Prediction method of malware infection spreading considering network scale

Yurina Nagasawa*, Keita Kishioka*, Tomotaka Kimura†, and Kouji Hirata‡
* Graduate School of Science and Engineering, Kansai University, Osaka, Japan
E-mail: {k110753, k266175}@kansai-u.ac.jp
† Faculty of Science and Engineering, Doshisha University, Kyoto, Japan
E-mail: tomkimur@mail.doshisha.ac.jp
‡ Faculty of Engineering Science, Kansai University, Osaka, Japan
E-mail: hirata@kansai-u.ac.jp

***Abstract*—In the past, a malware epidemic model based on overlay networks consisting of hosts has been considered. Furthermore, based on the epidemic model, the degree of infection spreading has been estimated through simulation experiments. However, the computation time of the simulation experiment is very long for large-scale networks. To resolve this problem, a prediction method of malware infection spreading using a convolutional neural network (CNN) has been proposed, assuming that the method is applied to fixed-size networks. To extend this work, in this paper, we propose a method to predict the malware spreading with CNN, considering the network scale. The proposed method resizes images without losing information on network structures. By using the resized images as input data to CNN, the proposed method predicts the malware spreading for networks of different scales based on the information on the small networks. Through experimental evaluation, this paper shows the effectiveness of the proposed method.**

I. Introduction

With the rapid growth of the Internet, the evolution of malware such as computer virus, worm, Trojan horse, botnet have become serious threats to the current information society [2], [3], [7]. In [6], an epidemic model of malware considering network structures has been considered. The epidemic model represents infection dynamics on overlay networks, which consist of hosts. The authors have revealed the behavior of malware infection spreading through simulation experiments based on a continuous-time Markov chain. Specifically, the infectious capacity of malware depends on overlay network structures and infection sources. However, the computation time of the simulation experiments considerably increases as the network size becomes large.

In [5], a prediction method of malware infection spreading with the use of a convolutional neural network (CNN) [4] has been proposed in order to resolve the computation time problem. This method prepares gray-scale images made from adjacency matrices, which represents the infectivity between each host pair, as input data for the CNN. The CNN outputs the infectious capacity of malware in a short time. This method, however, assumes that the sizes of networks (i.e. the sizes of images) are fixed in terms of the number of hosts. In particular, the sizes of adjacency matrices for all training data and test data are the same. Therefore, it cannot be directly applied to data sets for networks of different sizes.

To extend the work discussed in [5], in this paper, we propose a method that predicts the infectious capacity of malware with the use of a CNN, accommodating networks of different sizes. The proposed method assumes that the CNN is trained by data sets of small networks. Then it predicts the infectious capacity of malware in larger networks, using the trained CNN. When creating data sets for large-scale networks, the proposed method diminishes the sizes of their adjacency matrices and makes gray-scale images of the same size as the training data sets, without losing information on the network structure. By using the gray-scale images of the diminished adjacency matrices as input data to CNN, the proposed method predicts the infectious capacity of malware in the large-scale networks based on the information on the small size networks. Through experimental evaluation, this paper shows that the proposed method has superior performance over the bi-linear interpolation method, which is a popular image scaling technique.

II. Prediction of malware infection spreading with CNN [5]

A. Epidemic model on overlay networks

In [6], the authors have introduced a malware epidemic model based on a continuous-time Markov chain on an overlay network consisting of hosts. They have assumed a new type of botnets named self-evolving botnets as a target of the epidemic model. The self-evolving botnets discover unknown vulnerabilities by performing distributed machine learning with computing resources of infected hosts. Based on the discovered vulnerabilities, they infect normal hosts, and then make themselves bigger by taking in the newly infected hosts.

