URLDeep: Continuous Prediction of Malicious URL with Dynamic Deep Learning in Social Networks

Putra Wanda[1,2] and Huang Jin Jie[1]

(Corresponding author: Huang Jin Jie)

School of Computer Science & Technology, Harbin University of Science and Technology[1]
Xuefu Road No.52, Harbin, Heilongjiang 150080, China
University of Respati Yogyakarta, Indonesia[2]
(Email: wpwawan@gmail.com, huangjinjie163@163.com)

(Received May 20, 2018; Revised and Accepted Aug. 18, 2018; First Online Mar. 2, 2019)

Abstract

Malicious URLs are one of the cybercrimes techniques in digital services. They spread unsolicited content and attack victims. Attacking will make them victims suffer from scams activities such as theft of money, compromise identity, install malware and so forth. In this paper, we propose URLDeep, a deep architecture by novel CNN. Instead of using conventional CNN, we use Dynamic CNN. It can assign a similar signal in the same CNN channel. URLDeep's graph is dynamically updated after each layer of the network. Demonstrated by the experiments, the results of classification accuracy have produced useful accuracy. URLDeep is a novel deep learning framework to learn a nonlinear URL address.

Keywords: Dynamic CNN; Malicious URL; Prediction; Social Networks

1 Introduction

Social networks application have grown tremendously. It has a significant impact on the youth generation lifestyle [1]. It has built a digital environment become part of activities in our daily life. Conventional security technique like cryptography is no longer suitable for a dynamic environment or even common authentication [3,5,18]. So, dynamic environment in social networks such as peer to peer social networks [19] needs efficient security technique.

The machine learning approaches have shown successful performance, particularly when implementing a new concept called Convolutional Neural Networks (CNN) [9] is one of Deep learning architecture and are becoming increasingly popular in the current research. CNN's produced a promising performance for text classification [21]. CNN is different from other neural networks because of convolutional layer concept. Output feature in CNN convolves the previous feature maps with a set of filters in Deep Network. In this study, we designated a novel model of CNN's to learn Malicious URL feature for detecting suspicious address. However, instead of using conventional CNN which uses a static layer in the convolution process, we propose dynamic layer computation in convolutional process.

URLDeep, the underlying deep neural network is a Dynamic Convolutional Neural Network (CNN). In URLDeep, continuous graph updates beneficial to recompute the graph using the k-max concept. Therefore, it can better deal with the problem of data noise, alignment, and other data variations. Demonstrated by the experiments, the results of detection accuracy have shown that URLDeep algorithm is handy for Continuous prediction for malicious URL.

We will summarize the main contributions of our work as follows:

1) We present a novel continuous prediction method of malicious URL, particularly in social networks with Dynamic Deep Learning concept. It used to identify feature representation and learned in useful CNN algorithm.

2) We demonstrate URLDeep with the new dynamic graph in CNN. It is different with conventional CNN operations and has produced better accuracy.

3) With benchmark datasets, we show complete analysis and testing of k-max pooling function and attain state-of-the-art performance.

URLDeep is a novel framework to extract and learn URL address. In this method, we implement Dynamic Convolutional Neural Networks to both characters and words of the URL. It is to learn the URL embedding in the framework. The approach can capture more types of semantic information. As long as we know, it was not

possible by the existing models. The model learns a representation of the raw URL string directly without any designed help from the expert.

2 Related Work

The first alphabet of each word in the title of each section must be Capital Letter.

2.1 URL Detection with Machine Learning

A model developed to detect suspicious URL filtering based on reputation users in the networks [2]. Fast feature extraction technique [17], an approach with designed Multi-layer perception [10], and in big data security area, a study presented a model to secure big data with data mining analysis [15]. Behavioral analysis of attackers [8], prediction based on Community Detection [11].

Many researchers have used the Neural Networks algorithm for addressing issues in a computer environment. In a malicious URL case, many research proposed Machine Learning. A study discussed different machine learning techniques and unsupervised learning. It may able to detecting malicious URLs with the semi-supervised system to detect malicious URL or using Active Learning for Malicious URL Detection with Weighted Text-Based Features [16].

