ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/322151615

Deep neural networks in cyber attack detection systems

Article *in* International Journal of Civil Engineering and Technology · November 2017

CITATIONS
5

READS
167

5 authors, including:

Ihor Tereikovskyi
National Technical University of Ukraine Kiev Polytechnic Institute
18 PUBLICATIONS **48** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Malware Detection Using Artificial Neural Networks View project

Project Detection of Extremist content on Kazakh language in web resources View project

All content following this page was uploaded by Ihor Tereikovskyi on 12 November 2019.

The user has requested enhancement of the downloaded file.

International Journal of Civil Engineering and Technology (IJCIET)
Volume 8, Issue 11, November 2017, pp. 1086–1092, Article ID: IJCIET_08_11_106
Available online at http://http://www.iaeme.com/ijciet/issues.asp?JType=IJCIET&VType=8&IType=11
ISSN Print: 0976-6308 and ISSN Online: 0976-6316

© **IAEME** Publication **Scopus** Indexed

DEEP NEURAL NETWORKS IN CYBER ATTACK DETECTION SYSTEMS

Ideyat Melsovich Bapiyev, Bekmurza Husainovich Aitchanov
Kazakh National Research Technical University after K.I. Satpaev,
22a, Satpaev Street, 050013, Almaty, Republic of Kazakhstan

Ihor Anatolyevich Tereikovskyi
National Technikal University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute",
14-a Polytechnichna str., 03056, Kyiv, Ukraine

Liudmyla Alekseevna Tereikovska
Kyiv National University of Construction Architecture,
Povitroflotsky Avenue 31, 03037, Kyiv, Ukraine

Anna Alexandrovna Korchenko
National Aviation University,
1 Kosmonavta Komarova Ave., 03680, Kyiv, Ukraine

ABSTRACT

This article discusses potentials of mathematical support improvement of detection systems of remote cyber-attacks on network resources of data systems. Herewith, efficiency improvement of cyber-attacks management is achieved by models on the basis of deep neural networks. This is aided by appropriate neural network model which is pre-trained by means of sparse auto encoder. A deep neural network is trained by means of a set of algorithms simulating higher-level abstractions in analyzed data using architectures comprised of a set of non-linear transformations. The proposed model is supported by software which facilitated its approbation for detection of network cyber-attacks. The model testing demonstrated that the accuracy of its basic variant is comparable with that of modern detection systems of network cyber-attacks.

Keywords: data protection, deep neural networks, detection of network cyber-attacks.

Cite this Article: Ideyat Melsovich Bapiyev, Bekmurza Husainovich Aitchanov, Ihor Anatolyevich Tereikovskyi, Liudmyla Alekseevna Tereikovska and Anna Alexandrovna Korchenko, Deep Neural Networks in Cyber Attack Detection Systems, International Journal of Civil Engineering and Technology, 8(11), 2017, pp. 1086-1092
http://www.iaeme.com/IJCIET/issues.asp?JType=IJCIET&VType=8&IType=11

1. INTRODUCTION

Nowadays network cyber-attack detection systems (CADS) are the most important tool of data protection of numerous computer systems and networks. Such systems have been used for several decades, they are developed by qualified experts, numerous works are devoted to development of scientific and methodological base, nevertheless, practical experience demonstrates [3] that CADS are characterized by certain drawbacks. Major one of them is insufficient accuracy of detection of overall range of network cyber-attacks. This drawback is evidenced both by practical results [7] and known cases of successful cyber-attacks on computer systems and networks in the US, Russia, Kazakhstan, and EU countries.

It is commonly recognized that insufficient detection accuracy of CADS is mainly related with the imperfection of software of such systems. One of the main approaches to CADS accuracy improvement is comprised of application of software on the basis of artificial neural networks [3]. Such CADS proved their efficiency, for instance, in hard- and software complexes of data protection by Cisco Systems, Inc. It should be mentioned that approbated systems mainly apply neural network models based on double-layer perceptron, Kohonen maps and associative neural networks. Along with that development of theory of artificial neural networks is related mainly with the so called deep neural networks. At present deep neural networks confirmed their advantage in comparison with classic neural network models in complicated cases of detection, the management of which requires for high amount of computational resources. These are speech analysis, handwriting recognition, image analysis. This can be exemplified by voice recognition system in Google browser. At the same time, the deep neural networks in CADS are not widely applied at present. Therefore, this work is aimed at determination of potentials of application of model on the basis of deep neural network in the detection systems of cyber-attacks. It is recognized that this purpose could be achieved by development of appropriate neural network model and to approbate in upon detection of network cyber-attacks.

