSW, we use such metric

$$\sum_{i,j \in V} (predict(i) = predict(j)) = (not\ cluster(i) = cluster(j))$$

Because in household IoT networks, devices predicted in the same cluster may not prone to communicatoin while devices from different cluster can have more freqeunt communications. As a result, the class recognized by informap should be

[1] https://iotanalytics.unsw.edu.au/iottraces.html
[2] http://snap.stanford.edu/data/email-Eu-core.html
[3] http://snap.stanford.edu/data/ego-Facebook.html

TABLE I
DETAILED GRAPH INFORMATION OF SELECTED DATASETS

| Dataset | Node # | Edges # | Clusters # | $\Gamma$ | directed | weighted | cost | step |
|---|---|---|---|---|---|---|---|---|
| UNSW | 31 | 39 | 3 | 287707381 | ✓ | ✓ | 5000 | 1000 |
| email | 1005 | 25571 | 42 | 26071 | ✓ | ✗ | 5000 | 1 |
| Facebook | 4039 | 88234 | 7 | 93234 | ✗ | ✗ | 500 | 1 |

TABLE II
PERFORMANCE OF ALG 1 OVER DIFFERENT NETWORKS

| Network | cost (% $\Gamma$) | $S(G)$ | $S(G')$ | $S(G_c)$ | $S(G_R)$ | Metric($G$) | Metric($G'$) |
|---|---|---|---|---|---|---|---|
| UNSW | 2.5 | 1.79 | 1.80 | 9.58 | 1.79 | 0.4688 | 0.4387 |
| email | 1.96 | 13.62 | 15.03 | 23.06 | 21.10 | 0.9264 | 0.8815 |
| Facebook | 5.67 | 17.37 | 17.87 | 19.57 | 17.72 | 0.9962 | 0.6960 |

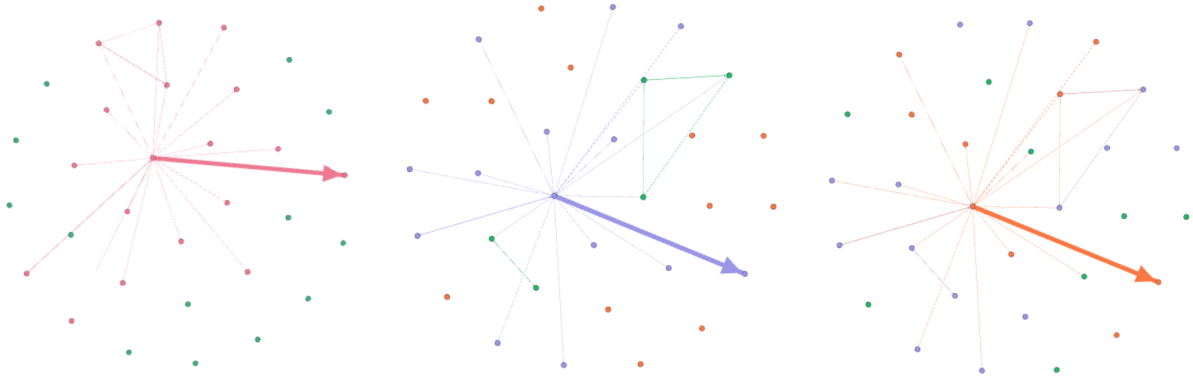

Fig. 3. The identification results before and after Alg. 1 and the ground truth label on UNSW.

corresponded to different ground truth labels. For email and facebook dataset, we use

$$\sum_{i,j\in V}(predict(i)=predict(j))=(cluster(i)=cluster(j))$$

which means predicted classes should be corresponded to true classes.

*4) Code Integrity:* The implementation is based on networkx package. I provide an API which can directly calculate the DWSE of a graph in the .gexf format. The graph file should have cluster information. Link to the API is https://github.com/daxixi/Extended-Structure-Entropy/blob/main/entropy.py.

In the open source project, the following part is provided. The decpetion part implements the algorithm of deceptor to add communications with a cost and constructred the confused graphs. Due to the diversity of dataset, for each dataset, there exist slight differences and the code is included in correspoding folders. The classify part is using the identification algorithm to evaluate the performance of deceptor, for each dataset, the new.gexf and the old.gexf generated by the deceptor part should be provided to evaluate the performace. More detailed references can be found in the open source project.

### B. Performance Evaluation

Table II shows the DWSE of original network and the condused netowrk with particular percent of costs. Also, the identification performace is provided. We can see we only need relatively small part of the original volumes ($\Gamma$) to decept the network. With increasing a little bit of the safety index the deceptor can sucessfully decieve the attacker. Especially over facebook dataset, the deceptor can lead the metric decrease about **0.3** which is a good improvement.

We also compare the result with two typical structures. $G_c$ means adding edges as the network is a complete graph (directed). $G_R$ means randomly assign the classes to the nodes. We can see with deceptor algorithm, we can even increase the safety index over randomly assign classes, which means such structure is more pointed to anti-identification. The $S(G_c)$ is the upper bound of unnormalized DWSE, and in that case there is almost no possibility for attacker to do the identification correctly. For smaller networks, it is harder to increase the strcutral entropy because the network is a little bit limited. But we still can achieve some improvements.