Machine learning is widely used for various cases because understand for non-linear relationships and own robust to outliers. However, it still owns weakness for unconstrained data and prone to overfitting in the training process. A widely used algorithm in machine learning such as SVM suffers over-fitting problem from optimizing the parameters to model selection. In SVM kernel models are very sensitive to over-fitting selection criterion in training process. Therefore, the researcher developed a new advanced algorithm like deep learning.

2.2 URL Detection with Deep Learning

Deep learning is a part of machine learning that is a set of algorithms imitate the structure and function of the brain. It operates on raw input signals and automating the process of feature extraction Deep learning is becoming increasingly popular in solving various applications, one of them is the authentication process.

Deep learning, a subfield of machine learning A study proposed, the eXpose neural network, which uses deep learning with convolutional neural network approach, is to detect artifacts like potentially malicious URLs, file paths, named pipes, and registry keys. The model will learn to simultaneously extract features and classify using character-level embedding [7].

Newly study in malicious URL with deep learning, called URLNet, the paper proposed an end-to-end deep learning framework to learn a nonlinear URL embedding for Malicious URL Detection directly from the URL. Mainly, the model implemented conventional CNN to both characters and words of the URL [6].

A recent study securing URL address, a technique using Event De-noising Convolutional Neural Network for Sequence Detection in malicious URL. The paper proposed a system for detecting malicious URL sequences from proxy logs with a low false positive rate [13].ED-CNN is a particular CNN to reduce the adverse effect of benign URLs redirected from compromised websites such as malicious URL sequences.

In this model, instead of using conventional CNN in deep learning as mentioned above, we propose URLDeep, Deep Learning approach based on Dynamic Convolutional Neural Network (D-CNN). The study uses Dynamic CNN model for detecting malicious URL. Different from conventional CNN, URLDeep can assign similar signal parts to the same CNN channel. In the network, the k-max pooling of a point changes from layer to layer.

3 Model

3.1 Conventional CNN

In common CNN model, the filters are the only parameters of the convolutional layer during training. In CNN algorithm, tensors are essential. In CNN, beginning with the input, intermediate representation, and parameters in computing process are all tensors.

The conventional CNN many identical neurons among the layers. In the CNN model, a computational process run large models computation with a little number of parameters. In Conventional Convolution Neural Network, while the layer receives a single input (the feature maps), it will compute feature maps as its output by convolving filters across the feature maps. The parameters of the convolution layer called filters and back-propagation model used to learn during training.

3.2 URLDeep Neural Network

In this section, we describe the process of URLDeep based on dynamic CNN, which is a generalization of the convolution layer. The discussion explores how the two layers differ from one another regarding input, output, the forward pass and backward pass.

URLDeep, a novel framework with dynamic max pooling in CNN concept. URLDeep model can re-compute the graph using determined layer number, assign similar signal parts to the same CNN channel, and it will able to determining the optimal alignment of weights. Therefore, it can better deal with the problem of data noise in URL address and other data variations. In this model, URLDeep implements dynamic max pooling in computing CNN network.

In contrast to the conventional CNN layer, the URLDeep layer receives two inputs within the operation.

The first input is the previous layer of the features maps and the second is the filters. In this study, URLDeep receives a URL string as input and applies dynamic CNNs to both characters and words in computing the URL address.

Character-level CNN will identify unique characters in the dataset and transform each character into a numerical vector. In this phase, a URL sequence is converted to a matrix representation; Then, the convolution process will compute the matrices. The part will identify critical information which contains maliciousness. Word-level CNN will identify unique words in the training dataset with special characters in URL address. The process will get a matrix representation of the URL or sequence of words. Word-level phase processes to identify useful patterns from certain words that appearing together.

