Advanced visualization of intrusions in flows by means of Beta-Hebbian Learning

HÉCTOR QUINTIÁN, *Department of Industrial Engineering, University of A Coruña, CTC, CITIC Avda. 19 de febrero s/n, 15405, Ferrol, A Coruña, Spain.*

ESTEBAN JOVE*, *Department of Industrial Engineering, University of A Coruña, CTC, CITIC Avda. 19 de febrero s/n, 15405, Ferrol, A Coruña, Spain.*

JOSÉ-LUIS CASTELEIRO-ROCA, *Department of Industrial Engineering, University of A Coruña, CTC, CITIC Avda. 19 de febrero s/n, 15405, Ferrol, A Coruña, Spain.*

DANIEL URDA, *Grupo de Inteligencia Computacional Aplicada (GICAP), Departamento de Ingeniería Informática, Escuela Politécnica Superior, Universidad de Burgos, Av. Cantabria s/n, 09006, Burgos, Spain.*

ÁNGEL ARROYO, *Grupo de Inteligencia Computacional Aplicada (GICAP), Departamento de Ingeniería Informática, Escuela Politécnica Superior, Universidad de Burgos, Av. Cantabria s/n, 09006, Burgos, Spain.*

JOSÉ LUIS CALVO-ROLLE, *Department of Industrial Engineering, University of A Coruña, CTC, CITIC Avda. 19 de febrero s/n, 15405, Ferrol, A Coruña, Spain.*

ÁLVARO HERRERO, *Grupo de Inteligencia Computacional Aplicada (GICAP), Departamento de Ingeniería Informática, Escuela Politécnica Superior, Universidad de Burgos, Av. Cantabria s/n, 09006, Burgos, Spain.*

EMILIO CORCHADO, *Edificio Departamental, University of Salamanca, Campus Unamuno, 37007 Salamanca, Spain.*

Abstract

Detecting intrusions in large networks is a highly demanding task. In order to reduce the computation demand of analysing every single packet travelling along one of such networks, some years ago flows were proposed as a way of summarizing traffic information. Very few research works have addressed intrusion detection in flows from a visualizations perspective. In order to bridge this gap, the present paper proposes the application of a novel projection method (Beta Hebbian Learning) under this framework. With the aim to validate this method, 8 traffic segments, containing many flows, have been analysed by

*E-mail: esteban.jove@udc.es

Vol. 30, No. 6, © The Author(s) 2022. Published by Oxford University Press.

Advance Access published 16 February 2022 https://doi.org/10.1093/jigpal/jzac013

means of this projection method. The promising results obtained for these segments, extracted from the University of Twente dataset, validate the proposed application.

Keywords: Intrusion detection, traffic flow, exploratory projection pursuit, visualization, artificial neural networks, unsupervised learning

1 Introduction

In a digitized world, the security of information and systems is a major concern. Within this field, intrusion detection (ID) can be defined as the identification of intrusive actions when or after they are performed. The continuous evolution of both technologies and strategies for compromising information systems is one of the main obstacles for ID [11]. To address this challenge, ID Systems (IDSs) were proposed some decades ago, being acknowledged at present time as one of the essential cybersecurity tools. The main target is identifying attempted or ongoing attacks, based on the anomalous detection idea. To process the high volume of data gathered from the traffic travelling along large networks, several alternatives exist. The two main ones are the analysis of the data at packet-level and reducing the data to traffic flows [21]. The latter one is the approach followed in the present study due to the reduced computational demands when compared with the former.

A wide variety of methods have been researched so far to be applied for ID. Among these methods, many of them come from the artificial intelligence (AI) field, while those based on supervised learning [9] are the most popular ones. ID, as well as other cybersecurity subfields such as the detection of malware [27] and web attacks [3], have also been addressed from the visualization perspective based on unsupervised learning. Differentiating from the supervised approach, the visualization one does not try to decide whether a new data instance (a traffic flow in the present study) is ‘normal’ or ‘anomalous’ (i.e. classifying it). The visualization proposal tries to depict all the data in such and intuitive way that the anomalous data can be identified with the naked eye. This is based on the human innate ability of visually identifying anomalous patterns.

Among all the methods in the unsupervised-learning family, exploratory projection pursuit (EPP) is focused in the present paper as it tries to solve the ‘curse of dimensionality’ problem by revealing the hidden structure of a dataset. In order to do it, this method projects the data under analysis onto a low dimensional subspace where the structures can be identified visually. More precisely, the present work proposes Beta Hebbian Learning (BHL), a novel neural projection method, to visualize traffic flows in order to detect the anomalous ones. BHL is compared and validated in this paper when applied to flow-based data; the analysed segments contain flows that have some intrusive instances. These segments were obtained from real-life attacks and are publicly available in the open dataset from the University of Twente [24]. The raw data were gathered from a honeypot directly connected to the Internet, giving for granted that such asset was the target of many attackers.

1.1 Previous work

In the field of cybersecurity, several authors have previously studied the interplay between visualization methods and anomaly detection. Malware detection can be considered as one of the fields where the visualization approach has been widely explored [2]. This is the case of [23], which describes a framework to monitor and visualize anomalous function calls by Android applications. The applied visualization method is a graph on a tree-like structure named dendrograms, using conventional database tables. In [18] GroDDViewer is proposed as a tool that offers two views of the execution of an Android malware. The first of them represents the execution at operating system

level (all the information flow between files, processes and sockets is considered). What happened in the code of the application, during its execution, is visualized in the second one. In [1] the authors propose a visualization-based approach to tackle the problem of investigating large and complex raw data sets from the Internet of Medical Things. Graph oriented data are depicted on a time wise line chart.

In addition to this related work on visualization of Malware, recently iNet [12] has been proposed as a combination of a rare category detection method and visualization techniques. Its main target is to identify and analyse anomalies in multivariate dynamic networks. It integrates two major visualization components, including a glyph-based rare category identifier.

Some other researchers have also investigated the application of unsupervised learning to visualize network data using scatter plots [5, 7, 10, 14, 17]. Differentiating from all these previous papers, the novel method BHL is applied for the first time in the present study. This method has been applied for data ID to different types of cyber-attacks [25–27], obtaining much better results than other well-known algorithms. Furthermore, in [20] it was applied for the first time to ID in traffic flows. Extending this seminal work, the present paper validates BHL when visualizing attacks in a larger and more complex range of traffic segments.