In the epidemic model, the state of each host transitions based on a Susceptible-Infected-Recovered-Susceptible (SIRS) model. In the SIRS model, “S” is a state where some vulnerabilities exist in the host (susceptible state). “I” is a state where the host is infected with the botnet malware (infected state). “R” is a state where the host has no known vulnerabilities (recovered state). The host in the susceptible state transitions to the infected state when a host infected with the botnet malware attacks the susceptible host. The epidemic model assumes that the infected host can attack only adjacent hosts on the overlay

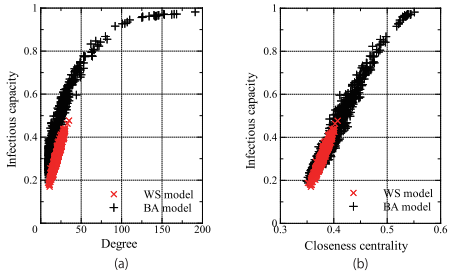

Fig. 1: Infectious capacity against the degree and the closeness centrality of infection sources.

network. On the other hand, it transitions to the recovered state when repairing its own vulnerabilities. The host in the infected state can transition to the recovered state by removing the botnet malware from itself. The host in the recovered state transitions to the susceptible state after the botnet malware discovered a new vulnerability. The botnet malware can infect the host by attacking the vulnerability.

The authors in [6] have conducted simulation experiments based on a continuous-time Markov chain, where the occurrence of each event defined by the SIRS model follows a Poisson process with given transition rates. As overlay networks, two types of networks are constructed by the Watts-Strogatz (WS) model [8] and the Barabasi-Albert (BA) model [1], respectively. They have special characteristics such as the small world property and the scale-free property. It is known that many actual networks have these properties. To evaluate the impact of each host on the overlay networks consisting N hosts, it is assumed that one host is infected and the remaining $N-1$ hosts belong to the susceptible state, as the initial state at time $t=0$. The infected host at time $t=0$ is referred to as the infection source. The number N of hosts is set to 1,000 and the average degree k is set to 20 in each network.

Fig. 1(a) represents the infectious capacity of the botnet malware as a function of the degree of infection sources. The infectious capacity is defined by the probability that there still exist one or more infected hosts after a sufficient amount of time (i.e., stationary state). Also, Fig. 1(b) represents the infectious capacity of the botnet malware as a function of the closeness centrality of infection sources. The closeness centrality of a host is an index indicating the distance from the host to every other host. The high value of the closeness centrality means that the host is located at center of the network. We observe that the infectious capacity of the botnet malware increases with the degree and the closeness centrality of infection sources. We also observe that it depends on the overlay network structures (i.e., the BA and WS models).

B. Prediction method using CNN

The computation time of the simulation experiments considerably increases as the network size becomes large. In order to overcome this issue, in [5], the authors have proposed the prediction method of the infectious capacity of the botnet malware with the use of CNN. Note that the prediction method

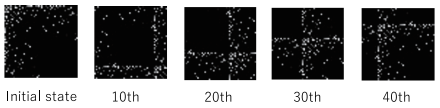

Fig. 2: Example of created images.

can be also applied to other epidemic models in addition to the self-evolving botnet model. The prediction method represents the structure of an overlay network with a weighted adjacency matrix. Let $\mathcal{G}=\{\mathcal{N},\mathcal{L}\}$ denote an overlay network, where $\mathcal{N}$ and $\mathcal{L}$ denote the sets of hosts and links. The overlay network has N $(=|\mathcal{N}|)$ nodes. The weighted adjacency matrix $\boldsymbol{A}_{\mathcal{G}}$ of the overlay network $\mathcal{G}$ is an $N\times N$ square matrix, which is given by

$$\boldsymbol{A}_{\mathcal{G}}=\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,N} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N,1} & a_{N,2} & \cdots & a_{N,N}, \end{pmatrix} \tag{1}$$

where $a_{i,j}$ denotes the connection relation between host i-j. In the prediction method, the infection rate between host i-j is used as the connection relation. The infection rate $a_{i,j}$ is defined as the rate at which host i infects adjacent susceptible host j on the overlay network. Note that $a_{i,j}=0$ if host i is not adjacent to host j on the overlay network.

Based on the weighted adjacency matrix, the prediction method creates an $N\times N$-size gray-scale image that is input to the CNN. Each pixel of the gray-scale image corresponds to the infection rate. Specifically, the value of $a_{i,j}$ is normalized and quantized by $\lceil 255\times a_{i,j}/a_{\max}\rceil$ because the pixel value is an integer in the range from 0 to 255, where $a_{\max}=\max\{\alpha,\{a_{i,j}\mid i,j=1,\ldots,N\}\}$ and α is a parameter. Each pixel value becomes close to 255 as the connection (i.e., infection rate) between the corresponding host pair becomes strong.

