# PMU Spoof Detection via Image Classification Methodology against Repeated Value Attacks by using Deep Learning

Alvin Huseinovic
*Dept. of Computer Science*
*Faculty of Electrical Engineering - University of Sarajevo*
ahuseinovic@etf.unsa.ba

Yusuf Korkmaz
*Dept. of Computer Science*
*University of Michigan - Flint*
ykorkmaz@umich.edu

Halil Bisgin
*Dept. of Computer Science*
*University of Michigan - Flint*
bisgin@umich.edu

Saša Mrdović
*Dept. of Computer Science*
*Faculty of Electrical Engineering - University of Sarajevo*
sasa.mrdovic@etf.unsa.ba

Suleyman Uludag
*Dept. of Computer Science*
*University of Michigan - Flint*
uludag@umich.edu

***Abstract*—Various devices and monitoring systems have been developed and deployed in order to monitor the power grid. Indeed, several real-world cyberattacks on power grid systems have been publicly reported. For the transmission and distribution, Phasor Measurement Units (PMUs) constitute the main sensing equipment of the overall wide area monitoring and situational awareness systems by collecting high-resolution data and sending them to Phasor Data Concentrators (PDCs). In this paper, we consider data spoofing attacks against PMU networks. The data between PMUs and PDC(s) are sent through the legacy networks, which are subject to many attack scenarios under with no, or inadequate, countermeasures in protocols, such as IEEE 37.118-2. We consider one potential attack, where an adversary may simply keep injecting a repeated measurement through a compromised PMU to disrupt the monitoring system. This attack is referred to as a Repeated Last Value (RLV) attack. We develop and evaluate countermeasures against RLV attacks using a 2D Convolutional Neural Network (CNN)-based approach, which operates in frames for each second mimicking images, in order to avoid the computational overhead of the classical sample-based classification algorithms, such as SVM. Further, we take this frame-based approach and use it with Support Vector Machine (SVM) for performance evaluation. Our preliminary results show that frame-based CNN as well as SVM provide promising results for RLV attacks while the efficacy of CNN over SVM frame becomes more pronounced as the attack intensity increases.**

***Index Terms*—CNN, Deep Learning, SVM, PMU, PMU Spoofing, PMU forged data, Repeat Last Value Attack.**

## I. Introduction

Smart Grid is an umbrella term used to refer to the efforts to transform, upgrade, and enhance the power grid though digital computing, communications, and industrial control systems and technologies [1]–[3]. A key element of the SG effort is in the incorporation of the bidirectional flow of power (for distributed and renewable energy sources) as well as the two-way communications and control capabilities. With all these efforts, a critical need emerges to address a variety of security and privacy related challenges [4]–[11].

Cybersecurity, as a consequence, becomes an indispensable component and a key enabler for the successful transformation from the electric power grid of yesterday into the SG of the future. Power grid infrastructure has become an attractive target [12] with lethal and vital economic and social consequences by means of disruption to electricity delivery [13]. World Economic Forum's 2018 report [14] emphasizes the increasing cyberattacks on the critical and strategic infrastructure that may result in disrupting the society. It is obvious that the power grid falls into the aforementioned definition of critical infrastructure [15]. There is definitely an imperative to implement and adopt cybersecurity technology, both within the SG and beyond.

An important enabler of the Smart Grid initiatives is the enhanced use of sensing and measurement capabilities. Phasor Measurement Units (PMUs) are the advanced, accurate, and synchronized measurement devices to take the situational awareness to a new level. While the traditional Supervisory Control And Data Acquisition (SCADA) measurements are taken every 2-4 seconds, PMU reports them 30-120 times per second with GPS time stamps. As compared to SCADA, PMU-enabled conceptual model of wide-area monitoring, protection, and control subsystem is illustrated [11] in Figure 1.

PMUs transmit data to the Phasor Data Concentrator (PDC) by using IEEE 37.118-2 synchrophasor protocol. Data received at PDC is then used for state estimation or historical analysis. It is relatively more recently recognized that the PMU data, especially over the IEEE 37.118-2 protocol, which has no security mechanisms [17], has many vulnerabilities [18], [19], such as transport layer attacks [20], data tampering attacks [21], etc.