BHL has also been employed to analyse the internal structure of a series of datasets [15, 16], providing a clear projection of the original data. Going one step further, this research proposes the application of BHL to the datasets that have been previously analysed by MOVICAB-IDS [13], to improve the obtained projections and provide a better visual representation of the internal structure of the dataset, in order to easily detect intrusions and other types of cyber-attacks. This facilitates the early identification of anomalous situations, which may be indicative of a cyber-attack in the computer network.

The rest of this paper is organized as follows: section 3 introduces the applied neural techniques while the analysed dataset is described in section 4. Experiments and the obtained results are discussed in section 5 while the main conclusions of this study are presented in section 6, together with some proposals for further research.

2 Unsupervised-learning models for intrusion visualization

The neural EPP methods applied in the present work are described in the following subsection.

2.1 Cooperative maximum likelihood Hebbian learning

Cooperative maximum likelihood Hebbian learning (CMLHL) is a family of rules based on exponential, which extends the likelihood Hebbian learning (MLHL) [16] by adding lateral connections to MLHL network, improving the results obtained by it. CMLHL can be expressed as:

$$Feed-forward : y_i = \sum_{j=1}^{N} W_{ij} x_j, \forall i \tag{1}$$

$$Lateralactivationpassing : y_i(t+1) = [y_i(t) - \tau(b - Ay)^2] \tag{2}$$

$$Feedback : e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i \tag{3}$$

$$Weightupdate : \Delta W_{ij} = \eta \cdot y_i sign(e_j) |e_j|^p \tag{4}$$

where x and y are input (N-dimensional) and output (M-dimensional) vectors, with W_{ij} weight connections between both. And η the learning rate, τ the 'strength' of the lateral connections, b the bias parameter, and p a parameter related to the energy function. Finally, A is a symmetric matrix used to modify the response to the data whose effect is based on the relation between the distances among the output neurons [6].

2.2 Beta Hebbian learning

Artificial neural networks (ANNs) are typically software simulations that emulate some of the features of real neural networks found in the animal brain. Among the range of applications of unsupervised artificial neural networks, data projection or visualization is the one that facilitates, human experts, the analysis of the internal structure of a dataset. This can be achieved by projecting data on a more informative axis or by generating maps that represent the inner structure of datasets. This kind of data visualization can usually be achieved with techniques such as EPP [4, 19], which project the data onto a low dimensional subspace, enabling the expert to search for structures through visual inspection.

The Beta Hebbian Learning technique [31] is an ANN belonging to the family of unsupervised EPP, which uses Beta distribution as part of the weight update process, for the extraction of information from high dimensional datasets by projecting the data onto low dimensional (typically 2 dimensional) subspaces. This technique is better than other exploratory methods in that it provides a clear representation of the internal structure of data.

BHL uses Beta distribution to update its learning rule to match the probability density function (PDF) of the residual (e) with the dataset distribution, where the residual is the difference between input and output feedback through the weights (8). Thus, the optimal cost function can be obtained if the PDF of the residuals is known. Therefore, the residual (e) can be expressed by 5 in terms of Beta distribution parameters ($B(\alpha \; and \; \beta)$):

$$p(e) = e^{\alpha-1}(1-e)^{\beta-1} = (x - Wy)^{\alpha-1}(1 - x + Wy)^{\beta-1} \tag{5}$$

where α and β control the PDF shape of the Beta distribution, e is the residual, x are the inputs of the network, W is the weight matrix and y is the output of the network. Finally, gradient descent can be used to maximize the likelihood of the weights (Eq. 6):

$$\begin{aligned} \frac{\partial p_i}{\partial Wij} = (e_j^{\alpha-2}(1-e_j)^{\beta-2}(-(\alpha-1)(1-e_j) + e_j(\beta-1))) = \\ (e_j^{\alpha-2}(1-e_j)^{\beta-2}(1-\alpha+e_j(\alpha+\beta-2))) \end{aligned} \tag{6}$$

Therefore, BHL architecture can be expressed by means of the following equations:

$$Feed-forward : y_i = \sum_{j=1}^{N} W_{ij}x_j, \forall i \tag{7}$$

$$Feedback : e_j = x_j - \sum_{i=1}^{M} W_{ij}y_i \tag{8}$$

$$Weightupdate : \Delta W_{ij} = \eta(e_j^{\alpha-2}(1-e_j)^{\beta-2}(1-\alpha+e_j(\alpha+\beta-2)))y_i \tag{9}$$

where η is the learning rate

3 Analysed dataset

In the present research, the methods described in section 2 are applied to the benchmark dataset containing traffic flows released by the University of Twente [24]. This is a publicly available and widely used datasets to assess IDS based on flow-based data [8].

More than 155 M packets, travelling along a large academic network, were collected in a 24 GB dump file. During 6 days, traffic addressed to a honeypot connected to the Internet was gathered. The following typical network services were running in the target server:

- Apache web server: just a basic login page was stored in this server.
- ftp: ProFTPd that uses the auth/ident service was chosen for additional authentication information about incoming connections.
- ssh: the OpenSSH service running on Debian was patched to track active hacking activities by logging sessions: for each login, the transcript (user typed commands) and the timing of the session was recorded.

Among the running services, those more frequently addressed by attackers were ssh and http.

In order to ease the analysis, the millions of captured packets were summarized in 14.2 M flows. Based on the captured traffic, the resulting dataset contains several types of flows: ssh-scan, ssh-conn, ftp-scan, ftp-conn, http-scan, http-conn, authident-sideeffect, irc-sideeffect, icmp-sideeffect. Only 6 connections to the ftp service are contained in the dataset. All of them contain data related to an opening of an ftp session, that is immediately closed.

The majority of the attacks targeted the ssh service and they can be divided into two categories:

- Manual: these are manual connection attempts, amounting to 28 in the dataset (among them 20 succeed). Differentiating from the previous ones, it is much more difficult to detect this type of attacks.
- Automated: these unmanned attacks are generated by specific-purpose tools and mainly comprise brute force scans, where a program enumerates usernames and passwords from large dictionary files. As each connection come to a new flow, it is particularly easy to identify such attacks at flow level.

All the http alerts labelled in the dataset are considered as attacks performed by hackers. This is mainly because no http attacks were artificially generated in the dataset. By executing a scripted series of connections, hackers tried to compromise the http service.

