DOI: 10.15439/2018F321 ISSN 2300-5963 ACSIS, Vol. 15

Classification of Computer Network Users with Convolutional Neural Networks

Jakub Nowak, Marcin Korytkowski, Rafal Scherer *Member, IEEE*
Computer Vision and Data Mining Lab, Institute of Computational Intelligence, Czestochowa University of Technology,
Al. Armii Krajowej 36, 42-200 Czestochowa, Poland
Email: {jakub.nowak, marcin.korytkowski, rafal.scherer}@iisi.pcz.pl

Abstract—**Automatic detection of abnormal behaviour of computer network users is a desirable and hard to achieve feature. We show that convolutional neural networks can classify users in local computer networks based on features of web pages which were requested by a user (e.g. URL address, URL category, the day of week or time when the web page was visited). We demonstrate our approach on data collected from a firewall over an eight-month period. This network traffic meta-data allowed to achieve satisfactory classification accuracy on unseen, future network traffic data.**

I. Introduction

FOR the past twenty years, the Internet and its utilisation have grown at an explosive rate. Moreover, for several years computer network users have been using various devices, not only personal computers. We also have to manage with many appliances being constantly online and small Internet of Things devices. Efficient computer network intrusion detection and user profiling are substantial for providing computer system security. Along with the proliferation of online devices, we witness more sophisticated security threats. It is possible to enumerate many ways to harm networks, starting from password weakness. Malicious software can be illicitly installed on devices inside the network to cause harm, steal information or to perform large tasks. Another source of weakness can be Bring Your Own Device schemes, where such devices can be infected outside the infrastructure. At last, social engineering can be used to acquire access to the corporate resources and data.

Each network user leaves traces, some of them are generated directly by the user, e.g. on social networks, others are closely related to the computer network mechanisms. Thanks to network traffic-filtering devices, network administrators nowadays have an enormous amount of data related to network traffic at their disposal. One of the ways to ensure security is to block traffic based on the categorisation of websites. Edge devices (e.g. firewalls or routers with firewall function) verify requested URLs based on the global URL databases and their category ultimately deciding whether a user can access a given page. An example of such devices and reputation databases can be PaloAlto with the Brightcloud database. In order to increase the security, high-end firewalls, simultaneously to filtering, log all the traffic passing through them, storing it, e.g. in relational databases, SYSLOG systems, etc. Thus, network operators can verify the actions of individual users. Log analysis is a crucial element of the network security diagnosis. Usually, the log content is analysed after the fact of an attack or a possible error. Registration of logs is also one of the basic requirements of the right to conduct telecommunications activities. It is mandatory for Internet providers to record who and when visited or shared network resources. Depending on the authentication methods used in a given network and the class of security devices, logs contain information from a very general level, e.g. user IP address, time of the event (of page visit), the address of the requested page up to the user's name.

In [1] the authors rely on the classification of users with all data stored in network logs. The aim was to identify users for the purposes of forensic applications. A compelling argument about why to identify users using data from network traffic and not using the IP addresses assigned to them is that people use mobile devices more often and identify less and less with one, single network. They do not limit the data, as in our case, to the URLs themselves. They use the meta-data of the traffic. However, that base only includes 46 users. The disadvantage of the system that uses all the data can be its performance. Using only URLs, we have fewer data to process, which contributes to the higher efficiency of our system. Events can also be detected by distributed MapReduce approaches [2]. The authors of [3] used a logger on each computer, which additionally logged applications, mouse movements, how were the keys pressed. The error was only 7.1 per cent for 21 users. However, the disadvantage of the method is the interference in the user's system and continuous logging of its behaviour in the system. We expect that artificial neural networks can improve the results on the problem presented in the paper [4][5][6]. A promising approach can be using space-time features [7]. A comprehensive surveys are presented in [8] and [9]. As we faced the challenge of processing a significant amount of data, it would be beneficial in the future to utilise some big data processing methods [10].

In the paper we use convolutional neural networks [11], [12] to classify computer network users based on URL requested by their devices.

