SZÉCHENYI EGYETEM
UNIVERSITY OF GYŐR

1st Conference on Sustainability – COS '23

DOI: 10.62897/COS2023.1-1.62

USE OF AI IN OPERATIONAL TECHNOLOGY NETWORKS AND PACKET-BASED ATTACKS DETECTION

Zoltán Dobrády[a,*], Szilárd L. Takács[b], Timót Hidvégi[c]

[a] Swarco Futurit GmbH. A-2380 Perchtolsdorf, Mühlgasse 86
[b] Szechenyi Istvan University H-9026 Gyor, Egyetem ter 1
[c] Szechenyi Istvan University H-9026 Gyor, Egyetem ter 1
dobrady_zoltan@icloud.com

This research is focused on cybersecurity, including the detection of packet-based attacks. We collected a large amount of data by creating Honeypots and hosting them on virtualised private servers (VPS) with open IP addresses. The acquired data was analysed using different deep learning methods, such as Long Short-Term Memory (LSTM) and one-dimensional convolutional neural network techniques. These algorithms were used to compare the measurements with currently used packet analysis techniques, resulting in the identification and development of the most efficient packet analysis procedure. Additionally, we conducted regression tests in isolated and simulated environments using the attack mechanisms that had already been detected. Once the packet analysis concept was developed, our goal was to improve a classification algorithm. The construction of a penalty decision algorithm was crucial. We also conducted extensive regression testing of the concept from various perspectives. Upon completion of our investigation, it was discovered that natural and statistically-based language models can identify cyber-attacks. Statistical models that better fitted were SVC, Logistic, and Naive Bayes, with a 69 % accuracy for packet-based attack detection.

1. Introduction

The Operational Technology (OT) network connection is used for production devices such as robots, transfer bands, pumps, packaging machines, and other similar devices. The machines are usually controlled by PLCs, and they use this network to communicate with each other.

A packet-based attack involves altering data streams to trigger harmful actions. The primary motive behind this procedure is achieving economic gain. The cyberattack, from a process engineering perspective, has the potential to result in particularly intricate failures that could potentially impede the operations of production machines, produce defective products, or even result in the complete shutdown of production. Nowadays, it's not uncommon for these attacks to affect entire manufacturing establishments.

Our literature review, focusing on articles published between 2021 and 2023, revealed a significant increase in the amount of scientific research on cybersecurity topics in OT networks. For example, an article (Aljabri, 2021) points to the importance of cyber defence and its escalation. This trend has also increased in the world of IoT (Kuzlu, 2021). The previous two studies are confirmed by a third one (Saharkizan, 2020). All three authors analyse the potential of artificial intelligence for cyber security. However, our goal is to apply pre-trained artificial intelligence to microcontrollers in the near future. Therefore, we decided to implement a TCP/IP network structure with a minimal number of clients.

Consider a Modbus TCP/IP network topology as an example. A data packet is sent to each peripheral. When searching for a failure or anomaly, these packets must be analysed to find the root cause of the problem.

This analysis is usually done manually by humans (e.g., in Wireshark or Tshark software), which adds an additional source of error.

The identification of the source and destination of packets on OT networks is accomplished by labelling them with IP addresses. Since a large number of clients share a common OT network, data streams could be generated every nanosecond.

The increased number of peripherals in OT networks makes the analysis of packets in real-time difficult or impossible. The speed of communication has also increased. Packets are now arriving on a given bus in real-time at nanosecond intervals. For the reasons above, data packets are rarely analysed in real-time on OT networks.

Because of the packet-based attacks, real-time analysis is a necessity (in OT or industrial systems networks). It can help to detect cyber-attacks and other failures. Combined with artificial intelligence, it could even be predictive.

A non-standard data stream can be interpreted as an attack. This allows us to intervene almost immediately, even at the initial stage of the attack. This can prevent significant financial damage.

This research focuses on finding attacks in network data, specifically looking at data packets from network traffic.

The main question addressed in this study is the performance of different artificial intelligence subfields (statistical learning and deep learning models) in packet analysis tasks.

2. Methodology

The main protocol in industrial systems is Modbus TCP/P. Modbus is a communication channel in industrial systems, much like a wired network at home. TCP is a communication protocol that ensures communication stability between sender and receiver. The IP indicates that packets are addressed using Internet protocol.