Forward Pass:

$$y_j^n = \sum_i k_{ij}^n * x_i^n. \tag{1}$$

In this process, the first network computes the features maps as the input layer to the dynamic CNN. In the Equation (1), $x_i^n$is the i-th input feature map of the sample n and k_{ij}^n is the ij input kernel of the sample n.

Backward Pass:

$$\begin{aligned} \frac{\partial l}{\partial x_i^n} &= \sum_j \left(\frac{\partial l}{\partial y_j^n}\right) * \left(k_{ij}^n\right) \\ \frac{\partial l}{\partial k_{ij}^n} &= \left(\frac{\partial l}{\partial k_j^n}\right) * x_i^n. \end{aligned} \tag{2}$$

The function of partial derivative computes the gradient of l with respect x_i^n. The values of the gradient calculated by the partial derivative function $\frac{\partial l}{\partial x_i^n}$ and passed to the first layer of the network. Partial derivative in Equation (2) computes the l with respect to k_{ij}^n .

In Dynamic CNN, when one model trained, the dynamic assignment approach is adapted to fit the input signal on a segment basis. The dynamic assignment contains two steps; they are Data Partition and Channel Fitting.

Data Partition process, each activity class, owns the trained model. Then, it will forward it into N parts which correspond to the N channels in the D-CNN. The features partitioned into N parts as well.

Channel Fitting, features which are similar to the same model part go to the same channel in the D-CNN. The distance d between the partitioned features and the model part depicts as Equation (3).

$$d = \frac{1}{n}\sum_{t=1}^{n} d_t, \tag{3}$$

where n is the size of the partitioned feature and t is acceleration signal at the time.

The N model parts correspond to N channels of CNN. In this model, the channel referred to as one path containing convolution and pooling operations in the CNN. Then, the results of N channels are concatenated to build the feature map. Last, two feature maps of gravity and body combined as input to another convolution and pooling operations to capture the correlation between two features.

4 Our Approach

4.1 General Idea

Nowadays, Deep Learning research is growing tremendously. It used to address the various problem in human life such as Image, Video, Audio, Traffic Management, Internet of Things until Biological Data Processing [4, 12, 20].

The various study has been done for detecting malicious URLs. The most common approach is blacklist filtering. The technique is simple but not scalable, though some enhanced approaches were utilizing fuzzy matching. Other study tried to use machine learning (ML) to extract features from URL strings. Recently, an approach using deep learning (DL) to extract features automatically. It applied a mechanical approach to generate feature vectors from URL strings [14].

Instead of using Common Deep Learning model, the model in this research has proposed a novel framework with dynamic CNN concept; it is to predict malicious URL in Social Networks in real-time. It is to classify a new URL as malicious or benign URL. To implement the concept, we will formulate and compute the problem as a binary classification task. In the research, consider S that represents URL is a set of URL$\{(u_1, y_1) \cdots (u_s, y_s)\}$, where u_1 for $s = 1$. $y_s \in \{-1, +1\}$ denotes URL's label. Malicious URL is $y = +1$ and $y_s = -1$ is the benign URL.

The first step in classification procedure is to get feature representation $u_s \rightarrow x_s$ where $x_s \in R^n$ is the n-dimensional feature for u_s as vector represenation of URL. Then, the second step of the model computes prediction CNN function $f : R^n \rightarrow R$. R is the parameter for score prediction of the URL x. The formula $Y_t = sign(f(x_s))$ is to minimize the mistake amount in data training.

This study proposed a novel method that can reach accurate results with dynamic graph concept. However, it will heighten robustness in detecting risk when compared with conventional detecting approaches. This study uses a dataset consisting of million URL address. We design distinctive classifiers for each modality for the mobile profiling and activity behavior. Notably, the URLDeep model runs the process by obtaining a URL string as input, then uses dynamic CNN concept for computing the URL.