In this paper, we consider a PMU data collection network where the threat environment assumes a compromised PMU injection spoofed data into the network to corrupt, or disrupt or confuse the state estimation and the situational awareness

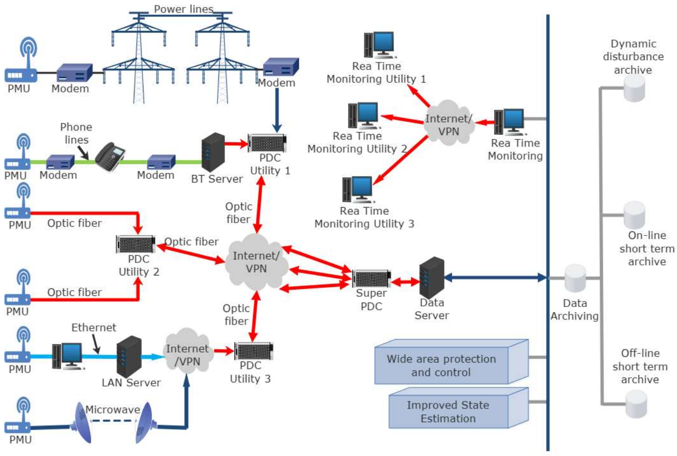

Fig. 1. A conceptual framework for a wide-area monitoring, protection, and control (WAMPAC) system for the Smart Grid made possible by PMUs [16].

of the overall power grid. We focus on a specific attack, repeat last value (RLV) attack, both plain and stealth strategies, whose detailed definition are given in Section III and provide deep learning-based countermeasure together with a preliminary results of its performance evaluation.

The rest of the paper is organized as follows: Section II consists of related work towards spoofing cyber attacks in the Smart Grid including but not limited to machine learning approaches. Section III focuses on the PMU data spoofing, dataset description and attack scenarios including the description of the Repeat Last Value (RLV) attack as presented in [22]. The Section IV introduces the machine learning methodologies for detecting the PMU spoofing attacks in terms of repeat last value scheme together with the countermeasures using frame-based 2D Convolutional Neural Network (CNN) approach and classical and frame-based SVM. Section V includes the experimentation setup together with the simulations and the discussion of these results. Concluding remarks and future work ideas are given in Section VI.

## II. Related Work

The most common spoofing attacks on PMU data include repeat last value attack [22], [23], time dilation attack [22]–[27], mirroring attack [22], [23], [26], play back attack [24], [25], data drop attack [24], [25] polynomial fit attack [26], and general false data injections attack [24], [28], [29].

From the countermeasure perspective, there are a wide variety of approaches both for intentional attacks by adversaries and unintentional faults in the system; SVM at the sample level is the most commonly employed one [22], [23] [22], [23] use SVM and Artificial Neural Network (ANN) to detect anomalies relying on the highly-correlated inter-PMU and intra-PMU parameters. However, it is not clear from the papers if the correlation values or PMU raw measurements of the most correlated features are used in the algorithms. [26], [27] focuses on SVM.

In [24], authors artificially create their datasets and use Recurrent Neural Networks (RNN) and Long short-term memory neural network (LSTM) to detect False Data Injection Attacks (FDI) against PMU based state estimators. Same authors [25] use different approach by introducing Symbolic Aggregation Approximation in data preprocessing phase. For detection of these attacks text mining using Bag of Pattern and Multivariate Bag of Pattern is used and compared. The feature extraction is obtained through Principal Component Analysis (PCA).

In [28] authors detect FDI attacks using Rule Based Autoregressive Moving Average (ARMA) and Autoregressive Integrated Moving Average (ARIMA) applied on calculations based on Kirchoffs laws.

In [30] authors use different types of clustering to identify different types of fault events that can occur in the power grid. The selected fault events are divided into: single-line-to-ground faults, line-to-line faults, three-phase faults and no-fault data. They use two different clustering approaches. The first is time series clustering that uses hierarchical clustering for which they claim to be the most appropriate in case of time series data. The other clustering method is instantaneous clustering that uses is based on k-means and Density Based Spatial Clustering of Applications with Noise (DBSCAN).

In [31] authors use k-nearest neighbor (KNN), binary SVM, multi-SVM and Decision Trees (DT) to detect events based either on event zone or event type. This approach requires field knowledge in order to correctly apply event labeling on the PMU data.

In [32] authors are completely agnostic to the data being transmitted between PMUs and PDC. Instead of checking the validity of the data they use k-means clustering to separate the network traffic not typical for PMU to PDC communication.

In [33] authors consider single-phase, two-phase and three phase types of faults. They also consider short circuits and ground for each individual phase. They applied Linear Discriminant Analysis (LDA), kNN, SVM and ANN machine learning approaches on simulated IEEE 123-bus distribution system.

In [29] phasor measurement unit data attacks (PMUDA) by using different machine learning algorithms that can be used for supervised and semi-supervised learning. The supervised machine learning algorithms are multi-layerperceptron (MLP), SVM, KNN, AdaBoost+, C4.5 DT and XGBoost. The semi-supervised learning techniques are deep autoencoders (DA) and one-class SVM (OC-SVM).

