University of Novi Sad
Faculty of Technical Sciences

39. SAVETOVANJE PROIZVODNOG MAŠINSTVA SRBIJE
- SPMS 2023 -

39th INTERNATIONAL CONFERENCE ON PRODUCTION ENGINEERING OF SERBIA
- ICPES 2023 -

ZBORNIK RADOVA
PROCEEDINGS

Novi Sad, 26-27 October 2023

PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON PRODUCTION ENGINEERING OF SERBIA - ICPES 2023

Novi Sad 2023

Publisher: **UNIVERSITY OF NOVI SAD, FACULTY OF TECHNICAL SCIENCES**
DEPARTMENT OF PRODUCTION ENGINEERING
DEPARTMENT OF INDUSTRIAL ENGINEERING AND ENGINEERING MANAGEMENT
21000 NOVI SAD, Trg Dositeja Obradovica 6, SERBIA

Organization of this Conference was approved by Educational-scientific Council of Faculty of Technical Sciences in Novi Sad

Technical treatment and design: Milana ILIĆ MIĆUNOVIĆ
Miloš RANISAVLJEV
Branko ŠTRBAC
Miodrag HADŽISTEVIĆ

Printing by: FTN, Graphic Centre
GRID, Novi Sad

CIP classification:

CIP - Каталогизација у публикацији
Библиотека Матице српске, Нови Сад

621.7/.9(082)

САВЕТОВАЊЕ производног машинства Србије (39 ; 2023 ; Нови Сад)
Zbornik radova = Proceedings / 39. Savetovanje proizvodnog mašinstva Srbije, SPMS 2023 - 39th International Conference of Production Engineering, ICPES 2023, 26-27. October 2023, Novi Sad. - Novi Sad : Faculty of Technical Sciences, 2023 (Novi Sad : GRID). - XXV, 391 str. :ilustr. ; 24 cm
Dostupno i na: http://spms.fink.rs/abstract.html . - Radovi na srp. i engl. jeziku. - Tekst štampan dvostubačno. - Tiraž 60. - Str. X-XI: Predgovor ; Foreword / Miodrag Hadžistević. - Bibliografija uz svaki rad.

ISBN 978-86-6022-610-7

а) Производно машинство – Зборници

COBISS.SR-ID 127854345

Financing of the Proceedings was sponsored by the Ministry of Education, Science and Technological Development of the Republic of Serbia.

SPMS/ICPES 2023

39TH INTERNATIONAL CONFERENCE ON PRODUCTION ENGINEERING OF SERBIA 26-27 OCTOBER 2023

CONTENTS

Society of Production Engineering

SPMS 2023
39. Savetovanje proizvodnog mašinstva Srbije
ICPES 2023
39th International Conference on Production Engineering of Serbia

Faculty of Technical Sciences
University of Novi Sad

Novi Sad, Serbia, 26. – 27. October 2023

GENERATION OF LIGHTWEIGHT MODELS FOR CYBER-ATTACKS DETECTION ALGORITHMS USING KNOWLEDGE DISTILLATION

Dušan NEDELJKOVIĆ[1,*], Živana JAKOVLJEVIĆ[1]
[1]University of Belgrade - Faculty of Mechanical Engineering, Belgrade, Serbia
*Corresponding author: dnedeljkovic@mas.bg.ac.rs

Abstract: *Industry 4.0 paradigm has brought about the changes in the way we manufacture. The integration of Cyber-Physical Systems into the Industrial Internet of Things represents the basis for the transition from traditionally centralized to distributed control systems where the overall control task is achieved through the cooperation of different devices which implies their mutual communication and constant information exchange. However, ubiquitous communication between devices with communication and computation capabilities opens up space for various cyber-attacks which can lead to catastrophic damage to equipment and also can endanger the environment and human lives. Therefore, the development and implementation of cyber-attacks detection mechanisms are necessary to prevent negative effects. Deep learning (DL) techniques are successfully applied to generate models on which cyber-attacks detection algorithms are based. However, the size of the DL models is often unsuitable for implementation on industrial control devices that usually have significant computational constraints. The use of complex DL models may disrupt the operation of control systems and introduce unacceptable delays in real-time cyber-attacks detection algorithms. This paper explores the possibilities for application of knowledge distillation technique to generate lightweight DL models. These models are designed to align with the limitations of the devices on which they are deployed. The paper evaluates the performance of lightweight models in cyber-attacks detection algorithms, and compares them to algorithms based on DL models before distillation.*