2. DEVELOPMENT OF MODEL OF DEEP NEURAL NETWORK FOR DETECTION OF NETWORK CYBER ATTACKS

According to [3, 7], in modern CADS neural networks are used both for detection of deviation of controlled computer safety parameters from normal state and for detection of conformity of the considered parameters to signatures of cyber-attacks.

It should be mentioned that detection of network cyber-attacks by analysis of deviation of safety parameters from normal state depends on properties of protected computer system. Therefore, a neural network model adapted to detection of network cyber-attacks on specified computer system should be readjusted for detection of cyber-attacks on another computer system. At the same time the parameters of neural network model adapted to detection of signatures in fact do not depend on composition and structure of protected computer system. Accordingly, the obtained results will have universal pattern and allow estimating potentials of application of deep neural networks for protection of different computer systems. As a consequence, attention is focused on development pf deep neural network model intended for detection of signatures of network cyber-attacks.

In general case deep neural network is an artificial neural network with more than two hidden layers [1,2,4]. Similar to regular neural networks, deep neural networks are capable to simulate complex non-linear links between elements. While training deep neural network, the obtained model is an object in the form of combination of simple primitives. For instance, in face detection such primitives can be presented by separate portions of face: nose, eyes, mouth and so on. In detection of cyber-attacks such primitives can be presented by combinations of various parameters of network traffic. Additional layers make it possible to simulate abstractions of higher levels, thus facilitating development of models for detection of complex real-time objects.

Another peculiar feature of deep neural networks is their training. Deep training is a set of algorithms simulating higher level abstractions in analyzed data on the basis of architectures comprised of numerous non-linear transformations.

There exist numerous architectures of deep neural networks. Most of them originate from basic architecture. Simultaneous comparison of efficiency of different architectures is not always possible since not all of them have been estimated on the basis of similar data sets. Deep training is a rapidly progressing trend and new architectures, variants or algorithms appear quite frequently. However, according to [4-6], most modern training methods are subdivided into two main stages: pre-training and training itself implemented in basic case by backpropagation algorithm.

Herewith, the modern methods are varied by implementation of pre-training based on application of autoencoder. Thus, for estimation of potentials one of basic models of deep neural network was used on the basis of three-layer perceptron which was pre-trained using sparse autoencoder.

The autoencoder structure is illustrated in Fig. 1 [4, 8]. It should be mentioned that in Fig. 1 output signals of input neurons are marked with "*x*", hidden neurons – by "*a*", outputs– by "*y*", and offset block – by "+1".

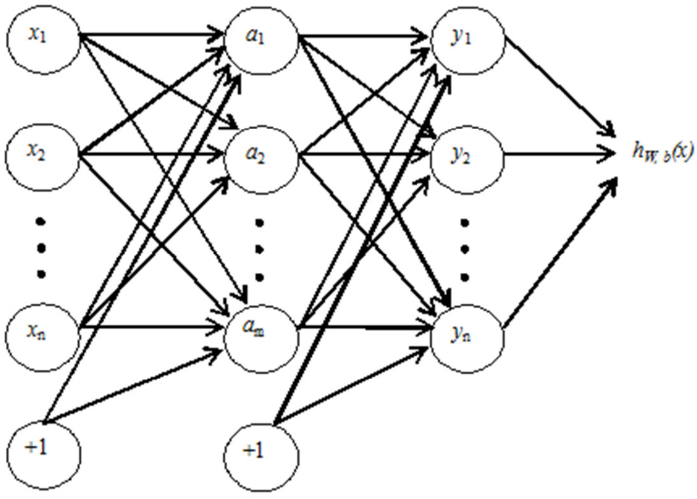

Figure 1 Architecture of auotencoder

Input data of auto encoder are presented by unformatted training set $x = (x_1, x_2, \ldots, x_i)$. Sigmoidal activation function is used in hidden and output neurons:

$$f\left(z_k\right) = \frac{1}{1+e^{-z_k}} \tag{1}$$

Where z_k is the cumulative input signal of the *k*-th neuron in hidden or output layer