This research, re-compute the graph using dynamic pooling in k-max with nearest neighbors. It will be pos-

sible and beneficial for getting efficient to make a continuous prediction in malicious URL. The model is very different from conventional CNN which is using a fixed input in the layer. It calculates effective operation in k-max pooling along the CNN networks and is a crucial distinction of conventional CNN. This study adopts an unsupervised learning model that detect suspicious URL by merely analyzing unlabeled data with the numerical vector. The unsupervised learning has no teacher; it used to malicious detection in social networks.

4.2 Lexical Feature

In training cases, a raw URL converted to a compatible feature vector $u \to x$ before starting training a prediction model. The machine learning uses to calculate the training process. To achieve an effective result, we choose the feature representation to classify different of the features of the URLs.

In this framework, we will focus on the lexical features approach. It is to obtain the feature representation for the URLs. Lexical features are the technique to obtain feature based on the properties of the URL name or string.

The Lexical features technique is very famous and useful to obtain features information of the URL by calculating string of the URL address. Deep learning is a new technique for training and testing process in various cases include in URL detection cases. This model adopts Lexical features for the first stage to convert raw URL u become a feature vector x. In this process, a URL split into words and characters. The model will identify characters and words by transforming each word w_i becomes a feature. Particular of M feature, u_s will be mapped to a vector $x_s \in R^M$.

4.3 URLDeep Architecture

URLDeep, a Dynamic CNN framework for detecting Malicious URL real-time. This model will compute the dynamic CNN networks with a vector representation of the URLs. CNNs have achieved extraordinary success in various tasks. It can learn the salient features from URL string value automatically.

The URLDeep framework based on deep neural network concept, this is a novel framework with dynamic CNN approach for detecting malicious URLs. It is used to learn structural information about the URL. Particularly for URL based on at both the character-level and word-level. This part will describe the malicious URL detection by dynamic CNN model. The process uses Dynamic CNN for computing and classifying malicious or benign URL by computing matrix representation $u \to x \in R^{L \times m}$.

In the model, a URL sequence u is combinatoin of words or characters and separated by special characters. Firstly, we have to obtain URL's matrix representation $u \to x \in R^{L \times m}$ while instances x consist of neighbors components $x_i, i = 1, \cdots, L$ in the sequence. Componens comprise a word or a character of the URL. Each of component is calculated by $x \in R^m$, is an m-dimensional vector. In this study, firstly we initialize the embedding matrix randomly, and use end-to-end optimization to learn the matrix. The m-dimensional representation is an embedding vector which is produced by an embedding matrix.

Instead of using static max-pooling and fixed graph concept (used in conventional CNN). In this study, we propose dynamic pooling and activation output operation. The concept of dynamic k-max pooling can assign similar signal parts to the same CNN channel. It is a novel approach to achieve authentication prediction in malicious URL issues. it can better deal with the problem of data noise, alignment, and other data variations

In the URLDeep, k max pooling k is a function of the input length and network's depth with the equation:

$$k_l = max(k_{top}[\frac{C-c}{C}V_{cw}]). \quad (4)$$

In the Equation (4), c is several of new convolution layer, and C represents the amount of convolution layer, k_{top} is the pooling parameter with the fixed value, it is the top of the convolutional layer. Then to detect URLs whether the malicious or benign URLs, the function need V as vector value of a character or word URL features.

It is to enhance the accuracy of multi-classifier systems and better deal with the problem of data noise, ill alignment, and other data variations. Growing URL addresses in Social Networks environment need accuracy in computing, it is to yield best real-time in the prediction process. Figure 1 describes the process of continuous prediction for malicious URL detection with URLDeep model. It implements Dynamic CNN concept for building algorithm.

This model has a two-tier of calculating matrix in URL detection. It consists of Character level and Word level. The first tier of this architecture called Character level URLDeep, it is to mapping URL address based on character. This phase will convert any characters in the address become unique values. Representation of URL address in numeric values can produce a particular matrix. The phase produces matrix representations result. URLDeep applies the concept of dynamic graph computation by calculating dynamic k-max pooling and output activation map along layers.