Keywords: *Cyber-Physical Systems, Industrial Internet of Things, Cybersecurity, Cyber-attacks detection, Machine learning, Knowledge distillation.*

1. INTRODUCTION

In the context of Industry 4.0 [1], manufacturing and in particular Industrial Control Systems (ICS) have undergone a significant transformation primarily driven by the integration of Cyber-Physical Systems (CPS) into the Industrial Internet of Things (IIoT). This integration signifies a shift from centralized to distributed control systems, where diverse devices cooperate through constant communication and information exchange.

The widespread communication between devices equipped with communication and computational modules opens up a broad area for cyber-attacks that could have catastrophic consequences. Cyber-attacks can cause damage to equipment, disrupt manufacturing processes, and even pose significant hazards to the environment and human lives. Hence, it is

essential to create and deploy mechanisms such as Intrusion Detection Systems (IDS) for timely detection of cyber-attacks and prevention of their negative outcomes.

There are two main approaches for cyber-attacks detection: design-driven and data-driven [2]. In continuously controlled systems both approaches model the signal that is communicated between devices and detect the attack as a discrepancy between modelled (estimated) and signal values received through communication link. Design-driven methods rely on predefined rules and mathematically formalized models of processes that usually require stringent assumptions and unacceptable simplifications leading to IDS that are hardly applicable in the real-world, especially for the non-linear continuously controlled systems. On the other hand, data-driven techniques can automatically obtain process models that in most cases provide high accuracy and good generalization properties. The drawback of the latter methods is that a large amount of data from process is required for model generation. Nevertheless, the data-driven approaches represent a technique of choice for designing IDS in continuously controlled processes.

The use of deep learning (DL) techniques has been successful in creating models for cyber-attacks detection algorithms [3]. However, DL models tend to be large, which can be unsuitable for their implementation on industrial control devices with limited computational capabilities. This can cause disruptions to the control system and lead to delays in real-time cyber-attacks detection algorithms.

Several techniques, such as pruning, parameter sharing, and quantization can be employed to adapt DL model size and ensure transfer of knowledge to resource-constrained environments [4]. Regardless of the chosen technique, the objective is to uphold the accuracy while minimizing the computational complexity involved.

For example, pruning is a technique that reduces neural network complexity by removing less important weights (setting them to zero). The high sparsity level (e.g., 75% in [5]) with negligible accuracy loss and simplicity of application make pruning a widely used technique. On the other hand, parameter sharing considers using the same set of weights and biases for multiple neurons within or across layers of a neural network. Another technique - the weight-sharing algorithm proposed in [6] compresses neural networks by assigning optimized weights from a pre-trained network to a particular cluster in a Gaussian mixture prior. Quantization refers to reducing the precision or representation of numerical values, typically the weights and activations, from a high-precision format to a lower-precision format.

Although presented techniques have been proven useful in many applications, they have some significant limitations. Pruning can introduce additional hyperparameters, such as the pruning ratio (the proportion of weights to prune) or optimal threshold value, which need to be tuned to achieve desired results. Parameter sharing can reduce model size but may not work well for complex network architecture requiring distinct information in different layers. Finally, the implementation of quantization has been observed to substantially impact the accuracy loss in the case of larger neural networks [7].

To overcome these limitations, in this paper we opted to utilize knowledge distillation [8] techniques and to transfer knowledge from a larger, well-trained model (teacher) to a smaller model (student). In particular, this paper investigates the use of knowledge distillation techniques to create lightweight DL models that are customized to fit industrial control devices limitations. We evaluate the performance of models obtained using knowledge distillation in detecting cyber-attacks and compare it to the performance of DL-based IDS algorithms before distillation.