In its turn:

$$z_k = \sum_{i=1}^{n} \left(W_{i,k} x_{i,k} + x_0 b_k\right) \tag{2}$$

where $W_{i,k}$ is the link weight from the *i*-th neuron of previous layer to the *k*-th neuron in hidden or output layer, $x_{i,k}$ is input signal of the *i*-th neuron of previous layer to the *k*-th neuron, x_0=1 is the link weight of neuron with itself, b_k is the offset of the *k*-th neuron (link weight coefficient of the *k*-th neuron with itself).

The output signal of autoencoder with *l* neural layers is:

$$h_{W,b}(x) = a^{(l)} \tag{3}$$

Where *W* is the array of weigh coefficients, *b* is the array of offsets, $a^{(l)}$ is the array of output values of neurons in the layer *l.*

As applied to Figure 1.

$$a^{(l)} = y \tag{4}$$

Where *y* is the array of input values of neurons of the last (*l*-th layer

A peculiar feature of autoencoder is training without teacher using backpropagation algorithm. With this aim the objective function of autoencoder training is defined as follows:

$$h_{W,b}(x) \approx x \tag{5}$$

The use of Eq. (5) assumes equality of output autoencoder signal to input signal. Therefore, training of classic autoencoder is reduced to searching for such values of weight coefficients by backpropagation algorithm when output signal equals to input signal [2, 5]. Herewith, training examples can be unmarked, that is, not contain expected input signal. Searching for optimum value of weight coefficients is performed using gradient descent by minimizing loss function:

$$J\left(W,b\right) = \left[\frac{1}{m}\sum_{i=1}^{m}\left(0{,}5\left\|h_{W,b}(x^{(i)}) - y^{(i)}\right\|^2\right)\right] + 0{,}5\lambda\sum_{l=1}^{m}\sum_{i=1}^{s_{l-1}}\sum_{j=1}^{s_l}\left(W_{j,i}^{(l-1)}\right)^2 \tag{6}$$

Where *m* is the number of hidden layers, s^l is the number of neurons in the layer *l*, $W_{j,i}^{(l-1)}$ is the link weight between the neuron *i* in the layer *l* and the neuron *j* in the layer *(l-1).*

The first part of the functional is the averaged squared error over all training examples, the second part is the regularization (or control of weight decay) which controls the order of weights and prevents re-training. Parameter λ, which controls weight decay, adjusts relative significance of the two parts of the functional.

Training is performed until:

$$J\left(W,b\right) < \theta \tag{7}$$

Where θ is the preliminary determined coefficient (threshold).

With regard to classic variant the peculiar feature of sparse autoencoder is restriction of number of simultaneously active neurons in intermediate layers. It is assumed that due to this the sparse autoencoder is automatically trained to highlight general features in input data which are reflected in weight coefficients. With this aim the additional component is added to the function:

$$P = \sum_{j=1}^{h}\left(p\log\frac{p}{\hat{p}_j} + (1-p)\log\frac{(1-p)}{(1-\hat{p}_j)}\right) \tag{8}$$

Where $\hat{p}_j$ is the average value of activation function of the neuron *j* over all training examples, $p \approx 0.05$ is the sparseness parameter.

It should be mentioned that a neuron is considered as active if its output signal is close to 1, and as inactive – to 0. With consideration for Eq. (8) the optimized loss function of sparse autoencoder is as follows:

$$J_s(W,b) = J(W,b) + \beta P \tag{9}$$

Where β is the predefined coefficient (in the first approximation $\beta \approx 3$).