The infectious capacity depends on not only the structure of an overlay network but also an infection source. We suppose that host $i=1$ is the infection source in order to distinguish the infection source from other hosts on the overlay network. In particular, the first row and the first column of the weighted adjacency matrix represents the connection relation of the infection source. The image for each host when it is selected as an infection source is created as follows. First, an image is created from the original adjacency matrix. A new adjacency matrix is then made by updating the indices of hosts as $i\leftarrow i-1$ for each host i $(\neq 1)$ and $i\leftarrow N$ for host $i=1$. The new adjacency has a shape in which the previous adjacency matrix is shifted to the upper left. By repeating this process $N-1$ times, N images each of which expresses each host as an infection source are created. Fig. 2 illustrates an example of images created by this process in the case of $N=50$. In this figure, we can see that the pixels are shifted from the lower right to the upper left.

III. Proposed method

A. Image resizing

In the prediction method using CNN, the sizes of images created by the above procedure depend on the network scale (i.e., the number of hosts). Specifically, the size of each gray-scale image made is $N \times N$ pixels. However, the input size of the CNN is fixed. Therefore, images of different sizes which has different network scales should be resized and made the same size when they are input to the CNN. The use of general image resizing methods such as bi-linear interpolation for the prediction method could decrease the prediction accuracy of the CNN because they do not consider the network structures. To resolve this problem, our proposed method aims to enhance the prediction accuracy by resizing images without losing information on the network structures.

Furthermore, it is not desirable to train the CNN with images of large-scale networks. Each training data consists of a gray-scale image and its infectious capacity which is a label calculated by a simulation experiment. We need much time to calculate the infectious capacity for large-scale networks through simulation experiments. Therefore, in the proposed method, we make data sets from small-size networks and train the CNN with the data sets. When predicting the infectious capacity of the botnet malware on a large-scale network, the proposed method creates resized images as follows:

1) For each host $i \in \mathcal{N}$, calculate $S_i = \sum_{j \in \mathcal{A}(i)} a_{i,j}$, where $\mathcal{A}(i)$ denotes the set of hosts adjacent to node i.
2) Select a host i_1 with the smallest S_i.
3) Select a host i_2 that has the smallest S_i from among the hosts adjacent to host h_1.
4) For each host $i \in \mathcal{A}(i_1) \setminus \{i_2\}$, make a link between it and host i_2 if they are not adjacent to each other.
5) For each host $i \in \mathcal{A}(i_1) \setminus \{i_2\}$, the infection rate $a_{i_2,i}$ (=a_{i,i_2}) is updated by
$$a_{i_2,i} \leftarrow a_{i_2,i} + \frac{a_{i_2,i_1} + a_{i_1,i}}{2}.$$
6) Remove host i_1 from the overlay network.
7) Until the number of hosts is equal to the target value, the above steps 1)-6) are repeated. Finally, make a gray-scale images from the resulting weighted adjacency matrix, using the way discussed in Section II-B.

Fig. 3 illustrates an example of the procedure of the proposed method. First, in steps 1) and 2), the host with the smallest $S_i = 0.4$ is selected. In step 3), then, another host adjacent to the selected host with the smallest $S_i = 1.0$ is selected. In step 4), a link between the host selected in step 3) and a host adjacent to the host selected in step 2). The infection rate is updated by $(0.3 + 0.1)/2 = 0.2$ in step 5). In this step, we use the average of the infection rates because we consider the strength of the connection through the removed host. The host selected in the step 2) is removed from the overlay network in step 6). By repeating this procedure, the proposed method can decrease the number of hosts without losing information on the overlay network structure, and create a resized image from the resulting overlay network. Fig. 4 shows an example of a resized image.