In [34] authors use Generative Adversarial Networks (GAN) and Neural ordinary differential equations (NODE) to artificially generate PMU data events. They observe three event types: Bus Fault, Line Tripping and Load shedding. The simulation environment includes a 10-machine IEEE 39 bus system. To classify events they use PCA and Discrete Wavelet Transformation(DWT) SVM kernels.

In [35] authors use SVM with online learning in order to predict short-term voltage instability on IEEE 39 bus based network.

## III. PMU Data Spoofing

### A. Threat Model

For a our threat model, we adopt the three attack vectors (spoofing techniques) described in [22], [23] based on the more general spoofing attacks from [26], [27]. These attacks are designed to avoid easy detection with the following characteristics: (a) reasonable (based on the historically valid data), (b) Continuity (falsified data should not create a discontinuity and consistent with the preceding and the following measurements), and (c) locality (should only based on the local knowledge at the PMU). The three attacks are: (1) Repeat Last Value Attack (RLV) [22], [23], where the adversary selects the last valid data in a sequence of $n$ measurements and keeps repeating it for the next $n$ values, as shown in Figure 2, (2) Time Dilation Attack (TDA) [22], [23], [26], where the adversary replaces every other value with its preceding measurement, and (3) Mirroring Attack (MA) [22], [23], [26], where the attacker takes a set of $n$ values and reverses the order of the data, swaps first with the last, second with the penultimate, etc. Our focus in this study is the RLV and we leave the other two attacks to a future work.

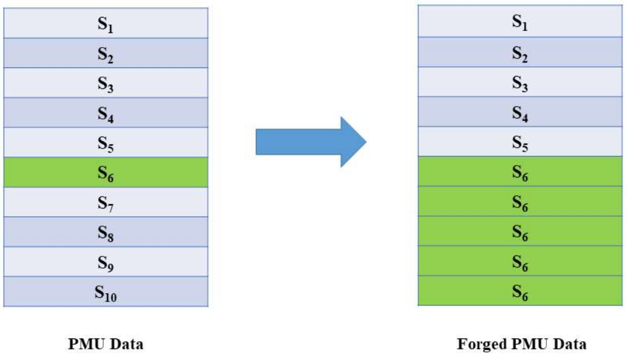

Fig. 2. Repeat Last Value Attack: The original sequence of the PMU data on the left is forged on the right by replacing $S_5$ through $S_{10}$ with the repeated $S_6$ value.

### B. Dataset and Variables

We make use of the EPFL dataset [36], [37], collected over a transmission network with 7 PMUs. Every hour 180,000 rows per PMU are being collected, giving the total of 4,320,000 samples per day for every PMU.

We make use of 9 common measurements across all measurement; i.e. latency, frequency, ROCOF (rate of change of frequency), magnitude of A, B, and C phase voltage, phase A, B, and C voltage angle.

### C. Attack Scenarios and Data Preprocessing

In this work, we use 24 hours of measurements and forge only one PMU under two different attack scenarios with RLV. Regardless of the scenario attack we design, we select the first 6 hours for training and 18 hours for testing as illustrated in Figure 3. To forge the data, after selecting the PMU and its 24-hour consecutive measurements, we first determine hourly attack counts ($a$) which range from 500 through 3,000 with a step size of 500. Then, we randomly select time points within each hour to spread attacks with minimum ($l$) and maximum number ($u$) of spoofed signals. Since our sampling rate ($S$) per second is 50, our study design includes two attack scenarios, 50+ and 10-40, to simulate attacks exceeding and under one second, respectively. While the former scenario has $l = 50$ and $u = a$ for random number generation of the attack count, the latter guarantees more stealth attacks under a second by forcing $l = 10$ and $u = 40$. In both scenarios, we observe varying number of RLV attacks in a second, but 50+ scenario also produces attacks that cover all 50 samples in a second or more.

After generating the attacks, we label those samples as forged (positive) and others as authentic (negative) and then merge it with the rest of the 6 PMUs' data at the PDC, yielding 64 columns and 4,320,000 rows. In all attack simulations the ratio of the forged sample size remain disproportionately small ranging between 0.4% and 1.8% depending on the attack count per hour, which introduces an additional challenge associated with the class imbalance problem. During this process, we also perform z-score normalization on all datasets.