In a Modbus TCP/IP communication, a request must be answered with a response. A practical example would be that a PLC sends a packet to a peripheral and waits for a response according to the protocol. If the request cannot be interpreted, the server sends an exception, which can be a cause for a system error. These exceptions can indicate attacks. Figure 1 shows a request and an exception. An exception has two codes (Function code and exception code).

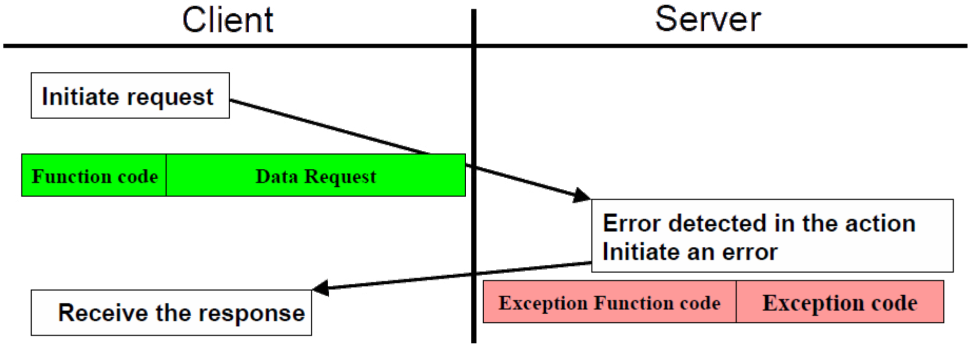


Figure 1: Modbus TCP protocol with an exception

For the simulated attack, a diverse real-world data set scenario had to be created together with a virtual environment. The structure of this environment is illustrated in Figure 2.

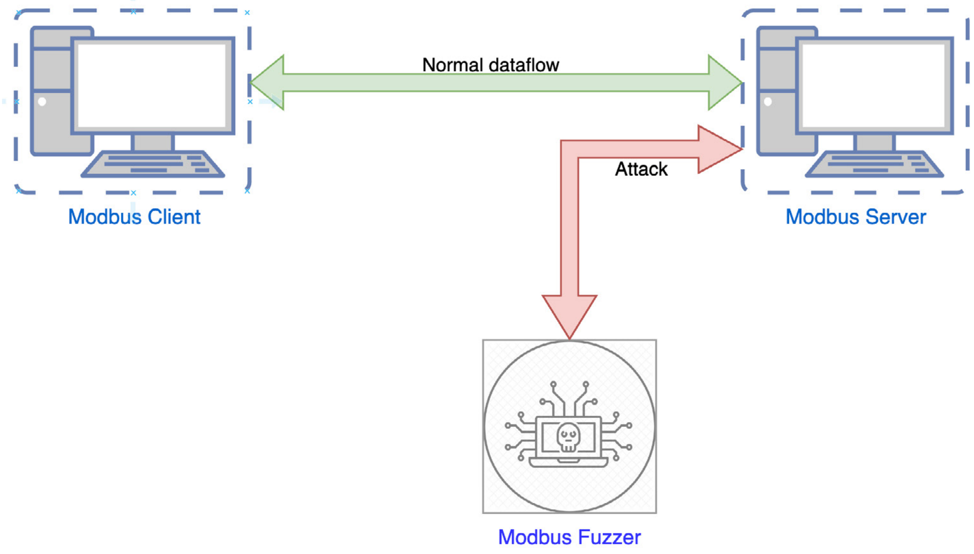


Figure 2: The theoretical construction of a simulated data acquisition environment