The second tier called Word level URLDeep. It is by mapping various URL address based on the word. In this phase, it converts any URLs address in the address become unique values which own different values among the words. Value of word will produce matrix representation; it will ease the calculation of CNN layer. This process also implements the concept of dynamic graph computation with dynamic k-max pooling and output activation map value o.

The next phase of the URLDeep architecture is processed to get a final score of classification. After calculating of matrix representation process, the architecture of the URLDeep shifts to classification. The process called a

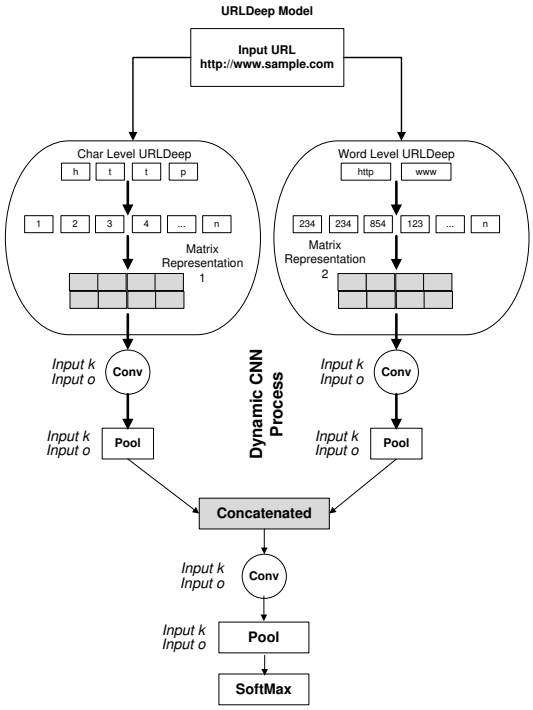

Figure 1: The URLDeep Model with deep learning framework to learn a nonlinear URL address with Characters and Word level. Calculating of matrix representation is to detect Malicious URL directly from the vectors with dynamic k-max pooling k and o is a number of output activation map. It is to learn the URL embedding in the framework

fully connected layer (FC). FC will compute that outputs a vector of K (the number of classes) dimensions in the classification process. The vector owns the probabilities for each class. In the top layer, we use a designated SoftMax function for calculating loss function of the networks to achieve an accurate result.

Character Level URLDeep: This phase presents the main ideas for building Character level URLDeep for Malicious URL Address Detection. It will learn to embed properties about the URL sequence characters. Firstly, in the dataset corpus, We identify all the unique alphanumeric and special characters. To calculate the URL representation into vector value, We set the length of the URL sequence $L_1 = 180$ characters. The phase truncates the characters which the length more than 180.

In the URLDeep process, the character will be embedded into a m ?dimensional vector. In this model, the value of m is dynamic for characters between 16 to 32. During the training process, the embedding value initialized and learned randomly. With the embedding concept, each URL u converts a character into a matrix $u \rightarrow x \in R^{L \times m}$. The formula computes dynamic m value and length of the URL address $L_1 = 180$. This is novel approach for calculating matrix representation in URL address.

Character-level URLDeep can obtain an embedding for new URLs easily in the dataset. The phase changes all the URLs become the URL matrix $x_t \forall t \in R$ as the training data. In this process, it uses dynamic Max-Pooling and followed by fully connected layer. The pooling result will be concatenated with other part of URLDeep.

Word Level URLDeep: In this phase, we identify the unique words in URLs dataset. Unique words depend on the size of data training. In each URL, new words can appear and is different with Character URLDeep with small unique character and fixed value. In this model, we identify unique words with Lexical Feature approachment. All of the unique words are a sequence of alphanumeric characters (including dish character '-' or '_'), length of the URLs number in the model is $L_2 = 180$. All of the unique words make a dictionary for training dataset.

In the next phase, the model gets m-dimensional vector representation. In this study, we use dynamic m value between 16 to 32, that is mean each word comprised to a 16-dimensional vector or 32-dimensional vector. For W unique words, we need to compute a matrix $W \in R^{W \times k}$. When computing with CNN algorithm, the representation of URLs are transformed to matrix representation $L_2 xk$. Word Level URLDeep uses the identic CNN model with Character Level URLDeep.