The remainder of the paper is structured as follows. In Section 2 we give a brief overview of the knowledge distillation technique, whereas Section 3 refers to the method used in this paper to develop IDS for ICS. The performance of IDS based on the distilled model and its

comparison with IDS based on the model before distillation is presented in Section 4. Finally, in Section 5 we provide conclusions and future work guidelines.

2. KNOWLEDGE DISTILLATION

As mentioned in Introduction, knowledge distillation techniques [8] transfer knowledge from a larger, well-trained model known as teacher to a smaller model referred to as student. Depending on the knowledge used for student learning, we can identify three basic categories of this technique [9]: 1) response-based, 2) feature-based, and 3) relation-based knowledge distillation. In response-based knowledge distillation, the student primarily tries to imitate the teacher model's final prediction by focusing on the response of its output layer. On the other side, the training of the student model in feature-based knowledge distillation is guided by employing both the final layer's output and the feature maps from intermediate layers. Finally, relation-based knowledge distillation thoroughly examines the connections and relationships between different layers of the teacher network.

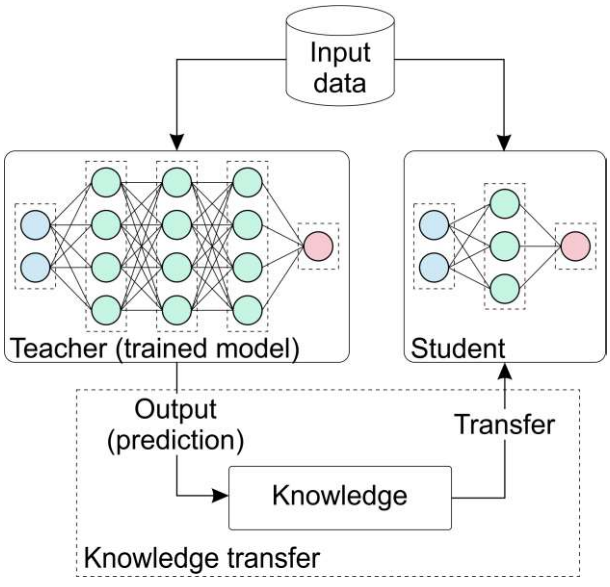

Figure 1. Response-based knowledge distillation

Since in our work the teacher model is previously chosen [10] using only its predictive performance, we opted for the response-based learning category to generate the distilled model. A schematic representation of the response-based knowledge distillation is shown in Fig. 1.

The process starts with input data fed into a larger, more complex teacher model and a smaller student model. The input data could be time series (1D), images (2D), or any data the models are designed to work with. The teacher model is already created based on the same input data, and a prediction is obtained at its output. On the other hand, the student model for training uses ordered pairs that are composed of input data and corresponding (desired) output data obtained from the teacher. In this way, the student tries to imitate the teacher and achieve the same prediction at output.

3. METHOD FOR THE DEVELOPMENT OF IDS IN ICS

Before developing the student model, it is essential to explain the process involved in creating the teacher model. In our previous research [10], we have developed a methodology for creating IDS in ICS utilizing a CNN-based approach. This method belongs to the class of self-supervised data-driven techniques and involves offline and online phases. During the offline phase, the method generates a CNN-based model of signals transmitted between IIoT devices. This model relies on the auto-regression of transmitted signal, estimating the current output y_i using a buffer of v previously received values $\mathbf{x}_i$, which can be written in the following way:

$$(\mathbf{x}_i, y_i) \in [([x_1,\ldots,x_v], x_{v+1}),([x_2,\ldots,x_{v+1}], x_{v+2}) \\ ,\ldots,([x_{i-v},\ldots,x_{i-1}], x_i),\ldots,([x_{n-v},\ldots,x_{n-1}], x_n)]. \quad (1)$$

With this approach, using the set criteria, it is possible to automatically select the appropriate model that represents the basis of IDS with good attack detection performance. In the offline phase, the hyperparameters of CNN-based model architecture are varied in such a way that they start from the model with the smallest number of parameters to obtain the least complex model that meets predefined criteria.