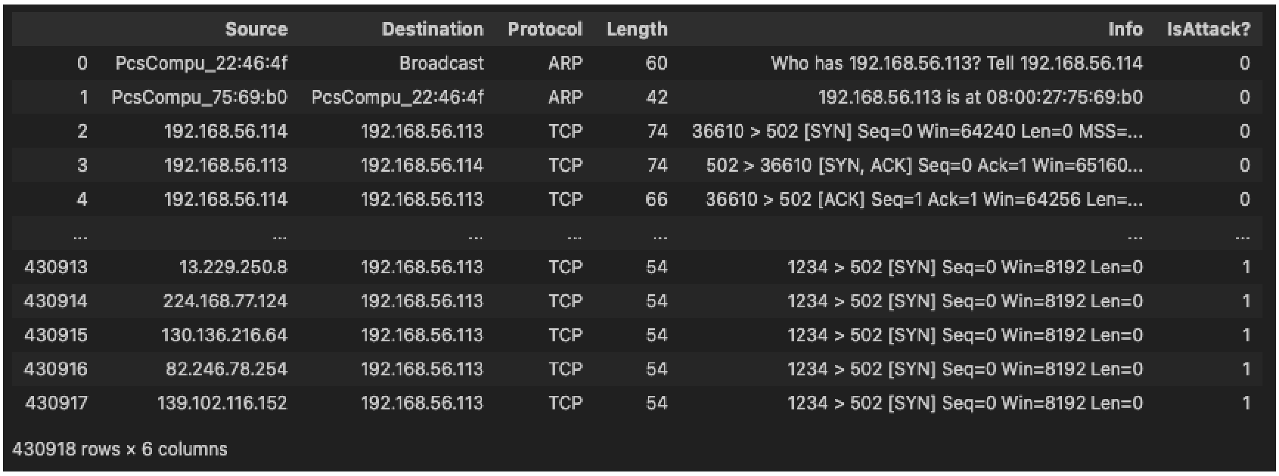
We used three virtual machines with standard Modbus Server/Client communication. A fuzzer was used in addition between the two devices. So, the third virtual machine was allocated to the fuzzer. Fuzzers are generally used for examining and testing applications/protocols. Fuzzers can be used to discover vulnerabilities and weaknesses in applications, making the implementation process easier. Therefore, signs of fuzzer usage should be monitored in an industrial network.

Our attack technique employs the Fuzzer to essentially flood the Modbus server with large amounts of data, resulting in little to no communication resources left on the client side. The impact of this suboptimal execution on the environment is that the peripherals controlled by the Client slow down or stop altogether. 430,918 unique packets have been generated for the simulation, from which 133,995 packets were part of the attack. During the attack mechanism, we conducted a comprehensive monitoring of the system processes by utilising the network packet analyser software, Wireshark. Additionally, the data streams were archived for each peripheral. We have labelled the individual packets of the complete data array for artificial intelligence (AI). Afterwards, we split this dataset into two parts, using an approximate 75 % - 25 % ratio, where the larger portion was used for the training method and the smaller portion for validation.

Following the pre-processing of the data, we employed two approaches for machine learning. Initially, we used statistical-based learning, and subsequently, we utilised a neural network, specifically a language model. In the case of statistical learning, we also applied Cross-validation, which provided six values for each model. We took into consideration the maximum value and the average of these values.

Figure 3 shows the dataset created. The dataset contains six columns, which are:

- Source – the source of the package
- Destination – the destination of the package
- Protocol – the protocol used to send the packets.
- Length – length of the package
- Info – other information about the package.
- IsAttack? – It shows whether the packet is an attack or not. 0 indicates no attack, while 1 indicates an attack.



	Source	Destination	Protocol	Length	Info	IsAttack?
0	PcsCompu_22:46:4f	Broadcast	ARP	60	Who has 192.168.56.113? Tell 192.168.56.114	0
1	PcsCompu_75:69:b0	PcsCompu_22:46:4f	ARP	42	192.168.56.113 is at 08:00:27:75:69:b0	0
2	192.168.56.114	192.168.56.113	TCP	74	36610 > 502 [SYN] Seq=0 Win=64240 Len=0 MSS=...	0
3	192.168.56.113	192.168.56.114	TCP	74	502 > 36610 [SYN, ACK] Seq=0 Ack=1 Win=65160...	0
4	192.168.56.114	192.168.56.113	TCP	66	36610 > 502 [ACK] Seq=1 Ack=1 Win=64256 Len=...	0
...	...	...	...	...	...	...
430913	13.229.250.8	192.168.56.113	TCP	54	1234 > 502 [SYN] Seq=0 Win=8192 Len=0	1
430914	224.168.77.124	192.168.56.113	TCP	54	1234 > 502 [SYN] Seq=0 Win=8192 Len=0	1
430915	130.136.216.64	192.168.56.113	TCP	54	1234 > 502 [SYN] Seq=0 Win=8192 Len=0	1
430916	82.246.78.254	192.168.56.113	TCP	54	1234 > 502 [SYN] Seq=0 Win=8192 Len=0	1
430917	139.102.116.152	192.168.56.113	TCP	54	1234 > 502 [SYN] Seq=0 Win=8192 Len=0	1

430918 rows × 6 columns

Figure 3: The prepared dataset

2.1. Using statistical learning

The focus of the article is on statistical learning methods. During the investigation, various statistical algorithms were applied, such as linear regression (Alpaydin, 2020), logistic regression (Russell and Norvig, 2005), k Nearest Neighbors (kNN) (Hastie et al., 2009), linear discriminant analysis (LDA) (Ghojogh and Crowley, 2019), quadratic discriminant analysis (QDA) (Ghojogh and Crowley, 2019), support vector machines (SVC) (Siegelmann and Vapnik, 2001), naive Bayes (Duda , 2001), and random forest (Breiman, Cutler, 2001).