The last URL matrix representation is the sum of calculation of the above matrices included Word Level URLDeep URL_w with Character Level URLDeep URL_c.

5 Experiment Result

5.1 Large Scale Dataset

In this research, we collected an extensive database of labeled URLs from VirusTotal and PhishTank. VirusTotal service used to validate a URL whether it is malicious or benign. We also collected about 30,000 malicious URLs from PhishTank.

We used a set of benign and malicious URLs for training and testing process. Then we create an annotated database to train and test the URLs. Table 1 depicts URL Dataset corpus for the research.

Table 1 depicts URLs Dataset from VirusTotal and PhishTank.

Table 1: Dataset testing in URLDeep model

	Benign	**Malicious**	**Total**
Training	4,983,425	1.016,575	6,000,000
Testing	9,066,850	933,150	10,000,000
Total	14,050,275	1.049,725	16,000,000

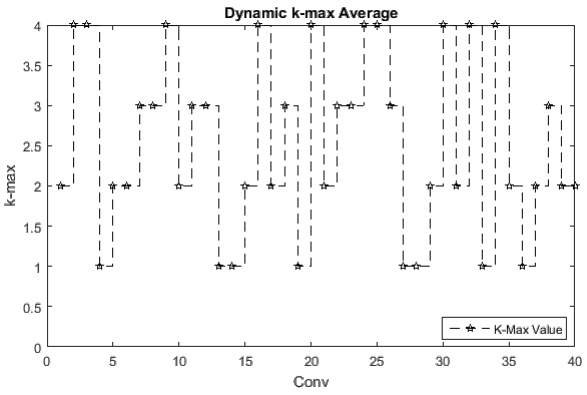

Figure 2: Value of k-max average to compute in the CNN's layer. Dynamic value of activation map will be computed with k value in CNN layer

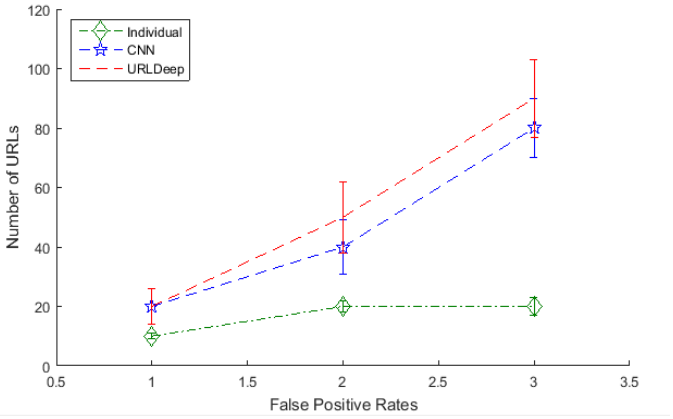

Figure 3: Value of Undetection rate of URL sequences based on FPR calculation

5.2 Analysis

In this study, we compute the character level of URL and word level of URL for detecting malicious URL in a social network environment. Run several training processes for URL addresses with dynamic CNN algorithms. The detection process determines whether an URL is a benign or malicious category.

This study produces the optimal result by URLDeep model-based dynamic CNN architecture. We see dynamic CNN by k-max pooling operation resulted in good accuracy to classify whether the URL address benign or malicious URL. Besides, it is a novel approach for securing social networks when accessing new address over the network. We find a significant increase in CNN graph's performance within dynamic k-max pooling operation for training data and classify URL address.

This model implements dynamic max-pooling of k and activation map value o. Parameter k is kernel parameter used to compute max-pooling layer. Parameter o is total of activation map in the layer. The model applies random activation map o and dynamic value of k-max pooling. Figure 2 shows dynamic k-max pooling value in URLDeep architecture.