II. Computer Network Data

This article is based on data collected from a WAN network infrastructure, which is used by residents of four districts in Poland, as well as network users who are employees of the local government offices and their organisational units,

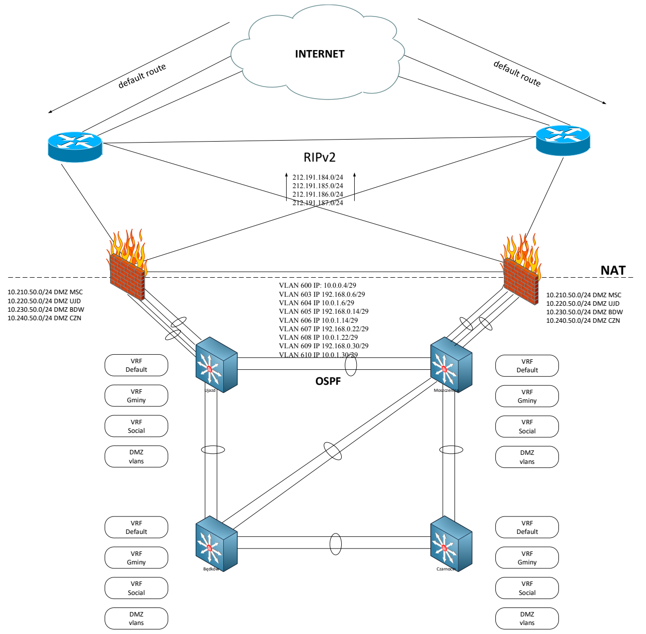
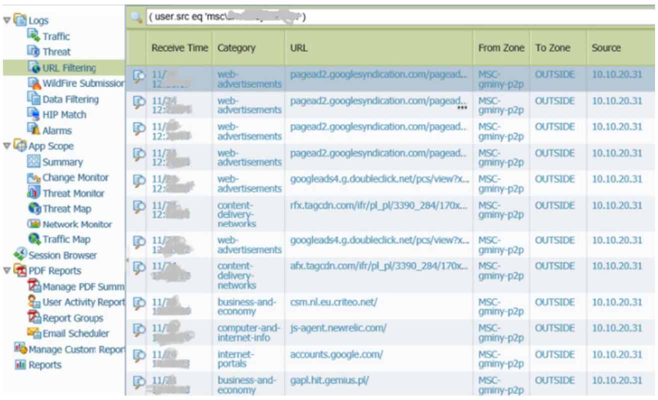


Fig. 1. Schema of the network infrastructure used in the paper to collect traffic data. The Internet is accessed by two routers, and the local network is protected by firewalls.



(user.src eq 'msc\[redacted]')

Receive Time	Category	URL	From Zone	To Zone	Source
11/[redacted]	web-advertisements	pagead2.googlesyndication.com/pagead...	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	web-advertisements	pagead2.googlesyndication.com/pagead...	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	web-advertisements	pagead2.googlesyndication.com/pagead...	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	web-advertisements	pagead2.googlesyndication.com/pagead...	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	web-advertisements	googleads4.g.doubleclick.net/pcs/view?x...	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	content-delivery-networks	rfx.tagcdn.com/ifr/pl_pl/3390_284/170x...	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	web-advertisements	googleads4.g.doubleclick.net/pcs/view?x...	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	content-delivery-networks	afx.tagcdn.com/ifr/pl_pl/3390_284/170x...	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	business-and-economy	csm.nl.eu.criteo.net/	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	computer-and-internet-info	js-agent.newrelic.com/	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	internet-portals	accounts.google.com/	MSC-gminy-p2p	OUTSIDE	10.10.20.31
11/[redacted]	business-and-economy	gapl.hit.gemius.pl/	MSC-gminy-p2p	OUTSIDE	10.10.20.31

Fig. 2. Part of the network log used to create the training data.

e.g. schools, hospitals, etc. Internet access to the analysed network is done with the help of two CISCO ASR edge routers that route packets using RIP version 2. A cluster of PaloAlto devices working in an active-active mode takes care of the network security. The network is routed by the Open Shortest Path First (OSPF) algorithm with virtual routing and forwarding (VRF). In each of the four districts, there is one CISCO core switch. The network is shown in Fig. 1. In the analysed network infrastructure, users are authenticated using accounts in Active Directory services. A RADIUS service is configured for users using the wireless network. PaloAlto devices integrate with the list of accounts contained in domain controllers, thanks to which each user's network traffic is logged using its Active Directory name. The data was acquired in the form of logs from the MS SQL database, then sorted by user names, by the dates when they were recorded, and by the name of the URL address. Sorting by date is designed to reflect the order of websites that were visited. We do not operate on exact dates in this case. It is important to note that additional sorting by name is important. It happened that between two identical URLs registered at exactly the same time a foreign address was requested by another program of that user. If the two URLs next to each other were the same, only one was left. This treatment had a big influence on the results and was able to improve the results.