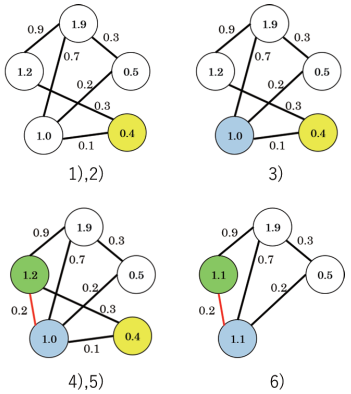

Fig. 3: Proposed method.

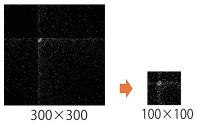

Fig. 4: Resized image.

B. Training and prediction

The proposed method prepares a training data set by means of the following procedure.

1) Make an overlay network $\mathcal{G} = \{\mathcal{N}, \mathcal{L}\}$ consisting of a relatively small number of hosts, where $N = |\mathcal{N}|$.
2) Let $\boldsymbol{A}_{\mathcal{G}}^{[i]}$ denote an $N \times N$ weighted adjacency matrix assuming that host i is the infection source on the overlay network. For each infection source $i \in \mathcal{N}$, calculate the infectious capacity $C(\boldsymbol{A}_{\mathcal{G}}^{[i]})$ of the botnet malware, which is a label, through simulation experiments.
3) For each infection source $i \in \mathcal{N}$, convert the weighted adjacency matrix of the overlay network into a gray-scale image $I(\boldsymbol{A}_{\mathcal{G}}^{[i]})$.
4) Obtain training data $(I(\boldsymbol{A}_{\mathcal{G}}^{[i]}), C(\boldsymbol{A}_{\mathcal{G}}^{[i]}))$ for each infection source $i \in \mathcal{N}$.

By repeating the above procedure, a training data set $\{(I(\boldsymbol{A}_{\mathcal{G}}^{[i]}), C(\boldsymbol{A}_{\mathcal{G}}^{[i]}))\}$ for a sufficiently large number of networks with different structures but the same size is prepared. The CNN is trained by the training data set.

The proposed method predicts the infectious capacity of the botnet malware on large-scale overlay networks, using the trained CNN. The prediction process for an overlay network is as follows.

1) Make an overlay network $\mathcal{G}^* = \{\mathcal{N}^*, \mathcal{L}^*\}$ whose size is equal to or larger than the networks of the training data set.
2) For each infection source $i \in \mathcal{N}^*$ on the overlay network, create a resized image $I(\boldsymbol{A}_{\mathcal{G}^*}^{*[i]})$ by means of the procedure discussed in Section III-A, where $\boldsymbol{A}_{\mathcal{G}^*}^{*[i]}$

TABLE I: Running time of simulation experiments.

# of hosts	100	200	300
Time	00:30:22	03:33:57	9:51:17

TABLE II: Time consumed for prediction by CNN.

# of hosts	100	200		300	
Method	N/A	①	②	①	②
Time	25.9 sec	23.4 sec	24.1 sec	24.3 sec	24.1 sec

①Bi-linear ②Proposed method

denotes a resulting $N \times N$ weighted adjacency matrix made from $\boldsymbol{A}^{[i]}_{\mathcal{G}^*}$ in the procedure.

3) Input the created images $\{I(\boldsymbol{A}^{*[i]}_{\mathcal{G}^*}); i \in \mathcal{N}^*\}$ to the CNN as a test data set.
4) Obtain the estimated infectious capacity $\{E(\boldsymbol{A}^{*[i]}_{\mathcal{G}^*}); i \in \mathcal{N}^*\}$ as output of the CNN.

We can calculate the prediction accuracy by comparing $\{E(\boldsymbol{A}^{*[i]}_{\mathcal{G}^*}); i \in \mathcal{N}^*\}$ with the correct infectious capacity $\{C(\boldsymbol{A}^{[i]}_{\mathcal{G}^*}); i \in \mathcal{N}^*\}$ obtained from simulation experiments.