This simulation process has used average neuron numbers to train the dataset. It produces relatively high accuracy with k-max pooling operation. In our experiment, more layers and neuron numbers of CNN could not engage to improve the predictive capability. It enlarges resource in the computing process. Therefore a limitation of neuron number and k-max pooling is an effective method to achieve efficient network in dynamic.

Beside of using pooling concept in each layer of neural network, this model has applied Stochastic Gradient Descent (SGD) with Optim. We also implement local variable concept to computing the loss function. To support the local variable concept, the model uses Optim library and Gradparams. SGD based Optim able to calculate the gradient of the loss concerning the weight with various learning rate.

The model train Stochastic Gradient Descent with Optim parameter to train and calculates loss value in each layer of dynamic CNN. It achieves a better result in dynamic CNN network. Algorithm 1 describes an algorithm designed SGD with Optim parameter.

Algorithm 1: Designed of SGD with Optim
Modeling Multi-Layer Perception (MLP)
Determine Input $x?R$ and Hidden Layers HU
Choose ReLU $f(x) = \max(x, 0)$
Determine the size of dataset or batch file
Define Learning rate: LR$>$ 0, $LR?R$
Train SGD with Optim
Print Tensor

In training process, this study various transformed data of the different user become different elements called E. the above graphs show the element $E_1 \cdots E_n$ when running a detection process. Dynamic CNN will analyze data transforming when user interaction in a social network.

We evaluate detection performance results of malicious URL sequences. The URLDeep has implemented general indexes to detection performance called False Positive Rates (FPR), is malicious URL which benign URL categories.

In another hand, the study evaluated the false alerts of malicious URL detection. The URLDeep detects characteristics of malicious redirection with a low FPR. Figure 3 depicts the false alert with FPR indexes.

It is noteworthy that the Undetection rate with URLDeep was significantly high compared to CNN and individual approaches. The URLDeep calculates malicious URL detection based on exploit URLs feature extraction. Besides, CNN approach detects malicious URL sequences based on landing URLs. Based on our experiment, the individual and CNN approach detected several malicious URLs with the sparse result. In contrast, the URLDeep model successfully detected the characteristics of redirections to exploit URLs, and it has produced a

highest Undetection rate of URL sequences based on FPR calculation.

6 Conclusions

The digital environment needs real-time and adaptive security model. Common security model as cryptography is no longer suitable for securing the digital environment. Nowadays, Deep Learning becomes much more popular to overcome various issues. This paper proposes URLDeep, a Dynamic CNN for detecting malicious URLs in the social network. The URLDeep concept can assign similar signal parts to the same CNN channel. Therefore, it can better deal with the problem of data noise, alignment, and other data variations. Demonstrated by the experiments, the results of classification accuracy have produced useful accuracy. URLDeep is used to detect Malicious URL directly from the URL address features. This method can classify malicious or benign URLs in Peer-to-Peer (P2P) Social Network.

Based on the experiment, detection performance results of malicious URL sequences produced by evaluating the false alerts of malicious URL detection. The undetection rate with URLDeep model was significantly higher compared to CNN and individual approaches.

Acknowledgments

This study was supported by the China Scholarship Council under Harbin University of Science and Technology. The authors gratefully acknowledge the anonymous reviewers for their valuable comments. Thank you for all contributors in supporting the research.

References

[1] B. Akashdeep, A. Vinay, G. Sam, "Impact of social networking on Indian youth - A survey," *International Journal of Electronics and Information Engineering (IJEIE'17)*, vol. 7, no. 1, pp. 41-51, Sep. 2017.

[2] C. M. Chen, J. J. Huang, Y. H. Ou, "Efficient suspicious URL filtering based on reputation," *Journal of Information Security and Applications*, vol. 20, no. 26-36, pp. 2214-212, 2015.

[3] C. L. Cheng, Y. Chou, M. S. Hwang, "A new privacy and authentication protocol for end-to-end mobile users," *International Journal of Communication Systems*, vol. 16, pp. 799-808, 2003.