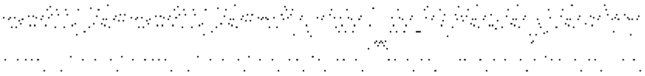


Fig. 3. Example of a part of the input image created from a set URLs.

III. Experiments

A URL is an address that allows locating a website on the Internet. The user encounters it mainly when using a web browser. However, the computer network logs we work on in the paper also contain URLs that have been used by programs running in the background such as antiviruses, system updates, etc. Each user's computer uses different applications and at a different time, which allows to even better distinguish them. As we have observed in our data, certain addresses can be typed in a characteristic way for a given person, e.g. www.google.pl, google.com, www.google.com are three different URLs from our point of view but referring to the same site.

We were inspired here by Zhang et al [12] to present text data in the form of a one-hot vector at the character level. The dictionary consisted of 70 characters:

```
abcdefghijklmnopqrstuvwxvyz_0123456789
−;.!?:/\|#$%&̂˜+=<>()[],"′|ˆ
```

From URL sequences, we created sessions consisted of 8 to 300 URL addresses. The session in the paper is a set of user's URLs where the interval between requests does not exceed 30 minutes. We choose 300 as the maximal value because it was the longest session in the data where there was no 30-minute break.

URLs are concatenated into one string (string), with one URL maximum being 45 characters long. We did not assume a minimum URL length. There was no such need in the collected data. If one URL next to the other was exactly the same, then only one remained. In our experiments, we were only interested in transitions between addresses. The maximum pessimistic length of the training vector is therefore 300 * 45 characters = 13,500.

After creating the input sequences, because only a few vectors had a length of 13,500, we truncated all the sessions to 8014, which allowed to speed up the learning of the network.

The data were collected from June 2017 till February 2018, where the last ten days of February 2018 were used as testing data. The data we collected allowed to divide it into 36,937 sessions, with 1,684,704 for training data, and 9,208 sessions

TABLE I
TOP MOST FREQUENT IP NUMBERS IN THE TRAINING DATA

IP	Count	Domain
212.77.101.148	15918	AS12827 Wirtualna Polska S.A.
172.217.20.174	14846	AS15169 Google LLC
40.77.226.250	13959	AS8075 Microsoft Corporation
172.217.22.14	13137	AS15169 Google LLC
86.111.241.163	10511	elara.iq.pl
185.184.8.30	10118	AS60558 PHOENIX NAP
65.55.44.108	9510	AS8075 Microsoft Corporation
212.77.100.82	8569	AS12827 Wirtualna Polska S.A.
173.241.240.143	8550	AS36089 OPENX TECHNOLOGIES
127.0.0.1	15182	localhost

TABLE II
TOP MOST FREQUENT IP NUMBERS IN THE TESTING DATASET

IP	Count	Domain
217.74.66.216	10639	AS16138 INTERIA.PL Sp z.o.o.
172.217.22.14	10326	AS15169 Google LLC
212.77.101.148	7773	AS12827 Wirtualna Polska S.A.
172.217.23.174	6751	AS15169 Google LLC
40.77.226.250	6126	AS8075 Microsoft Corporation
185.14.253.220	6069	s11.smartsupp.com
185.184.8.30	5005	AS60558 PHOENIX NAP
172.217.22.110	4903	AS15169 Google LLC
172.217.22.3	4636	AS15169 Google LLC
216.58.208.46	4559	AS15169 Google LLC
86.111.241.163	4502	IQ PL Sp. z o.o.

with 788,086 URLs for testing data. The testing dataset was created from the last data in the aforementioned period, thus the testing was performed on the future, unseen URLs. Table I presents top IP numbers present in the training data.

We built convolutional networks [11], [12] and trained them with the backpropagation algorithm [13]. To improve the accuracy we added the linear embedding layer [14] that was also trained from the data. The first network (CNN1), shown in Fig. 4 obtained 33% classification accuracy. It had 32-output linear embedding layer, convolution, max pooling and finally three-layer full-connected network with 62 outputs. The second network (CNN2) is shown in Fig. 5 and obtained 27% classification accuracy. It had the same structure and the only difference was adding additional two input channels encoding time of the week (work days vs. weekends). It allowed improving the accuracy. The next experiment was performed with the same structure with increased the number of outputs in the linear embedding layer to 42 (Fig. 5). It reduced slightly the classification error to 26%. The network training is shown in Figures 7-9.