The following flow features are used by the EPP methods to detect intrusive actions:

- src-ip: anonymized source IP address (encoded as 32-bit number).
- dst-ip: anonymized destination IP address (encoded as 32-bit number).
- packets: number of packets in the flow.
- octets: number of bytes in the flow.
- start-time: UNIX start time (number of seconds).
- start-msec: start time (milliseconds part).
- end-time: UNIX end time (number of seconds).
- end-msec: end time (milliseconds part).
- src-port: source port number.
- dst-port: destination port number.
- tcp-flags: TCP flags obtained by ORing the TCP flags field of all packets of the flow.
- prot: IP protocol number.

TABLE 1. Information about the analysed segments.

Segment ID	Flows	Attacks
1	12,179	ssh-conn, ftp-conn and irc-sideeffect
30	12,172	ssh-conn, ftp-conn and irc-sideeffect
58	1,214	ssh-conn, http-conn and irc-sideeffect
59	1,216	ssh-conn and irc-sideeffect
107	19,061	ssh-conn, authident-sideeffect, irc-sideeffect and icmp-sideeffect
131	122,274	ssh-conn, http-conn, irc-sideeffect and icmp-sideeffect
211	80,944	authident-sideeffect, irc-sideeffect and icmp-sideeffect
545	731	ssh-conn and http-conn

TABLE 2. CMLHL and BHL parameters for segment 545.

Algorithm	Parameters
CMLHL	iters=5000, lrate=0.01, $p = 1.2$
BHL	iters=5000, lrate=0.001, $\alpha = 4$, $\beta = 3$

Additionally, the alert-type feature (also contained in the dataset) is used for depicting the data and validating the results.

The analysed dataset is partitioned, according to the segmentation strategy initially proposed under the frame of MOVICAB-IDS[13]. As a result, all the flows whose timestamp is between the segment initial and final time limit are contained in such segment. As the total length of the dataset is 539,520 seconds, the segment length has been defined as 782 seconds. There is an overlap between consecutive segments, that is defined as 10 seconds. As a result, 709 segments were generated from the original dataset.

For brevity, results on only few of the generated segments can be included in the present paper. Thus, some of the segments must be selected. The main criteria for that is prioritizing those with the minimum number of flows present with a specific number of attack types. Additionally, these segments have been selected in order to compare the obtained results with those of previous work [22]. Basic information about the studied segments studied is shown in 1.

4 Experiments and results

As previously stated, BHL is applied to the segments described in the previous section. The best projections obtained from such segments are presented in this section. Additionally, they are compared with previous results obtained by CMLHL, as it was the method that provided best visualizations, according to previous research studies.

The date are projected by means of the corresponding EPP method in a scatter plot. Additionally, attack label information is added to the projections, mainly by the glyph metaphor (different colours and symbols).

In all cases for the BHL experiments a normalization of each variable between the range -1 to 1 has been applied to guarantee the stability of the BHL network during the training process [19]. Finally best projections are presented and each type of attack is presented in different colour.

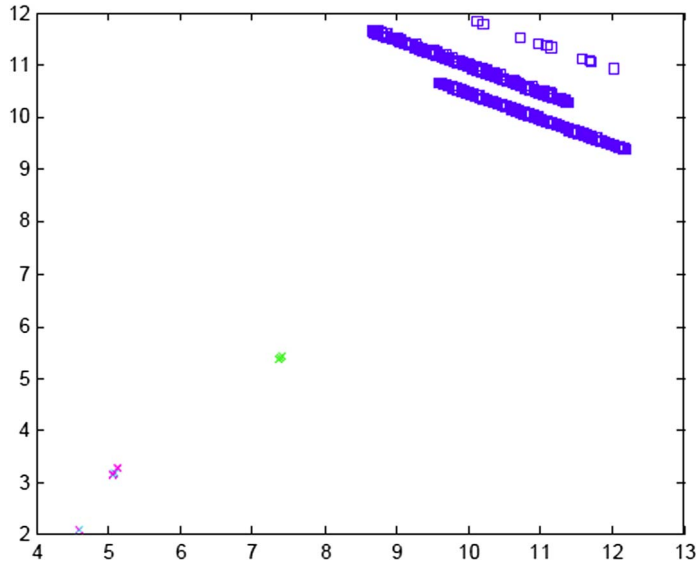

FIGURE 1. CMLHL projection for segment 545.

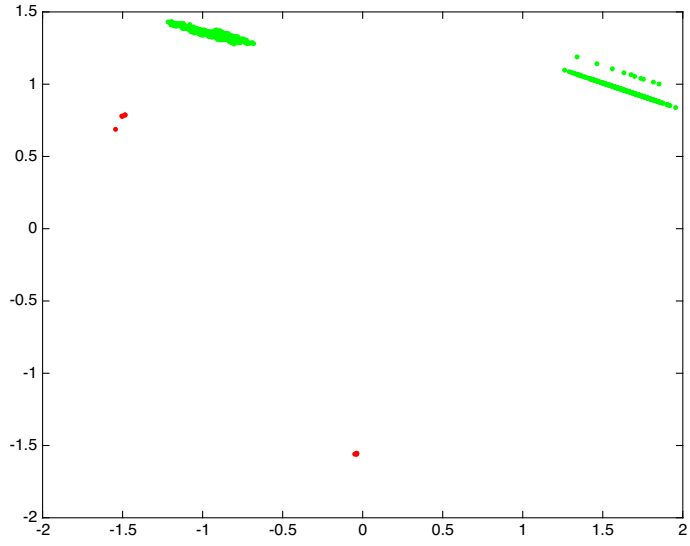

FIGURE 2. BHL projection for segment 545.

4.1 Visualizations of segment 545

This dataset contains 2 kinds of attacks, ssh_conn and http_conn attacks. Dataset consists of 731 samples and 9 variables.

Table 2 shows the best combination of parameters for the obtained projections by BHL and CMLHL when analysing segment 545.

TABLE 3. CMLHL and BHL parameters for segment 30.