2.2. Natural Language Processing

After the statistical-based machine learning, we switched to a different approach, where we employed deep learning techniques. Among these techniques, we utilised a language model, which posed several challenges, one of which was that the artificially generated dataset was not always suitable for training such models. As a result, we decided to use the same data as during the statistical learning.

To achieve this, we applied two different language learning models: the Long Short-Term Memory (LSTM) model (Bayer, 2015) and the 1D Convolutional model. We prepared the dataset by combining the 'Source IP', 'Destination IP', 'Protocol', and 'Length' attributes into a single string and labelled the rows to indicate whether they were considered attacks or not. The newly created dataset consists of two columns: one containing the mentioned attributes ('data'), and the other indicating whether the corresponding row is an attack or not ('IsAttack?'). The structure of the finalised dataset is shown in Table 1.

Table 1: The new dataset

Data				IsAttack?
Source IP	Destination IP	Protocol	Length	(0 = No attack,1 = Attack)
210.11.140.185	192.168.56.113	TCP	54	1
14.221.153.215	192.168.56.113	TCP	54	1
176.137.215.247	192.168.56.113	TCP	54	1
88.64.227.9 192	192.168.56.113	TCP	60	0
3.156.6.135 192	192.168.56.113	TCP	60	0

3. Result and analysis

After pre-processing the dataset, we applied word-based tokenization on the 'data' column and created embeddings to construct the training dataset, which could be processed by the language models.

3.1. Using statistics learning

In statistics-based learning, running the learning, validation methods produced the results shown in Table 2.

Table 2: Accuracy of statistical learning algorithms

Models	Accuracy score
Linear regression	0.443869
Logistic regression	69.04627
kNN(k Nearest Neighbors)	31.23024
Linear Discriminant Analysis	68.80032
Quadratic Discriminant Analysis	68.75143
Support Vector Machine	69.04627
Naive Bayes	69.04627
Random Forest	46.13880

Figure 4 illustrates the results achieved by different algorithms. The kNN algorithm demonstrated the lowest performance, which could be due to the encoding process. After creating a mapping during the process, we formed a library that contained all IP addresses and protocols, which we used to establish the training database based on the encoded table. However, due to the occurrence order, the information was lost. In Figure 4b, we can observe the same comparison without the kNN and random forest algorithms. It becomes evident that the SVC, logistic regression, and naive Bayes algorithms yielded the best results.