IV. CONCLUSION

In this article, we proposed a method to classify computer network users based on URLs they have visited. To this end, we encoded URLs as one-hot vectors and presented them as inputs to convolutional neural networks. The obtained results show that the use of additional input data channels with information on users' work days (working days or weekends) resulted in improvement of profiling quality by over 6%. Also, very good effects brought the addition of the embedding layer in the structure of the convolution network. However, increasing the size of this layer does not significantly improve the quality of classification results. The limitation of the presented method is, of course, limited possibility to accurately detect users solely basing on the requested URLs.

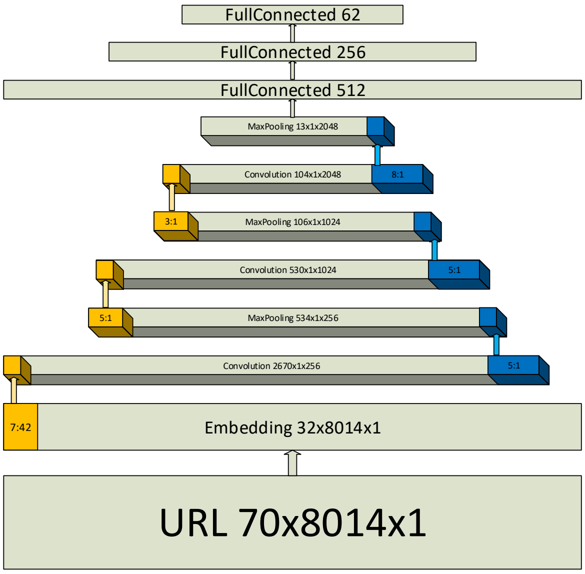


Fig. 4. Convolutional network architecture (CNN1) with 32-output embedding layer without work day information.

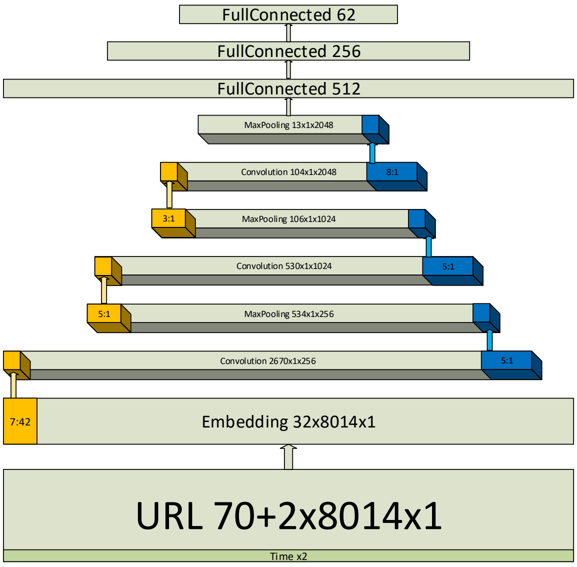


Fig. 5. Convolutional network architecture (CNN2) with 32-output embedding layer with work day information encoded in two last characters.

REFERENCES

[1] N. Clarke, F. Li, and S. Furnell, "A novel privacy preserving user identification approach for network traffic," *Computers & Security*, vol. 70, pp. 335 – 350, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404817301384

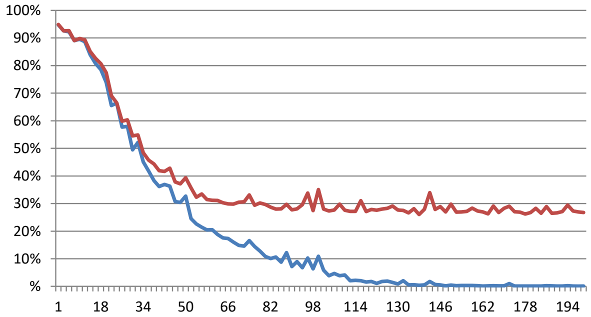


Fig. 9. Convolutional network training (blue line) and validation (red line error (y axis) through epochs (x axis) for CNN3 (Fig. 6).

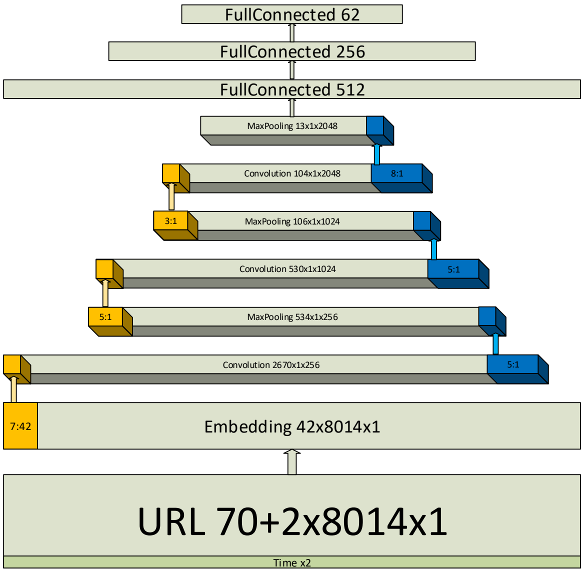


Fig. 6. Convolutional network architecture (CNN3) with 42-output embedding layer with work day information encoded in two last characters.