Offline IDS development involves three steps. The first part represents signal preprocessing and includes normalization by its maximum value, signal filtering (using FIR filters), creating the ordered pairs for training, and data shuffling. The second step implies the creation of unique Machine Learning (ML) model through the variation of hyperparameter values from previously defined sets. In the third step, the model is selected out based on two criteria:

1. Statistical characteristics (mean value and standard deviation) of the discrepancy between the real and estimated values must be similar for the training data and data for model selection.
2. The IDS should be robust to false positives; the robustness is tested on data received during normal conditions (without attacks), and the criterion is met if IDS does not detect any attack on this data.

If the model meets both criteria, it is selected as appropriate and the offline phase stops. IDS based on the selected model is used in the online phase for cyber-attacks detection through comparison of estimated and values received through communication links. An attack is detected if the difference between received and estimated values exceeds the threshold for *z* consecutive samples.

Since IDSs based on the created models have shown good detection capabilities [10], the question is whether it is feasible to create significantly smaller models that can provide equivalent detection performance.

4. LIGHTWEIGHT MODELS GENERATION USING KNOWLEDGE DISTILLATION

In this paper, we will utilize five different signals from two publicly available datasets to develop the ML models and test their performance. The following datasets are employed: 1) Secure Water Treatment (SWaT) and 2) Electro-pneumatic positioning system (DisEPP).

SWaT testbed [11] represents a fully operational scaled-down water treatment plant capable of producing 5 gallons of purified water per minute. The whole process is divided into 6 sequentially placed stages, each controlled by an independent Programmable Logic Controller (PLC). The data acquisition from 51 devices (25 sensors and 26 actuators) lasted 11 days. For the first 7 days, the system was operated under normal conditions (without attacks), and during the last 4 days, a total of 41 (5 without any physical impact on the system) attacks of various duration and intensity have been launched.

Sensors are divided into four different classes depending on the quantity they measure: flow (FIT), liquid level (LIT), pressure (PIT), and chemical properties (AIT). In this paper we will consider 4 out of 25 sensory signals. Two signals (LIT301 and PIT501) were chosen to include as many attacks as possible that affected their work directly or through adjacent devices. The other signals (FIT101 and AIT401) were chosen based on the most complex ML models obtained in [10] to demonstrate the advantages of the application of the knowledge distillation (possible higher reduction rate between original and distilled model). In addition, each chosen sensor belongs to a unique sensor class and stage.

The DisEPP, on the other hand, was created in the Laboratory for Manufacturing Automation at the University of Belgrade - Faculty of Mechanical Engineering. The main goal of DisEPP is to achieve the desired position of the pneumatic cylinder piston. The system is comprised of a smart actuator (rodless pneumatic cylinder with electro-pneumatic pressure regulator and local controller 1) and a smart sensor (electromagnetic linear encoder with local controller 2). The local controllers represent wireless nodes based on ARM Cortex-M3 that run at 96 MHz [12] augmented with IEEE 802.15.4-compliant wireless transceiver Microchip MRF24J40MA [13]. This dataset [14] was obtained by acquiring communicated data between the local controller 1 and local controller 2. The signal recorded during the piston's movement along a

trajectory of 3 positions was selected for further analysis; the sampling rate was 33.3 Hz. Since the dataset includes only signals recorded during normal behavior (without attacks), a total of 6 different attacks were created to test the performance of IDS in [10]. These attacks will be used to compare the performance of the IDSs based on the teacher and student models.

4.1 Generation of the student model

The generation of the student models is carried out in the same way as in the previously explained procedure, except that in equation (1) where the prediction given by the teacher model $\hat{y}_i$ is used as the output instead of the y_i; in this way, knowledge is transferred from teacher to the student. The selection of a suitable student model is based on the second criterion (Section 3). This means that the first model that satisfies this criterion is selected as appropriate and used in the online phase for attacks detection.