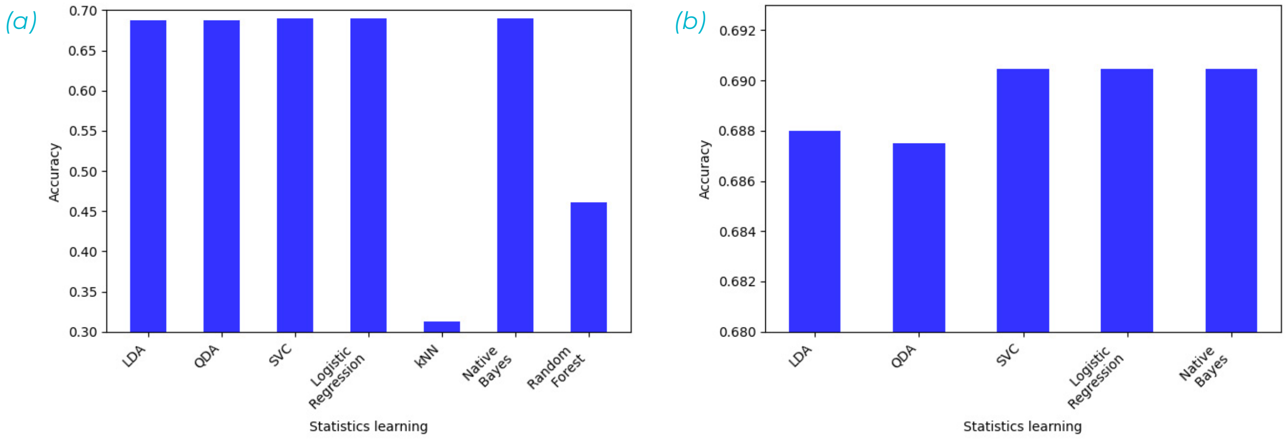


Figure 4: Results obtained with different statistics learning algorithms

3.2. Using natural learning

The LSTM model (Bayer, 2015) demonstrated higher efficiency compared to the 1D Convolutional model (Luo et al., 2016). Both LSTM and 1D Convolutional models achieved outstanding results in detection, with accuracy near the maximum threshold (approximately 99.9 %). The models were trained over 20 epochs.

Figure 5 shows a comparison of the LSTM model and 1D Convolution mesh. From the data set, we can observe that they are not appropriate for the model. Nevertheless, we observed that the dataset is not optimal for model training, as seen earlier, where statistical models also efficiently recognized the IP addresses. Two issues arise in this context:

- Limited or artificial data sample
- Overlay

These are discussed in the next two sub-chapters.

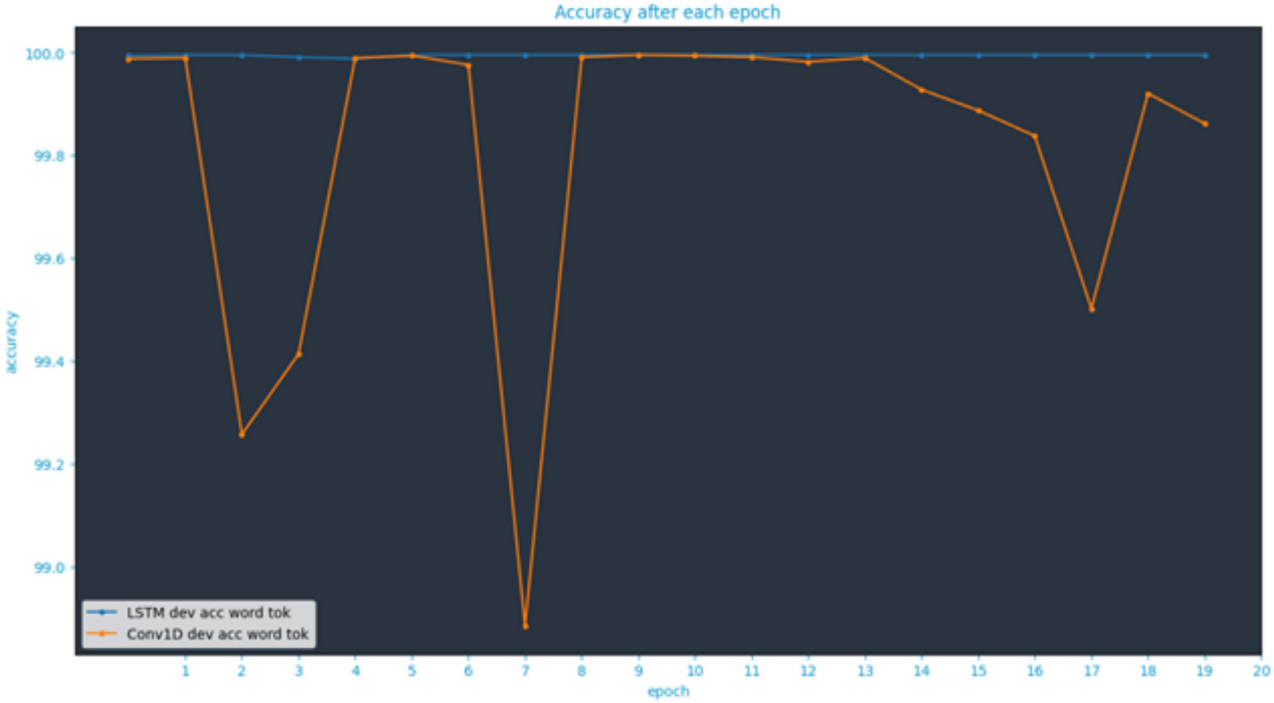


Figure 5: Comparison of LSRTM and 1D convolution model accuracy over 20 epochs

If the language model is trained and tested on a restricted or artificially generated dataset that does not adequately reflect the diversity and complexity of the real world, it is possible that the model may exhibit superior performance on this particular dataset; however, this may not be the case for other datasets. The dataset we have prepared simulated real data but is not fully adequate for teaching the language model. This is because in real attacks, our computer may receive a specific packet more than once. Attacks can be very diverse and independent. of each other.