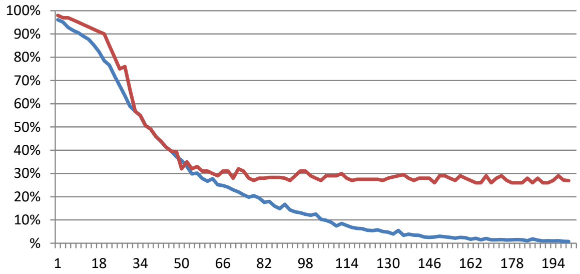


Fig. 7. Convolutional network training (blue line) and validation (red line error (y axis) through epochs (x axis) for CNN1 (Fig. 4).

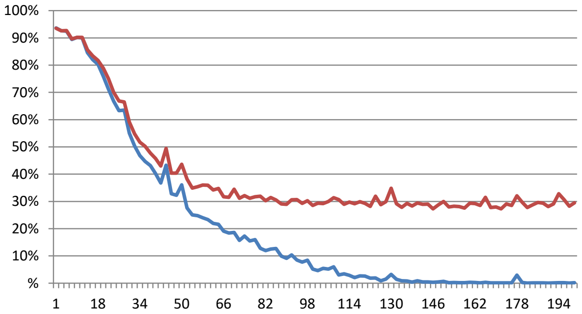


Fig. 8. Convolutional network training (blue line) and validation (red line) error (y axis) through epochs (x axis) for CNN2 (Fig. 5).

[2] P. Yan, “Mapreduce and semantics enabled event detection using social media,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 7, no. 3, pp. 201–213, 2017.

[3] A. Aupy and N. Clarke, “User authentication by service utilisation profiling,” in *Proceedings of the ISOneWorld 2005, Las Vegas, USA*, 2005.

[4] G. Bologna and Y. Hayashi, “Characterization of symbolic rules embedded in deep dimlp networks: a challenge to transparency of deep learning,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 7, no. 4, pp. 265–286, 2017.

[5] Y. Ke and M. Hagiwara, “An english neural network that learns texts, finds hidden knowledge, and answers questions,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 7, no. 4, pp. 229–242, 2017.

[6] T. Minemoto, T. Isokawa, H. Nishimura, and N. Matsui, “Pseudo-orthogonalization of memory patterns for complex-valued and quaternionic associative memories,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 7, no. 4, pp. 257–264, 2017.

[7] O. Chang, P. Constante, A. Gordon, and M. Singana, “A novel deep neural network that uses space-time features for tracking and recognizing a moving object,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 7, no. 2, pp. 125–136, 2017.

[8] A. L. Buczak and E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[9] B. Lee, S. Amaresh, C. Green, and D. Engels, “Comparative study of deep learning models for network intrusion detection,” *SMU Data Science Review*, vol. 1, no. 1, p. 8, 2018.

[10] Z. Marszalek, M. Wozniak, G. Borowik, R. Wazirali, C. Napoli, G. Pappalardo, and E. Tramontana, “Benchmark tests on improved merge for big data processing,” in *2015 Asia-Pacific Conference on Computer Aided System Engineering*, July 2015, pp. 96–101.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[12] X. Zhang, J. Zhao, and Y. LeCun, “Character-level convolutional networks for text classification,” in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS’15. Cambridge, MA, USA: MIT Press, 2015, pp. 649–657. [Online]. Available: http://dl.acm.org/citation.cfm?id=2969239.2969312

[13] D. E. Rumelhart, G. E. Hinton, R. J. Williams *et al.*, “Learning representations by back-propagating errors,” *Cognitive modeling*, vol. 5, no. 3, p. 1, 1988.

[14] A. Conneau, H. Schwenk, Y. Cun, and L. Barrault, “Very deep convolutional networks for text classification,” in *Long Papers - Continued*, vol. 1. Association for Computational Linguistics (ACL), 1 2017, pp. 1107–1116.