[4] G. Cheng, C. Yang, X. Yao, L. Guo and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 5, pp. 2811-2821, May 2018.

[5] J. Du, C. Jiang, K. C. Chen, Y. Ren and H. V. Poor, "Community-structured evolutionary game for privacy protection in social networks," in *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 3, pp. 574-589, Mar. 2018.

[6] L. Hung, P. Quang, S. Doyen, C. H. H. Steven, "URLNet: Learning a URL representation with deep learning for malicious URL detection," *Cryptography and Security*, 2018. (`https://arxiv.org/abs/1802.03162`)

[7] S. Joshua, K. Berlin, "eXpose: A character-level convolutional neural network with embeddings for detecting malicious URLs, file paths and registry keys," *Cryptography and Security*, 2017.(`https://arxiv.org/abs/1702.08568`)

[8] S. Kim, J. Kim, B. B. H. Kang," Malicious URL protection based on attackers' habitual behavioral analysis," *Computers & Security*, vol. 77, pp. 790-806, 2018.

[9] M. Kumar, H. K. Verma, G. Sikka, "A secure lightweight signature-based authentication for Cloud-IoT crowdsensing environment," *Translations on Emerging Telecommunications Technologies*, 2018. (`https://onlinelibrary.wiley.com/toc/21613915/0/0`)

[10] Z. Li-Xiong *et al.*, "Malicious URL prediction based on community detection," in *International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC'15)*, pp. 1-7, 2015.

[11] Z. Li-Xiong *et al.*, "Malicious URL prediction based on community detection," in *International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC'15)*, pp. 1-7, 2015.

[12] M. Mahmud, M. S. Kaiser, A. Hussain and S. Vassanelli, "Applications of deep learning and reinforcement learning to biological data," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2063-2079, June 2018.

[13] T. Shibahara *et al.*, "Malicious URL sequence detection using event de-noising convolutional neural network," in *IEEE International Conference on Communications (ICC'17)*, pp. 1-7, 2017.

[14] K. Shima *et al.*, "Classification of URL bitstreams using bag of bytes," in *The 21st Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN'18)*, pp. 1-5, 2018.

[15] S. Thakur, E. Meenakshi and A. Priya, "Detection of malicious URLs in big data using RIPPER algorithm," in *The 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT'17)*, pp. 1296-1301, 2017.

[16] F. Vanhoenshoven, G. Nápoles, R. Falcon, K. Vanhoof and M. Köppen, "Detecting malicious URLs using machine learning techniques," *IEEE Symposium Series on Computational Intelligence (SSCI'16)*, pp. 1-8, 2016.

[17] R. Verma, A. Das, "What's in a URL: Fast feature extraction and malicious URL detection," in *ResearchGate*, pp. 24, 2017.
[18] P. Wanda, Selo and B. S. Hantono, "Efficient message security based Hyper Elliptic Curve Cryptosystem (HECC) for mobile instant messenger," in *The 1st International Conference on Information Technology, Computer and Electrical Engineering*, pp. 245-249, 2014.
[19] P. Wanda, Selo and B. S. Hantono, "Model of secure P2P mobile instant messaging based on virtual network," in *International Conference on Information Technology Systems and Innovation (ICITSI'14)*, pp. 81-85, 2014.
[20] S. Yao *et al.*, "Deep learning for the internet of things," in *Computer*, vol. 51, no. 5, May 2018.
[21] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in Neural Information Processing Systems*, pp. 649-657, 2015.

Biography

Putra Wanda. He received B.Eng in University of Respati Yogyakarta, Indonesia and M.Eng in Computer Science, Gadjah Mada University, Indonesia. Now, he is a PhD Student in Institute of Research in Information Processing Laboratory, Harbin University of Science and Technology (email: wpwawan@gmail.com).

Huang Jin Jie. He is an Associate Professor and Ph.D. Supervisorr in School of Computer Science, Harbin University of Science and Technology with speciality in Artificial Intelligence, Robotic and Control (email: huangjinjie163@163.com).