Deep neural network with *m* neuron layers is pre-trained as follows:

1) Weight coefficients of all synaptic links are randomly initiated.
2) On the basis of required training accuracy the coefficient θ is predefined.
3) The number of trained layer is preset $l = 2$ (the number of input layer is 1).
4) A new additional layer is connected to the *l*-the layer of neurons.
5) A set of training examples is delivered to the input of the *l*-th layer.
6) Using Eqs. (1-9) the matrix of coefficients of link weights of the *l*-th layer of neurons is calculated.
7) The neuron layer connected at stage 4 is removed.
8) If $l < m$, then $l = l + 1$ and transition to stage 5 is carried out. Otherwise, pre-training is terminated.

After the pre-training stage two last layers of deep neural network are trained on marked data.

3. ANALYSIS OF THE DEVELOPED NEURAL NETWORK MODEL

The developed model is implemented in the form of appropriate software written in programming language Python. Selection of the programming language is stipulated by its approbation in tasks of machine training. In addition, while developing the software the supplemental library TensorFlow was used (developed by Google). This library makes it possible to automate most operations related with training and detection of various types of neural network models. Additional advantages of the library are its free-of-charge basis and open source code.

Training set was generated in the basis of NSL-KDD database which is a modification of well-known KDD-99 database. Brief description of NSL-KDD attributes is summarized in Table 1. These attributes served as input parameters for neural network model. Therefore, the number of input neurons is 40 which corresponds to the number of attributes.

Table 1 Characteristics of NSL-KDD database attributes

No.	Attribute	Description
1	duration	Time of connection in seconds
2	protocol_type	Protocol (TCP, UDP)
3	service	Network service (http, telnet, etc)
4	flag	Connection status (connection, error)
5	src_bytes	Data amount transferred from source to recipient in bytes
6	dst_bytes	Data amount transferred from recipient to source in bytes

The mentioned attributes are combined into four groups:

- Basic attributes – parameters of TCP/IP connection (1-10).
- Traffic time attributes – they are estimated in two seconds after established connection (22-31).
- Content attributes (11-21).
- Host traffic attributes (32-41).
- Database contains values of each attribute for detection of the following types of network cyber-attacks:
- Distributed Denial of Service (DDoS) — cyber-attacks aimed at network blocking.
- Probe — cyber-attacks aimed at data scanning or detecting network vulnerability for subsequent attacks on other networks.
- Remote to Local (U2L) — cyber-attack aimed at establishing of non-authorized connection by sending packages to this network.
- User to Root (U2R) — cyber-attack aimed at obtaining administrator rights by common user.

The number of input neurons is 4 which corresponds to the number of detectable cyber-attacks. According to [3] the number of neurons in each hidden layer is 300. After training the developed neural network model was used for detection of examples not applied for training. Test set contained examples describing network connections in 24 hours. The results of detection are illustrated in Fig. 2. It should be mentioned that in Fig. 2 the number and types of detected cyber-attacks are shown for each hour. Average detection accuracy is about 90% which corresponds to the detection accuracy of cyber-attacks by means of well-known CADS [3, 7]