The preprocessing procedure (normalization, FIR filtering, ordered pairs generation, and shuffling) is applied to input signals. A buffer size of v=16 samples was used to predict the current value. The datasets are divided into training, validation, and data for model selection, with a share of 70/10/20%, respectively. The model was trained for 10 epochs using the Adam optimizer with a learning rate 0.001, and the cost function was the mean squared error (MSE).

To make the model development process less time-consuming and reduce the number of unique models, we employed a general architecture that has proven effective [10]. This architecture is comprised of two sequentially placed blocks containing two 1D-CNN layers and a max pooling layer. A flattening layer follows the second max pooling layer, and the network ends with two fully connected layers (Fig. 2). The CNN hyperparameters that will be varied during the development of unique ML models are the number of filters f_i ($i \in \{1,\ldots4\}$) and filter size fs in 1D-CNN layers, as well as the number of neurons in the first fully connected layer d_1. The downsampling rate in the max pooling layers is set to p=2, whereas the number of neurons in the output (fully connected) layer is determined by the number of output parameters (d_2=1).

The sets of hyperparameter values are defined so that even the most complex student model has fewer parameters than the simplest teacher model (table 1). The filter size was the only hyperparameter whose sets were the same for both models.

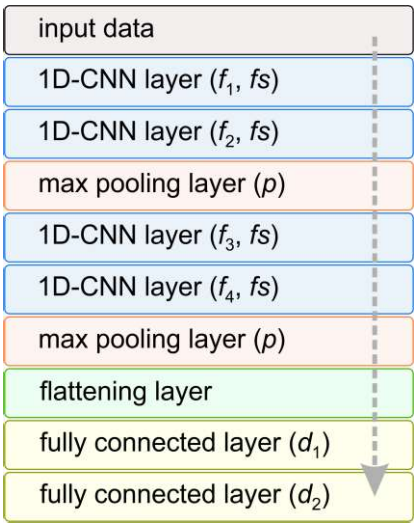

Figure 2. General 1D-CNN architecture

Table 1. Varied 1D-CNN parameters

hyperpar.	teacher	student
f_1	4, 8, 16	2, 4, 8
f_2	8, 16, 32	2, 4, 8
f_3	8, 16, 32	2, 4, 8
f_4	16, 32, 64	2, 4, 8
fs	2, 3, 4	2, 3, 4
d_1	30, 40, 50	5, 10, 20

Table 2 represents the architectures of obtained teacher and student models for considered sensory signals. The complete procedure for generation of teacher models is presented in [10], whereas the student models are created using the procedure from Section 2. The architectures of the created models differ in the number of filters f_1-...-f_4 and the number of neurons in the first fully connected layer d_1. The filter size for each signal and model is set to fs=2 and it is not presented in table 2.

4.2 Comparison of teacher and student models

As can be observed from table 2, the number of parameters in the selected student models ranges from 87 to 603. Comparing student models to the teacher models used for knowledge distillation confirms that the numbers of parameters have been successfully reduced for all sensory signals. A notable instance is observed in the AIT401 sensor signal, where the teacher model originally had 10,209 parameters but was distilled down to 211. The student models have 282.6, whereas the teacher models have 5585.8 parameters in average, which shows that the number of parameters has been reduced over 19 times in average.

Table 2. Teacher and student models architectures

sensor	model	f_1-...-f_4	d_1	param.
FIT101	teacher	4-8-8-64	30	9,049
	student	4-4-4-4	5	211
LIT301	teacher	4-8-8-16	30	2,473
	student	4-4-4-4	10	301
AIT401	teacher	4-8-16-64	30	10,209
	student	4-4-4-4	5	211
PIT501	teacher	4-8-8-16	30	2,473
	student	2-2-2-2	5	87
DisEPP	teacher	8-8-32-16	30	3,725
	student	8-8-8-8	5	603

During the online phase, an attack is detected if the discrepancy between the real and values estimated using generated model exceeds the threshold T for z=15 consecutive signal samples. The threshold is defined as a sum of the mean value (μ) and three times the standard deviation (σ) of the discrepancies between real and estimated values of the subset of the data used for model selection:

$$T = \mu_{ms} + 3\sigma_{ms} \quad (2)$$

Using IDSs based on student models, all 18 attacks on considered sensors and adjacent devices were successfully detected without false positive results (table 3). Table 3 shows the same results were achieved using IDSs based on the teacher models. The obtained results confirm that using the knowledge distillation technique based on existing models we can generate models with significantly fewer parameters that will provide the same detection performance as the original models.