Algorithm	Parameters
CMLHL	iters=100000, lrate=0.01, $p = 1.1$
BHL	iters=100000, lrate=0.001, $\alpha = 3$, $\beta = 4$

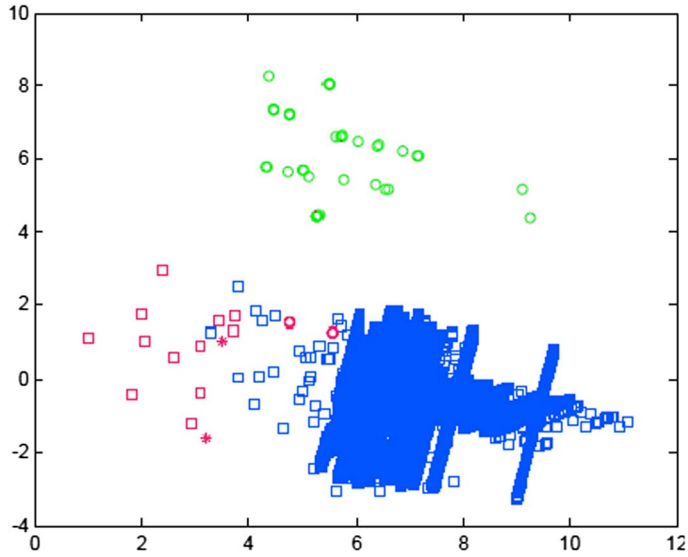

FIGURE 3. CMLHL projection for segment 30.

Figure 1 shows the CMLHL projection. In such visualization, both types of attacks (Category 2 and 6) are clearly differentiated and separated in the scatterplot.

Results obtained by BHL on this same dataset segment are shown in Figure 2. This visualization also shows a clear separation between the 2 types of attacks; however, it is not possible to provide better results than CMLHL as they are good enough and no classes are mixed. The only remarkable difference with respect to the previous CMLHL projection is that BHL separates the first type of attack (green dots in Figure 2), based on the different source IP of each type of attack.

4.2 Visualizations of segment 30

3 different attacks are present in this dataset segment, ssh_conn (category 2), ftp_conn (category 4) and irc_sideeffect (category 8), which represents a total of 121,72 sample and 9 variables.

Table 3 shows the best combination of parameters for the obtained projections in case of BHL and CMLHL for this segment.

In this case, the CMLHL projections present categories 2, 4 and 8 that are mixed as can be seen in the central part of the Figure (3 (blue-circle, asterisk and blue-green squares respectively)). Therefore, it is difficult to differentiate the type of attack in this projection.

However, BHL projections shows a clear separation between all samples of the different type of attacks, represented in Figure 4 as green dots (category 2), red dots (category 4) and blue dots

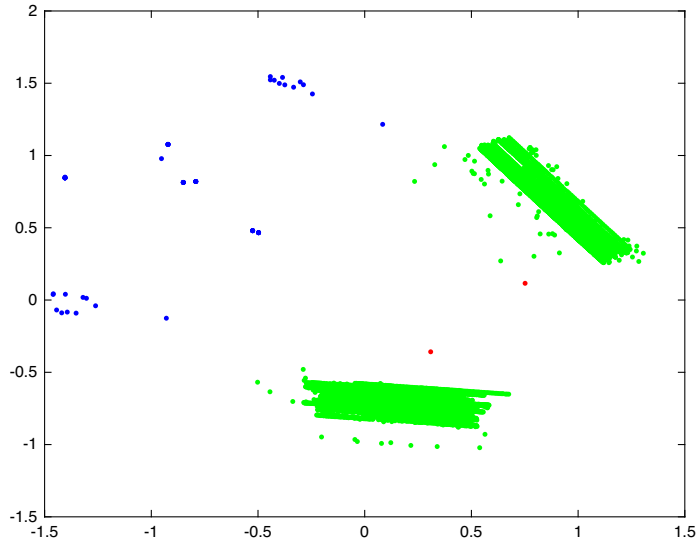

FIGURE 4. BHL projection for segment 30.

TABLE 4. CMLHL and BHL parameters for segment 107.

Algorithm	Parameters
CMLHL	iters=100000, lrate=0.01, $p = 1.16$
BHL	iters=100000, lrate=0.001, $\alpha = 5$, $\beta = 3$

(category 8). Again, as happened in previous dataset category 2 (green dots) is divided in 2 parts corresponding to 2 different source IP.

4.3 Visualizations of segment 107

Dataset segment 107 has a total of 19,061 samples and 9 variables, which correspond to 4 types of attacks (ssh_conn, authident_sideeffect, irc_sideeffect and icmp_sideeffect) labeled as categories 2, 7, 8 and 9 respectively.

Table 4 shows the best combination of parameters for the obtained projections in case of BHL and CMLHL.

Best CMLHL projections are presented in Figure 5, where samples of categories 2 and 8 are mixed. In spite of samples of other categories are not mixed, separation between clusters is quite small, so it is difficult to clearly differentiate the boundaries between clusters.

The best BHL projection is presented in Figure 6, here it can be seen that there is not mixed samples of different clusters, and separation between clusters is greater than in case of CMLHL, specially between samples of category 2 (green dots) and category 8 (blue dots).

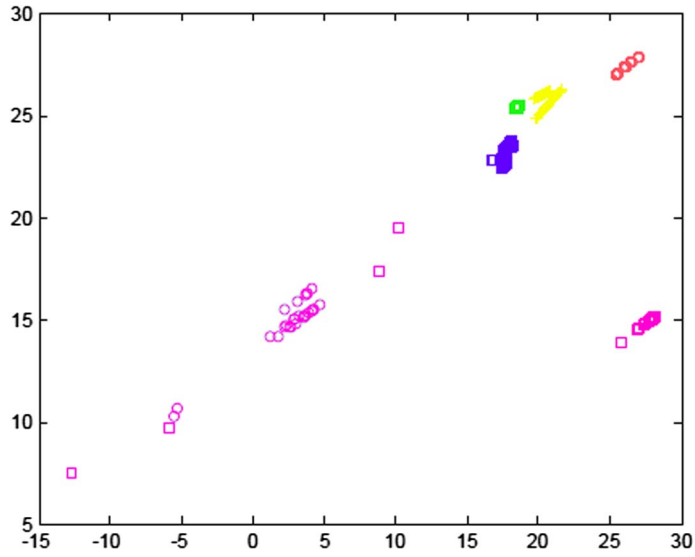

FIGURE 5. CMLHL projection for segment 107.