There are several reasons why the model may be not suitable for the task. In the learning phase, it may learn features that only appear in the data set and cannot be generalised to other real-time data. This problem may arise if the model is not properly validated or if there is inadequate control against noise and redundancy in the data set.

It is likely that the models will start to learn the source IP addresses and ignore the other information. This is a problem because the 'protocol' and 'length' are relevant information. However, in this case, overfitting occurs. This presents a challenge as it is feasible to deceive an attack detection system based on IP addresses due to their potential for modification.

This raises the question of how other information should be considered in order to optimally train the model.

To prevent over-learning, further simulations and comparison measurements could be carried out by extending the simulation database. In this case, the IP address of the normal Modbus client-server data stream should also be changed.

4. Conclusions

The purpose of the research was to identify packet-based attacks. We applied and compared statistics-based learning and natural language learning models. In the first step, we utilised statistical learning algorithms on the generated dataset, achieving approximately 70% accuracy with support vector machines (SVC), naive Bayes, and logistic regression. The model considered the source IP address, destination IP address, used protocol, and packet length. The utilisation of statistical learning algorithms alone may not be sufficient for reliably detecting attacks; however, the combination of different models could enhance this accuracy.

Next, we applied natural language models, including Long Short-Term Memory and 1D convolutional models. In both cases, the data was assumed to be overfitted, as the models could have focused only on the source IP address and ignored other relevant data (destination IP, protocol, length, etc.).

This is also the reason for the over-adjusted results produced by the natural language models. In order to verify whether the neural network abandons this behaviour, it is imperative to conduct a test in a larger network where the attacker’s IP address is randomly changed. Even though statistical and neural language models can identify data packets used for cyberattacks, they aren’t ready to be used in practice yet. Further research and testing of the above-mentioned recommendations is required.

Acknowledgement

We express our gratitude to Andrews IT Engineering Ltd for their assistance in generating the artificial data packets.

References

- Aljabri M., Aljameel S.S., Mohammad R.M.A., Almotiri S.H., Mirza S., Anis F.M., Aboulnour M., Alomari D.M., Alhamed D.H., Altamimi H.S., 2021, Intelligent Techniques for Detecting Network Attacks: Review and Research Directions. Sensors, 21, 7070.
- Alpaydin E., 2020, Introduction to machine learning, MIT Press Cambridge, Massachusetts, London, England 140 – 145.
- Bayer J.S., 2015, Learning sequence representations, PhD Thesis, Technical University of Munich, Munich, Germany, 13 – 16, 32 – 34.
- Ben-Hur A., Horn D., Siegelmann H.T., Vapnik V., 2001, Support vector clustering, Journal of Machine Learning Research, 2, 125–137.
- Breiman L., 2001, Random Forests. Machine Learning, 45, 5 – 32.
- Duda R.O., Hart P.E., Stork D.G., 2001, Pattern Classification. Wiley, New York, USA.
- Ghojogh B., Crowley M., 2019, Linear and Quadratic Discriminant Analysis: Tutorial, arXiv:1906.02590, <arxiv.org/abs/1906.02590>, accessed 21.08.2023.
- Hastie T., Tibshirani R., Fiedman J., 2009, The elements of statistical learning: Data Mining, Inference, and Prediction, Springer Science and Business, New York, NY, USA, 14 – 18.
- Kuzlu M., Fair C., Guler O.,2021, Role of Artificial Intelligence in the Internet of Things (IoT) cybersecurity. Discover Internet Things, 1, 7.
- Luo W., Li Y., Urtasun R., Zemel R.S., 2016, Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. Neural Information Processing Systems, 29, 4898 – 4906.
- Russell S.J., Norvig P., 2005, Artificial Intelligence a Modern Approach. Pearson – Prentice Hall, Upper Saddle River, New Jersey, 635 – 638, 726 – 727.
- Saharkhizan M., Azmoodeh A., Dehghantanha A., Choo K.R., Parizi R.M., 2020, An Ensemble of Deep Recurrent Neural Networks for Detecting IoT Cyber Attacks Using Network Traffic. IEEE Internet of Things Journal, 7, 8852 – 8859.