Table 3. Performances of the selected models

model	number of detected attacks on:				
	FIT101	LIT301	AIT401	PIT501	DisEPP
teacher	/	7	/	5	6
student	/	7	/	5	6

Fig. 3 presents the examples of 3 out of 7 detected attacks on LIT301 sensor. Blue and red lines represent received data and obtained prediction, whereas the start and moment of

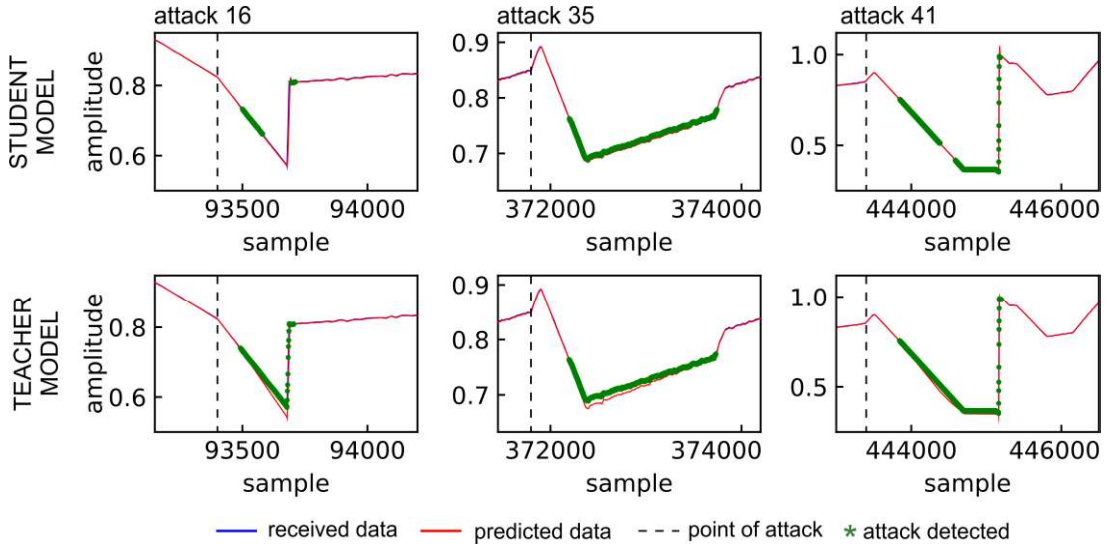

Figure 3. Example of the three detected cyber-attacks on LIT301

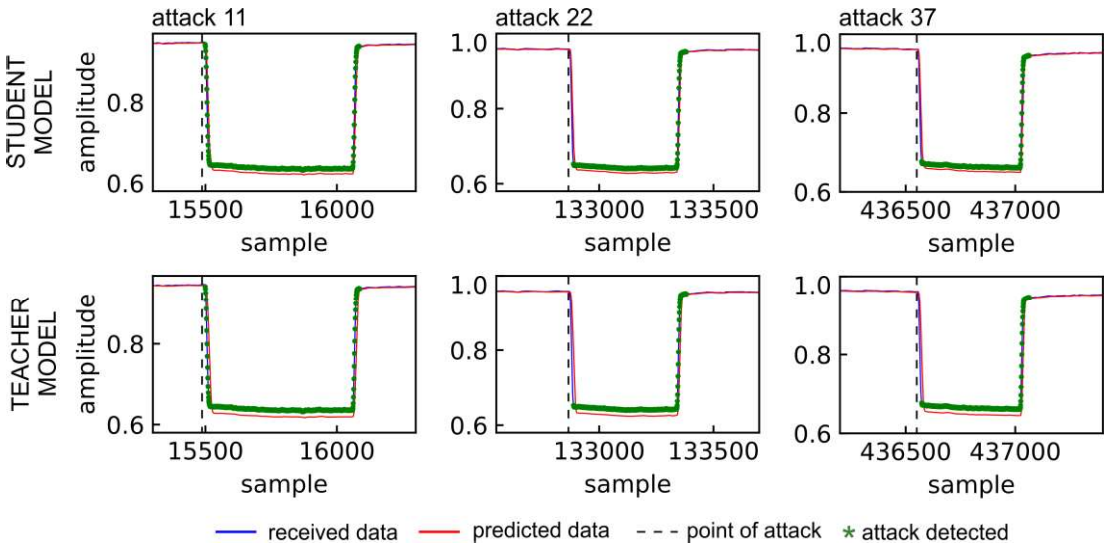

Figure 4. Example of the three detected cyber-attacks on PIT501

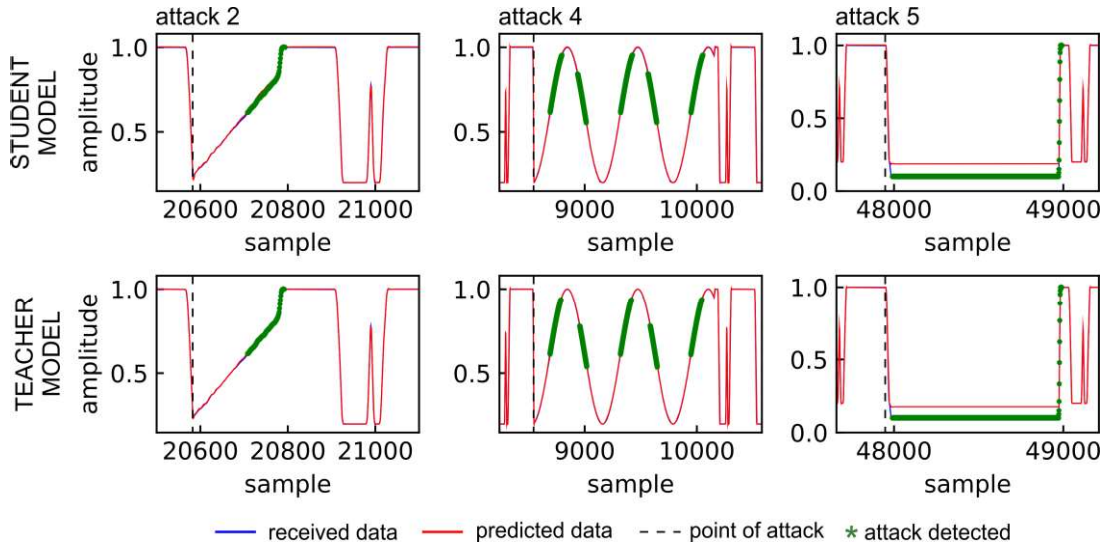

Figure 5. Example of the three detected cyber-attacks on DisEPP

the detection of the attack are marked with a black dashed line and a green marker, respectively. A linear increase/decrease in the signal value characterizes the shown attacks on LIT301.

Examples of detected attacks (3 out of 5) on PIT501 are shown in Fig. 4. The received data, prediction, start of the attack, and the moment of its detection are represented in the same way as in Fig. 3. Attacks (11, 22, and 37) on PIT501 have similar dynamics characterized by oscillating around the set signal value. For the signal from DisEPP, 3 out of 6 detected attacks are shown in Fig. 5. In addition, the same format of lines and markers was used to represent received data, prediction, and the moments of start and detection of attacks. The shown attacks on DisEPP have different dynamics defined by a linear increase (attack 2), changing the signal value as a harmonic function (attack 4) or setting the signal value to a constant for a certain period (attack 5).

In addition to the capability of IDSs based on the student models to detect all attacks, it can be noted that the moments of detection are almost the same as in the case of IDSs based on teacher models.