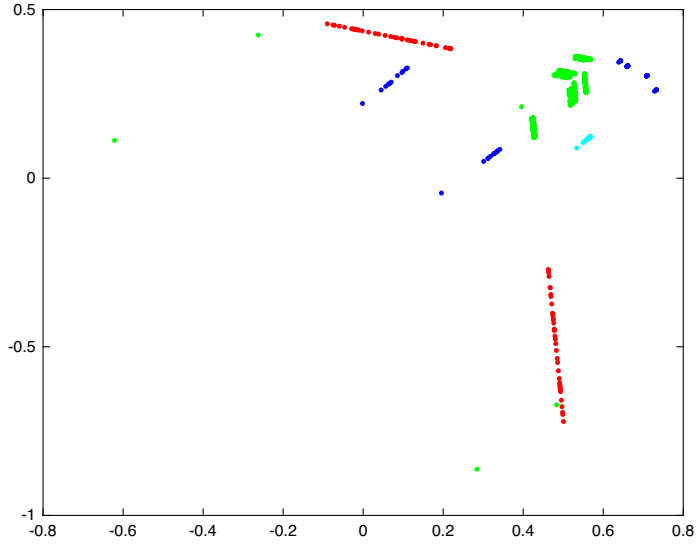

FIGURE 6. BHL projection for segment 107.

4.4 Visualizations of segment 131

Finally, it is here presented dataset segment 131, such dataset has a total of 122,274 samples and same 9 variables as in previous datasets. This dataset contains 4 types of attacks, labeled as category 2 (ssh_conn), 6 (http_conn), 8 (irc_sideeffect) and 9 (icmp_sideeffect).

TABLE 5. CMLHL and BHL parameters for segment 131.

Algorithm	Parameters
CMLHL	iters=1000000, lrate=0.01, $p = 1.1$
BHL	iters=1000000, lrate=0.001, $\alpha = 4$, $\beta = 3$

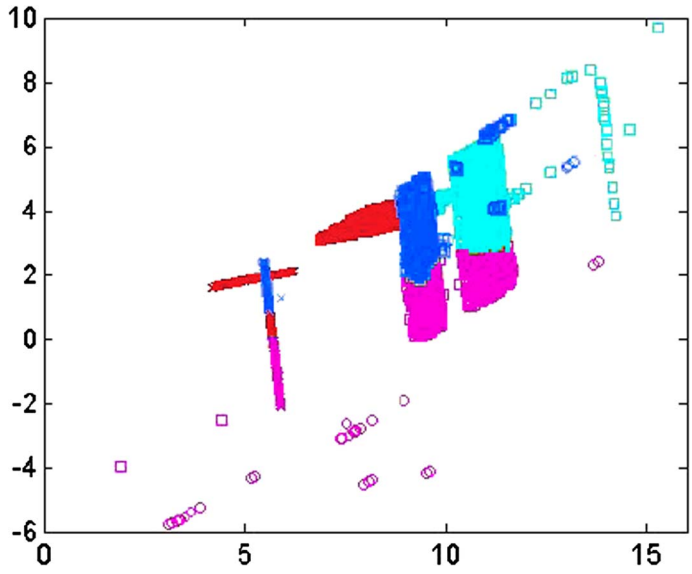

FIGURE 7. CMLHL projection for segment 131.

Table 5 shows the best combination of parameters for the obtained projections in the case of BHL and CMLHL.

The CMLHL projection is presented in Figure 7. Here several samples of different categories are mixed in same clusters as can be appreciated in the legend of the figure. For instance, cluster 1 has samples of category 2 and 6, cluster 2 mix categories 2 and 9, to cluster 3 belongs samples of categories 2 and 6 and finally in case of cluster 4 all categories are mixed. Therefore, such results do not provide a useful projection in order to be used for as ID tool.

However, in the case of projections provided by BHL (see Figure 8), samples of different categories are not mixed, but in the case of the 2D projection (Figure 8), some clusters are near to others. Nevertheless the BHL 3D projection (Figure 9) provides a much better visualization of the clusters, being such clusters more compact and separated from each other. In any case, both projections clearly improve the results provided by the CMLHL algorithm.

4.5 Visualizations of additional segments

Based on the previous results that show how BHL outperforms CMLHL on the analysed segments, complementary results are shown below. In this subsection, the BHL visualizations for 4 additional segments, with different combination of types of attacks, are shown.

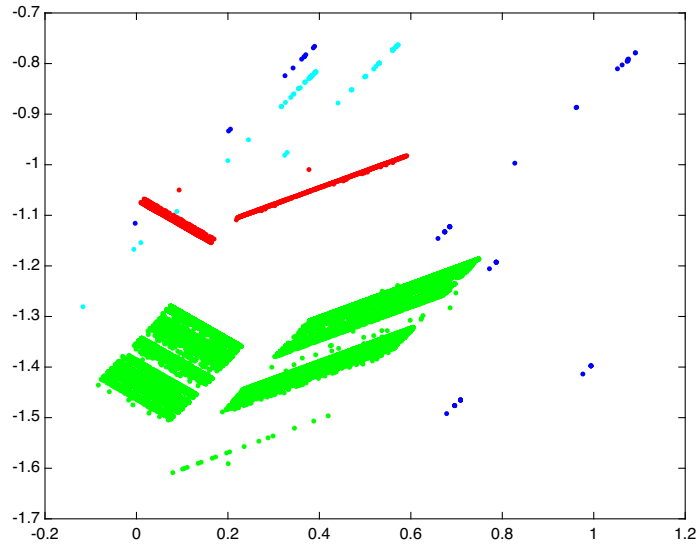

FIGURE 8. BHL 2D projection for segment 131.

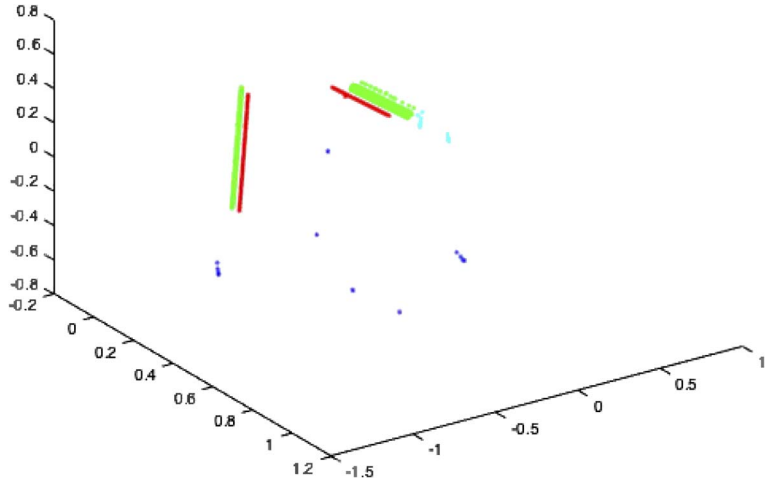

FIGURE 9. BHL 3 first dimensions projection for dataset segment 131.