IV. Performance evaluation

A. Model

We examine the prediction performance of the proposed method through performance evaluation using CNN trained by results from simulation experiments. The CNN consists of three pairs of a convolutional layer and a pooling layer. As an activation function in each convolutional layer, we use the ReLU function. We then use a fully connected layer to estimate the infectious capacity, which is the output of the CNN. In this paper, we prepare a training data set and test data sets for overlay networks with the average degree $k = 4, 6, 8, 10$. The overlay networks are constructed based on the WS model and the BA model. The infection rate $a_{i,j}$ between each host pair is randomly selected from among [0,1] and the parameter α is set to 1.0. We prepare 19,200 images as the training data set, which are equally created for each average degree k, and the size of each image is 100×100 pixels (i.e., $N = 100$). As test data sets, we prepare 4,800 images for different overlay networks with $N = 100, 200, 300$ each. Note that the images are resized to 100×100 pixels by the proposed method. For the sake of comparison, we use the bi-linear interpolation. The correct results about the infectious capacity are calculated from 100 samples obtained in the simulation experiments.

B. Result

Table I shows the total running time in the simulation experiments to calculate the infectious capacity for the 4,800 images of each test data set. Also, Table II shows the total time spent predicting the infectious capacity for the 4,800 images by the CNN. As we can see from these results, the total running time in the simulation experiments greatly increases as the number of hosts increases. On the other hand, the CNN can predict the infectious capacity in short time independent of the number of hosts.

TABLE III: Prediction results.

# of hosts	100	200		300	
Method	N/A	①	②	①	②
Mean error	0.090	0.236	0.143	0.238	0.144
Dev.	0.109	0.277	0.174	0.307	0.183

①Bi-linear ②Proposed method

Table III shows the mean absolute error and the standard deviation. The mean absolute error (MAE) is given by

$$\mathrm{MAE} = \frac{1}{M} \sum_{m=1}^{M} |C_m - E_m|,$$

where C_m denotes the correct infectious capacity for mth image obtained from simulation experiments, which corresponds to $\{C(\boldsymbol{A}^{[i]}_{\mathcal{G}^*}); i \in \mathcal{N}^*\}$, and E_m denotes the infectious capacity for mth image estimated by the CNN, which corresponds to $\{E(\boldsymbol{A}^{*[i]}_{\mathcal{G}^*}); i \in \mathcal{N}^*\}$. The small value of MAE indicates that the CNN accurately predict the infectious capacity. From this table, we observe that the proposed method can improve MAE, compared with the bi-linear interpolation, regardless of the number of hosts. This result implies that the proposed method can create small-size images while keeping information on network structures.

V. Conclusion

This paper proposed a prediction method of malware infection spreading considering the network scale, using CNN. The proposed method resizes images without losing information on network structures. The images are used as input data to CNN. The proposed method predicts the malware spreading for networks of different scales based on the information on the small networks. Through experimental evaluation, this paper showed the effectiveness of the proposed method.

Acknowledgement This research was partially supported by JSPS KAKENHI Grant No. 20H04184.

References

[1] A. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, pp.509–512, 1999.
[2] J. Borello and L. Me, "Code obfuscation techniques for metamorphic viruses," *Journal in Computer Virology*, vol. 4, no. 3, pp. 211–220, 2008.
[3] A. Cani, M. Gaudesi, E. Sanchez, G. Squillero, and A. Tonda, "Towards automated malware creation: code generation and code integration," in *Proc. Symposium on Applied Computing*, Gyeongju, Korea, Mar. 2014.
[4] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, and B. Shuai, "Recent advances in convolutional neural networks," arXiv preprint arXiv:1512.07108, 2015.
[5] K. Kishioka, K. Hongyo, T. Kimura, T. Kudo, Y. Inoue, and K. Hirata, "Prediction method of infection spreading with CNN for Self-evolving Botnets," in *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC 2018)*, Honolulu, HI, Nov. 2018.
[6] T. Kudo, T. Kimura, Y. Inoue, H. Aman, and K. Hirata, "Stochastic modeling of self-evolving botnets with vulnerability discovery," *Computer Communications*, vol. 124, pp. 101-110, 2018.
[7] S. Noreen, S. Murtaza, M. Z. Shafiq, and M. Farooq, "Evolvable Malware," in *Proc. Genetic and Evolutionary Computation Conference*, Montreal, Canada, Jul. 2009.
[8] D. Watts and S. Strogatz, "Collective dynamics of 'small world' networks," *Nature*, vol. 393, pp.440–442, 1998.