5. CONCLUSION

This paper explored the utilization of knowledge distillation to generate lightweight models for cyber-attack detection in ICS. The development of lightweight models was based on the proven 1D-CNN models obtained in previous research. Appropriate models were

chosen based on specific criteria, resulting in a significant reduction in the number of parameters (e.g., from 10,209 to 211 parameters for the AIT401 signal), which can be a crucial factor for the timely detection of cyber-attacks in real-time tasks. The performance of cyber-attacks detection algorithms based on the generated models was tested on five signals from two publicly available datasets. Using algorithms based on the student models, all 18 attacks were detected with no false positives, which was also the case with teacher models (models before distillation). In this way, it is shown that models with a very small number of parameters (e.g., 87 parameters in the case of PIT501 signal) can be used as successfully as models with a few thousand parameters.

In future work, we will implement the IDS based on the distilled model on the local controller within DisEPP to test its performance in real-world conditions. Further research will also include the application of the knowledge distillation technique to models for cyber-attacks detection algorithms on sequences of two-dimensional signals.

ACKNOWLEDGEMENT

This research was supported by the Ministry of Science, Technological Development and Innovations of the Serbian Government under the contract No. 451-03-47/2023-01/200105.

REFERENCES

[1] H. Kagermann, J. Helbig, A. Hellinger, W. Wahlster: Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0, Forschungsunion, 2013.

[2] H. S. Sánchez, D. Rotondo, T. Escobet, V. Puig, J. Quevedo: Bibliographical review on cyber attacks from a control oriented perspective, Annual Reviews in Control, Vol. 48, pp. 103-128, 2019.

[3] M. Macas, C. Wu, W. Fuertes: A survey on deep learning for cybersecurity: Progress, challenges, and opportunities, Computer Networks, Vol. 212, art. 109032, 2022.

[4] M. M. H. Shuvo, S. K. Islam, J. Cheng, B. I. Morshed: Efficient acceleration of deep learning inference on resource-constrained edge devices: A review, Proceedings of the IEEE, 2022.

[5] M. Zhu, T. Zhang, Z. Gu, Y. Xie: Sparse tensor core: Algorithm and hardware co-design for vector-wise sparse neural networks on modern gpus, in: *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, 12-16.10.2019, Columbus, USA, pp. 359-371.

[6] K. Ullrich, E. Meeds, M. Welling: Soft weight-sharing for neural network compression. arXiv preprint arXiv:1702.04008, 2017.

[7] Y. Cheng, D. Wang, P. Zhou, T. Zhang: Model compression and acceleration for deep neural networks: The principles, progress, and challenges, IEEE Signal Processing Magazine, Vol. 35, No. 1, pp. 126-136, 2018.

[8] G. Hinton, O. Vinyals, J. Dean: Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531, 2015.

[9] J. Gou, B. Yu, S. J. Maybank, D. Tao: Knowledge distillation: A survey, International Journal of Computer Vision, Vol. 129, pp. 1789-1819, 2021.

[10] D. Nedeljkovic, Z. Jakovljevic: CNN based method for the development of cyber-attacks detection algorithms in industrial control systems, Computers & Security, Vol. 114, art. 102585, 2022.

[11] J. Goh, S. Adepu, K. N. Junejo, A. Mathur: A dataset to support research in the design of secure water treatment systems, in: *Critical Information Infrastructures Security: 11th International Conference*, 10-12.10.2016, Paris, France, pp. 88-99.

[12] NXP Semiconductors N.V., LPC1769/68/66/65/64/63 32-bit ARM Cortex-M3 microcontroller, available at: https://www.nxp.com/docs/en/data-sheet/LPC1769_68_67_66_65_64_63.pdf, accessed: 02.10.2023.

[13] Microchip Technology Inc., MRF24J40MA 2.4 GHz IEEE Std. 802.15.4TM RF Transceiver Module, available at: http://ww1.microchip.com/downloads/en/DeviceDoc/70329b.pdf, accessed: 02.10.2023.

[14] D. Nedeljkovic, Z. Jakovljevic: New datasets obtained from experimental installations with centralized control – v2.0, 2021, available at: https://zenodo.org/record/5514351, accessed: 02.10.2023.