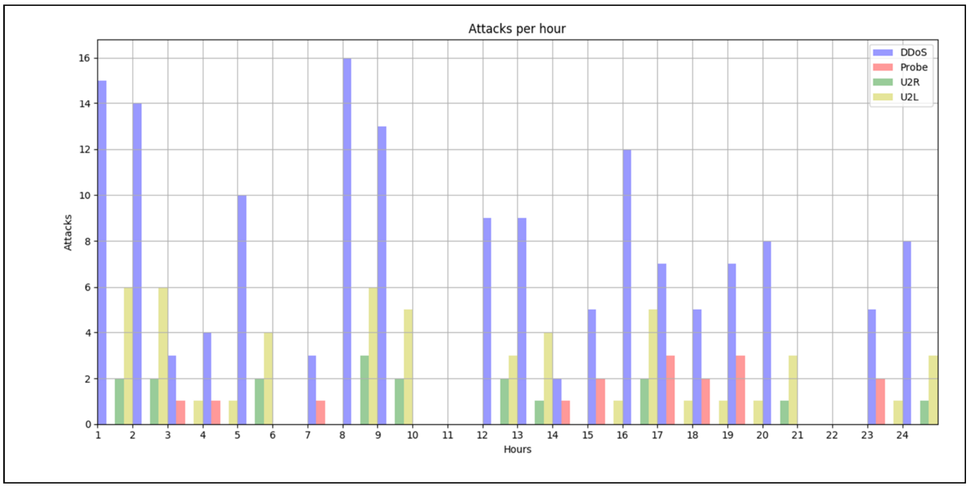

Figure 2 Resulting diagram of detection

4. CONCLUSION

It is demonstrated that one of the most promising approaches to development of detection systems of network cyber-attacks is improvement of their software by application of modern models on the basis of deep neural networks. This was aided by development of appropriate neural network model, which was pre-trained by sparse autoencoder. The proposed model was supported by software with subsequent approbation for detection of network cyber-attacks, the parameters of these attacks were stored in NSL-KDD database. The results of model testing demonstrated that the accuracy of basic variant is comparable with the accuracy of modern detection systems of network cyber-attacks.

REFERENCES

[1] Aytchanov B.H. and Bapiyev I.M. Razrabotka protsedury opredeleniya ozhidayemogo vykhodnogo signala neyrosetevoy modeli raspoznavaniya kiberatak, [Development of determination procedure of expected output signal of neural model of cyber-attack detection]. International Journal of Applied And Fundamental Research, 2017; 5: 8-11.

[2] Korchenko A., Tereykovskiy I., Karpinskiy N. and Tynymbayev S. Neyrosetevye modeli, metody isredstva otsenki parametrov bezopasnosti Internet-oriyentirovannykh informatsionnykh system, [Neural network models, methods and tools of estimation of safety parameters of interment information systems]. Kiev: TOV NashFormat, 2016

[3] Tereykovskaya L.A. and Tereykovskiy I.A. Using the expertise in the development of neural network model for recognition of phonemes in the voice signal [Text] the proceedings of the II International scientific-practical conference Information and telecommunication technologies: education, science and practice, Almaty, Kazakhstan, 2015, pp. 258–261.

[4] Akhmetov B.B., Korchenko A.G., Tereykovskiy I.A., Alibiyeva Zh.M. and Bapiyev I.M. Parameters of efficiency estimation of neural networks of cyber-attacks recognition on network resources of information systems. Reports of the National Academy of Sciences of the Republic of Kazakhstan, 2017; 2: 28–37.

[5] Bapiyev I.M., Akhmetov B.S., Korchenko A.G. and Tereykovskiy I.A. Primeneniye neyronnoy seti s radial'nymi bazisnymi funktsiyami dlya raspoznavaniya skriptovykh virusov, [Application of neural network with radial basic functions for detection of script viruses] II International Scientific and Practical Conference Actual issues of cybersecurity and information protection, Kyiv, Ukraine, 2016, pp. 21–24.

[6] Grishin A.V. Neural network technology in problems of detection of computer attacks. Information technology and computer systems, 2011; 1: 53-64.

[7] Yemel'yanova Yu.G. Analiz problem i perspektivy sozdaniya intellektual'noy sistemy obnaruzheniya i predotvrashcheniya setevykh atak na oblachnyye vychisleniya. [Analysis of issues and possibilities of development of smart system of detection and prevention of network attacks on cloud computations]. Program systems: Theory and application, 2011; 4: 17-31.

[8] Mustafayev A.G.) Neyrosetevaya sistema obnaruzheniya komp'yuternykh atak na osnove analiza setevogo trafika. [Neural network system of detection of attacks based on network traffic]. Voprosy bezopasnosti, 2016; 2: 1-7.

[9] R Vinoth Kumar and K Kishore Kumar, Exploitation of Content Management System Vulnerabilities To Launch Large Scale Cyber Attacks, International Journal of Civil Engineering and Technology, 8(10), 2017, pp. 1381–1395.

[10] Prof. Abhinav V. Deshpande. Implementation of a Robust and Safe Cyber Security System by Preventing the Intrusion of Outsiders by Formulation of a Novel and Efficient Cyber Law Enforcement Policy. International Journal of Information Technology & Management Information System (IJITMIS), 6(2), 2015, pp. 01-10.

View publication stats