4.5.1 Visualizations of segment 1 This segment contains 3 types of attacks: ssh-conn, ftp-conn and irc-sideeffect. It comprises a total of 12,179 samples and 9 variables.

After training the BHL algorithm with this segment, the best projection obtained is shown in Figure 10, It can be seen how all attacks are clearly identified being linearly separable.

In Table 6, the BHL parameter values are detailed.

Despite the fact that the dataset is clearly unbalanced (there are much more samples of attack 2, than the others), the BHL can make clear groups for the different attacks.

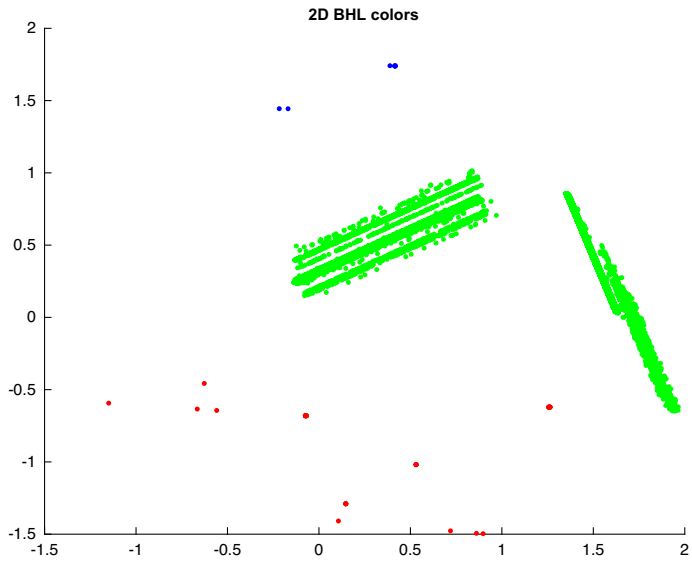

FIGURE 10. BHL 2D projection for segment 1.

TABLE 6. BHL parameters for segment 1.

Algorithm	Parameters
BHL	iters=10000, lrate=0.001, $\alpha = 4$, $\beta = 3$

4.5.2 Visualizations of segment 58 This segment contains 1,214 samples divided in 3 types of attacks: ssh-conn (1,160 samples), http-conn (2 samples) and irc-sideeffect (52 samples). Once again, this dataset is clearly unbalanced and in spite of it, BHL generates a clear separation in its projection, as shown in Figure 11

Table 7 shows the parameters used for training BH for segment 58.

4.5.3 Visualizations of segment 59 In segment 59, there are 2 types of attacks: ssh-conn attack (1160 samples) and irc-sideeffect (56 samples). All in all, there are 1,216 flows and 9 variables.

Once BHL is trained with parameters shown in the table 8, the best 2D projection (components 1 and 4) obtained is presented in Figure 12.

Results for this segments reveal a plain visualization of both types of attacks, with no mixing samples, and linearly separable.

4.5.4 Dataset segment 211 In this final dataset, a total of 4 different types of attacks are present: ssh-conn (80469 samples), authident-sideeffect (2 samples), irc-sideeffect (39 samples) and icmp-sideeffect (434 samples). Again, the types of attacks are clearly unbalanced in the dataset representing 99.416%, 0.002%, 0.0482% and 0.533% respectively.

The best combinations of BHL training parameters are presented in Table 9. The 2D best projection is shown in Figure 13, where it can be seen as some samples of ssh-conn and authident-

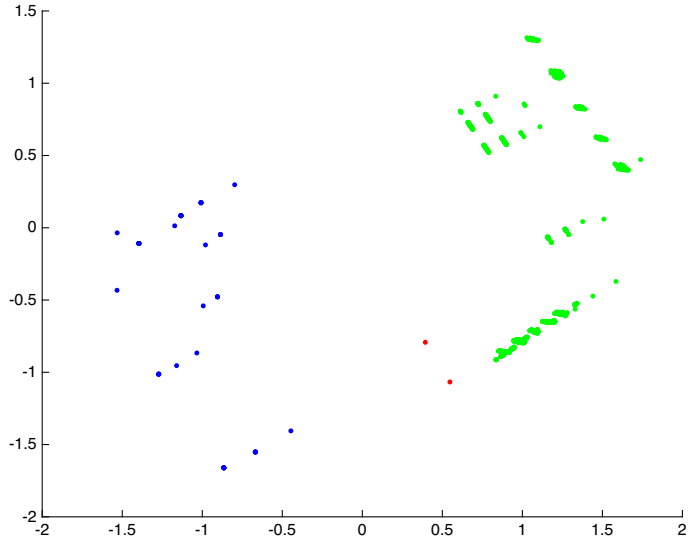

FIGURE 11. BHL 2D projection for segment 58.

TABLE 7. BHL parameters for segment 58.

Algorithm	Parameters
BHL	iters=10000, lrate=0.001, $\alpha = 4$, $\beta = 3$

TABLE 8. BHL parameters for segment 59.

Algorithm	Parameters
BHL	iters=10000, lrate=0.001, $\alpha = 4$, $\beta = 3$

sideeffect cannot be separated. However, if 3D projection is used, it can be obtained a better representation, where there are no mixed samples of any type of attack as it can be seen in Figure 14.

Finally, a [0,1] normalization of the dataset were tested obtaining a better representation, which is shown in Figure 15

5 Conclusions and future work

The detection of intrusions in large networks still is an open challenge. In order to address it, the present paper proposes the novel application of a new EPP technique from the visualization perspective. Hence, this model, based on unsupervised learning, is applied to reduce the dimensionality of flow data in order to generate 2D or 3D visualizations of the data. As a result, the different intrusions can be visually monitored and analysed.