The result shows that the algorithm is effective in deception and it can not only be applied to IoT network where it is originally designed, the safety index can also be applied to other networks such as email network and social network. It

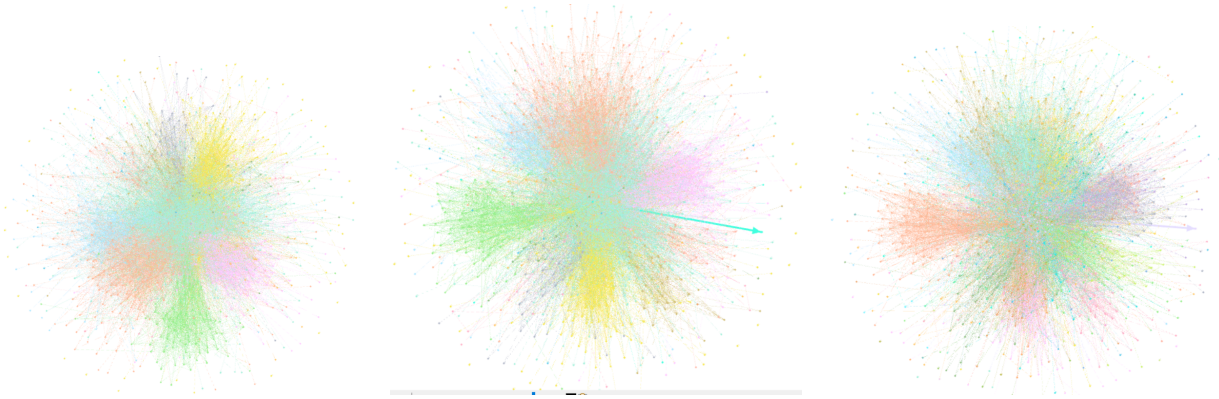

Fig. 4. The identification results before and after Alg. 1 and the ground truth label on email.

Fig. 5. The identification results before and after Alg. 1 and the ground truth label on facebook.

has a good extension.

### C. Identification Visualization

We also give the visualization result of the graph in figures 3 ,4,and 5. The images from the left to the right represent the graph colored with the prediction classes on original graph, the graph colored with the prediction classes on confused graph and the graph colored with the ground truth labels. The layout is generated by gephi and we can see that though the confused graph is very close to the original graph on point of human vision, it can already decepted identification algorithms. Comparing to the ground truth, the identification algorithm fails to give correct label with a given structure.

## VI. Conclusion

Since IoT is becoming more and more popular, the IoT security problem also becomes more and more important. In this project, we find the problem of device identification and propose safety index based on structral entropy. On base of it, we derive the algorithms to do anti device detection. We evaluate the performance of our algorithm over different networks and the evaluation shows that our method can well solve such problem. We can also extend the method to networks besides IoT networks. Finally, we also opensource the code of our project.

## References


[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *IEEE communications surveys & tutorials*, vol. 16, no. 1, pp. 414–454, 2013.

[3] P. Pedarsani and M. Grossglauser, "On the privacy of anonymized networks," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011, pp. 1235–1243.

[4] S. Riis, "Graph entropy, network coding and guessing games," *arXiv preprint arXiv:0711.4175*, 2007.

[5] A. Mowshowitz, "Entropy and the complexity of graphs: I. an index of the relative complexity of a graph," *The bulletin of mathematical biophysics*, vol. 30, no. 1, pp. 175–204, 1968.

[6] D. Bonchev and N. Trinajsti, "Information theory, distance matrix, and molecular branching," *The Journal of Chemical Physics*, vol. 67, no. 10, pp. 4517–4533, 1977.

[7] S. L. Braunstein, S. Ghosh, and S. Severini, "The laplacian of a graph as a density matrix: a basic combinatorial approach to separability of mixed states," *Annals of Combinatorics*, vol. 10, no. 3, pp. 291–317, 2006.

[8] A. Li and Y. Pan, "Structural information and dynamical complexity of networks," *IEEE Transactions on Information Theory*, vol. 62, no. 6, pp. 3290–3339, 2016.

[9] R. Mahmoud, T. Yousuf, F. Aloul, and I. Zualkernan, "Internet of things (iot) security: Current status, challenges and prospective measures," in *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*. IEEE, 2015, pp. 336–341.

[10] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, "Network intrusion detection for iot security based on learning techniques," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019.

[11] M. R. Shahid, G. Blanc, Z. Zhang, and H. Debar, "Iot devices recognition through network traffic analysis," in *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, 2018, pp. 5187–5192.

[12] R. R. Chowdhury, S. Aneja, N. Aneja, and E. Abas, "Network traffic analysis based iot device identification," in *Proceedings of the 2020 the 4th International Conference on Big Data and Internet of Things*, 2020, pp. 79–89.

[13] R. Kazemi, "Entropy of weighted graphs with the degree-based topological indices as weights," *MATCH Commun. Math. Comput. Chem*, vol. 76, pp. 69–80, 2016.

[14] Z. Chen, M. Dehmer, F. Emmert-Streib, and Y. Shi, "Entropy of weighted graphs with randi c weights," *Entropy*, vol. 17, no. 6, pp. 3710–3723, 2015.

[15] Y. Liu, J. Liu, Z. Zhang, L. Zhu, and A. Li, "Rem: From structural entropy to community structure deception," *Advances in Neural Information Processing Systems*, vol. 32, pp. 12 938–12 948, 2019.

[16] A. Li and Y. Pan, "Structure entropy and resistor graphs," *arXiv preprint arXiv:1801.03404*, 2018.

[17] 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 Transactions on Mobile Computing*, vol. 18, no. 8, pp. 1745–1759, 2018.

[18] H. Yin, A. R. Benson, J. Leskovec, and D. F. Gleich, "Local higher-order graph clustering," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 2017, pp. 555–564.

[19] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, pp. 2–es, 2007.

[20] J. J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks." in *NIPS*, vol. 2012. Citeseer, 2012, pp. 548–56.

[21] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.

[22] M. Rosvall, D. Axelsson, and C. T. Bergstrom, "The map equation," *The European Physical Journal Special Topics*, vol. 178, no. 1, pp. 13–23, 2009.