In order to validate the proposed application, 8 different datasets are analysed. They contain a varied set of attacks and a different number of them. The experimental results prove that BHL can be successfully applied for such visualization task. When applied to the selected segments, the BHL

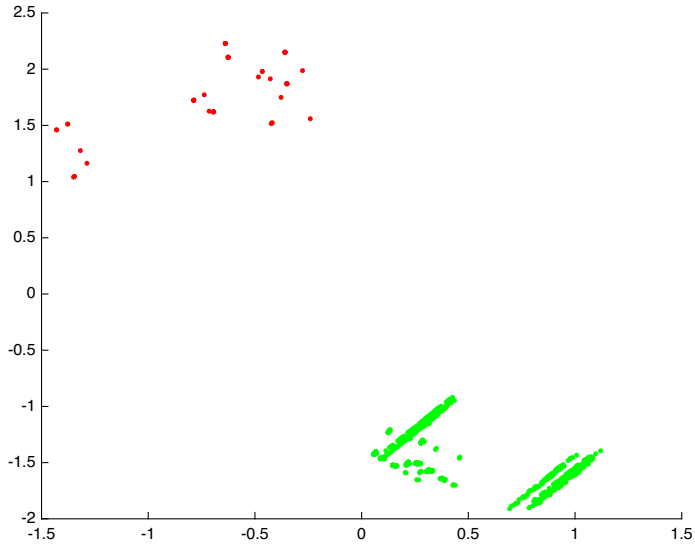

FIGURE 12. BHL 2D projection for segment 59.

TABLE 9. BHL parameters for segment 211.

Algorithm	Parameters
BHL	iters=10000, lrate=0.001, $\alpha = 4$, $\beta = 3$

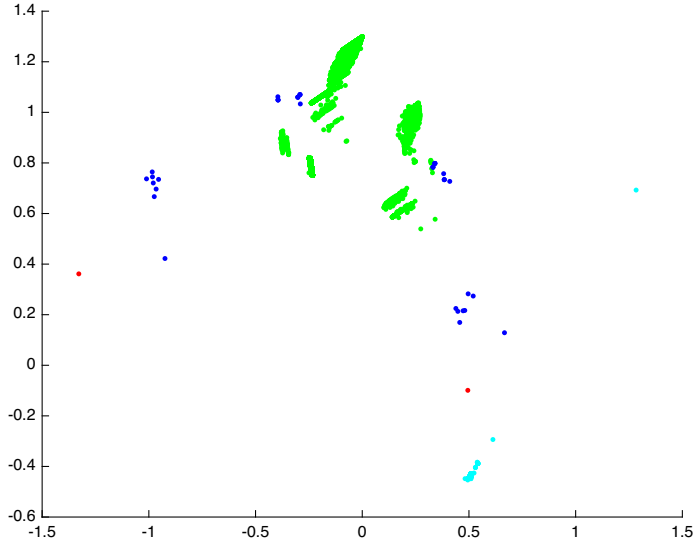

FIGURE 13. BHL 2D projection for segment 211.

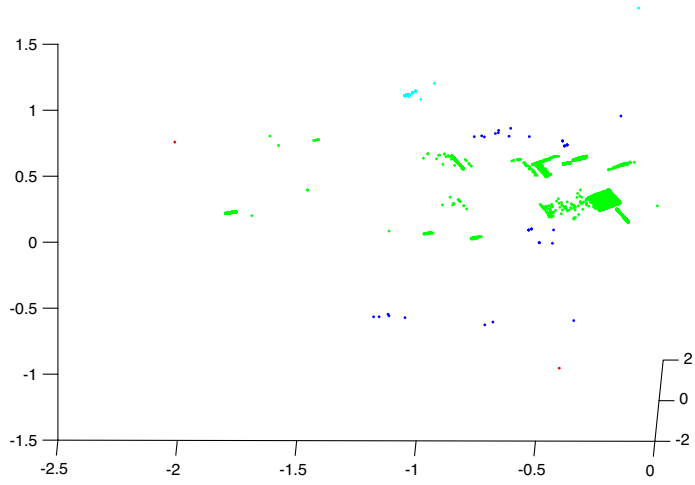

FIGURE 14. BHL 3D projection for segment 211.

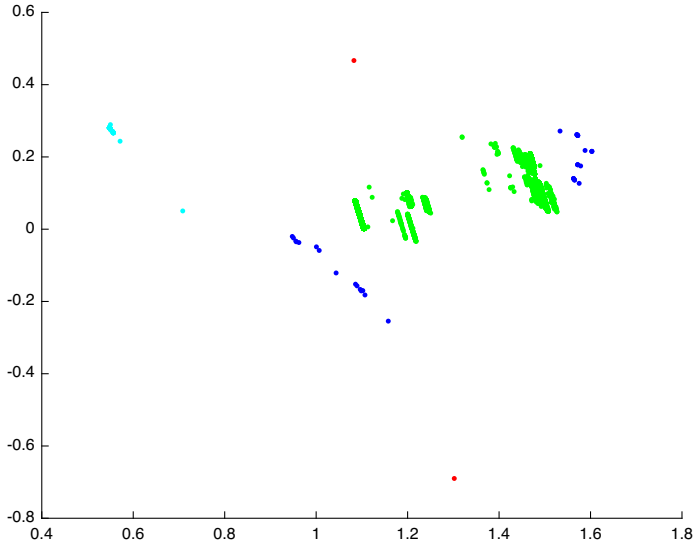

FIGURE 15. BHL 2D projection for segment 211, with [0,1] normalization.

projections are more informative than those obtained by other EPP methods in most cases. For the rest of segments, they are at least as good as those obtained by alternative methods.

The results of the conducted experiment have proven that BHL's performance is superior to that of the techniques used in previous researches, proving comprehensible projections, where attacks are clearly distinguished from the normal behaviour of the network, even when different types of attacks occur at the same time.

Thanks to this advanced AI visualization, security staff could easily monitor large networks and identify anomalous situations at a glance. Furthermore, this supervision could be performed without

extensive training on the applied techniques and without requiring a wide knowledge about the visualization resources.

In order to extend the present research, the authors propose the combination of the BHL projections with some other unsupervised visualization methods such as clustering. Furthermore, the applied method could be also validated in other cybersecurity problems, including the detection of malware and some other attacks (such as SQL injection).

Funding

Funding for open access charge: Universidade da Coruña/CISUG.

References

[1] I. Ahmad, M. A. Shah, H. A. Khattak, Z. Ameer, M. Khan and K. Han. Fiviz: forensics investigation through visualization for malware in internet of things. *Sustainability*, **12**, 2020.

[2] E. F. E. Ahmet, S. Saleh Hussin and H. A. Hussin.. Malware visualization techniques. *International Journal of Applied Mathematics Electronics and Computers*, **8**, 7–20, 2020.

[3] D. Atienza, Á. Herrero and E. Corchado. Neural analysis of http traffic for web attack detection. In *International Joint Conference*, , , , and Á. Herrero, B. Baruque, J. Sedano, H. Quintián and E. Corchado., eds, pp. 201–212. Springer International Publishing, Cham, 2015.

[4] A. Berro, S. L. Marie-Sainte and A. Ruiz-Gazen. Genetic algorithms and particle swarm optimization for exploratory projection pursuit. *Annals of Mathematics and Artificial Intelligence*, **60**, 153–178, 10 2010.

[5] V. Bulavas. Investigation of network intrusion detection using data visualization methods. In *2018 59th International Scientific Conference on Information Technology and Management Science of Riga Technical University (ITMS)*, pp. 1–6, 2018.

[6] E. Corchado and C. Fyfe. Connectionist techniques for the identification and suppression of interfering underlying factors. *IJPRAI*, **17**, 1447–1466, 2003.

[7] E. Corchado and Á. Herrero. Neural visualization of network traffic data for intrusion detection. *Applied Soft Computing*, **11**, 2042–2056, 2011.

[8] M. A. Ferrag, L. Maglaras, S. Moschoyiannis and H. Janicke. Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study. *Journal of Information Security and Applications*, **50**, 102419, 2020.

[9] E. Gandotra and D. Gupta. Improving spoofed website detection using machine learning. *Cybernetics and Systems*, **52**, 169–190, 2021.

[10] A. González, Á. Herrero and E. Corchado. Neural visualization of android malware families. In *Proceedings of the International Joint Conference SOCO'16-CISIS'16-ICEUTE'16*, pp. 574–583, 2016.

[11] S. Hajj, R. El Sibai, J. B. Abdo, J. Demerjian, A. Makhoul and C. Guyeux. Anomaly-based intrusion detection systems: the requirements, methods, measurements, and datasets. *Transactions on Emerging Telecommunications Technologies*, **32**, e4240, 2021.

[12] D. Han, J. Pan, R. Pan, D. Zhou, N. Cao, J. He, X. Mingliang and W. Chen. inet: visual analysis of irregular transition in multivariate dynamic networks. *Frontiers of Computer Science*, **16**, 1–16, 2022.

[13] Á. Herrero, E. Corchado and J. M. Sáiz. Movicab-ids: visual analysis of network traffic data streams for intrusion detection. In *Intelligent Data Engineering and Automated Learning—*

IDEAL 2006, E. Corchado, H. Yin, V. Botti and C. Fyfe., eds, pp. 1424–1433. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[14] Á. Herrero, U. Zurutuza and E. Corchado. A neural-visualization IDS for honeynet data. *International Journal of Neural Systems*, **22**, 2012.

[15] E. Jove, J. L. Casteleiro-Roca, H. Quintián, J. A. M. Pérez and J. L. Calvo-Rolle. A new approach for system malfunctioning over an industrial system control loop based on unsupervised techniques. In *International Joint Conference SOCO'18-CISIS'18-ICEUTE'18—San Sebastián*, pp. 415–425. Proceedings, Spain, June 6–8, 2018, 2018.

[16] E. Jove, J. L. Casteleiro-Roca, H. Quintián, J. A. M. Pérez and J. L. Calvo-Rolle. A fault detection system based on unsupervised techniques for industrial control loops. *Expert Systems*, **36**, 2019.

[17] A. Karami. An anomaly-based intrusion detection system in presence of benign outliers with visualization capabilities. *Expert Systems with Applications*, **108**, 36–60, 2018.

[18] J.-F. Lalande, M. Simon and V. V. T. Tong. Groddviewer: dynamic dual view of android malware. In *Graphical Models for Security*, I. I. I. Harley Eades and O. Gadyatskaya., eds, pp. 127–139. Springer International Publishing, Cham, 2020.

[19] H. Quintián and E. Corchado. Beta hebbian learning as a new method for exploratory projection pursuit. *International Journal of Neural Systems*, **27**, 1–16, 2017.

[20] H. Quintián, E. Jove, J.-L. Casteleiro-Roca, D. Urda, Á. Arroyo, J. L. Calvo-Rolle, Á. Herrero and E. Corchado. Beta-hebbian learning for visualizing intrusions in flows. In *13th International Conference on Computational Intelligence in Security for Information Systems (CISIS 2020)*, Á. Herrero, C. Cambra, D. Urda, J. Sedano, H. Quintián and E. Corchado., eds, pp. 446–459. Springer International Publishing, Cham, 2021.

[21] Á. H. R. Sánchez and E. Corchado. Visualization and clustering for snmp intrusion detection. *Cybernetics and Systems*, **44**, 505–532, 2013.

[22] R. Sánchez, Á. Herrero and E. Corchado. Clustering extension of MOVICAB-IDS to distinguish intrusions in flow-based data. *Logic Journal of the IGPL*, **25**, 83–102, 2016.

[23] O. Somarriba, U. Zurutuza, R. Uribeetxeberria, L. Delosieres and S. Nadjm-Tehrani. Detection and visualization of android malware behavior. *Journal of Electrical and Computer Engineering*, **2016**, 2016.

[24] A. Sperotto, R. Sadre, F. Van Vliet and A. Pras. A labeled data set for flow-based intrusion detection. In *International Workshop on IP Operations and Management*, pp. 39–50. Springer, 2009.

[25] R. V. Vega, P. Chamoso, A. G. Briones, J.-L. Casteleiro-Roca, E. Jove, M. Meizoso-López, B. Rodríguez-Gómez, H. Quintián, Á. Herrero, K. Matsui, E. Corchado and J. Calvo-Rolle. Intrusion detection with unsupervised techniques for network management protocols over smart grids. *Applied Sciences*, **10**, 2276, 2020.

[26] R. V. Vega, H. Quintián, C. Cambra, N. Basurto, Á. Herrero and J. L. Calvo-Rolle. Delving into android malware families with a novel neural projection method. *Complexity*, **2019**, 6101697:1–6101697:10, 2019.

[27] R. V. Vega, H. Quintián, J. L. Calvo-Rolle, Á. Herrero and E. Corchado. Gaining deep knowledge of Android malware families through dimensionality reduction techniques. *Logic Journal of the IGPL*, **27**, 160–176, 09 2018.

Received 20 February 2021