## IV. RLV Attack Detection Methodology

### A. Frame Approach

While the ideal spoof detection would be at the sample level, its computational overhead for our high-volume PMU traffic is overwhelming for detection. Thus, we transform data into a coarser granularity representation through data slices for every second, which we call *frames* to imitate a sequence of images as shown in Figure 3 to use image classification algorithms. In particular, we constructed frames for each sample with a size of $P \times N$, where $P$ and $N$ represent number of PMUs and variables for each PMU, respectively. Depending on the sampling rate, $S$, per second, we obtained $T$ frames (images representations) with a size of $S \times \underbrace{P \times N}$. In our particular case, where $S = 50$, $P = 7$, and $N = 9$ due to the columns in common across seven PMUs, we ended up with 86,400 frames (images) of size 50x63 over 24 hours. Please note that while

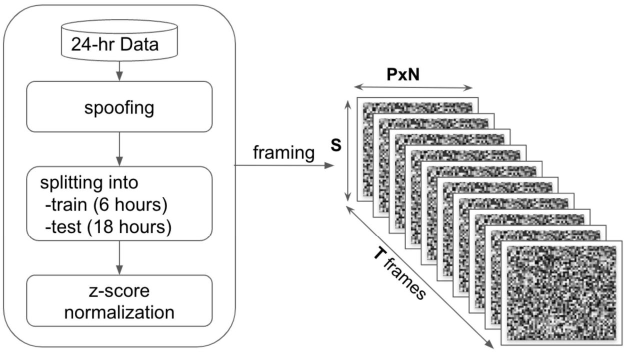

Fig. 3. Data spoofing, preprocessing, and transformation. Each pixel corresponds to a measurement.

we sacrifice some good measurements at the sample level as bad at the frame level the statistical nature of the time series data would not be impacted by this *too* conservative approach to bring the computational complexity to acceptable levels for scalability reasons.

Due to the random and stealth nature of the attacks, some frames had no positive cases whereas others had at least one forged sample in them. While transforming our data, we labeled those frames that had at least one spoofed sample as forged (positive) frames. This led to a reorganization of class labels at the frame level, but the class imbalance problem problem continued to exist with a small ratio of the positive cases, which ranged from 0.39% to 4.88% depending on the hourly attack count and $l$ and $u$ values. As 50+ scenario allowed attacks covering all 50 samples in a second, we observed a high ratio (48%-84%) of such frames among the forged ones. For instance, for hourly attack count of 500, 159 out of 334 forged frames turned out to have its samples all spoofed.

### *B. Machine Learning Algorithms*

In this work, we are able to take advantage of Convolutional Neural Network (CNN), a deep learning (DL) approach, as a result of data transformation into frames, in addition to the more classical and widely used machine learning algorithm, SVM.

*1) 2D CNN:* CNNs are one of the DL neural networks that consist of multiple layers [38] [39] [40] and are mostly used for image classification problems [41]. One of the advantages of CNN over other conventional image classification methods is that CNN does not only perform predictions, but also learns and extracts features through their convolutional layers coupled with pooling layers [42], which can be stacked similar to the generic architecture as shown in Figure 4 [43].

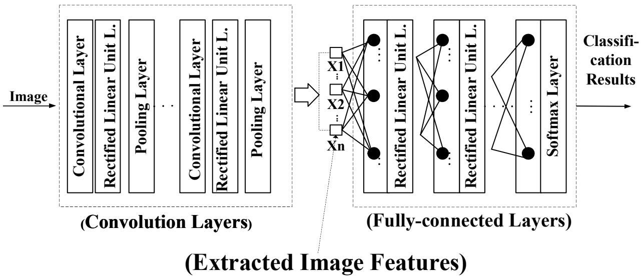

Fig. 4. Generic CNN Structure.

Our network architecture consists of five layers. The first layer is a 2D convolutional layer with a kernel size of 3x3 and 256 filters. The activation function in this layer is rectified linear unit (RLU). The second layer is a pooling layer with a 2x2 pool size. Next two layers are flattening and dropout layers. The drop rate in the dropout layer is set to 0.5. The output is a dense layer with sigmoid activation function. Training is performed over 25 epochs with a batch size of 512.

*2) Support Vector Machine:* Support Vector Machines (SVMs) are another set of well-known classification methods, which are capable of handling both linear and non-linear data through kernel functions that are responsible for data transformation. SVM models search for the best hyperplane with a maximum margin in $N$-dimensional space, where $N$ is the number of features in a dataset, to achieve the best separation [44].

In the current work, we employed a widely-used kernel function [45], radial basis function (RBF), as shown in Eq. 1

$$K(x, x') = exp(-\gamma||x - x'||^2) \tag{1}$$

on any two samples, $x$ and $x'$, due to its overall superiority observed in empirical studies over other kernel functions [46]. However, this necessitated a search for the best $\gamma$ for RBF in addition to regularization parameter, $C$, for SVM. Therefore, we performed a grid search on the training data to find the best $\gamma - C$ pair, which we used on the test set as well.

Since SVM only needed a set of features to operate on, we first attempted to use it on the data right after the spoofing without frames. We then introduced frame-based SVM, $SVM_f$, which worked on image-like frames to achieve a fair comparison of CNN and SVM. In the latter case, however, we summarized each frame with the standard deviations of PMU variables which formed our features for SVM, whereas CNN relied on its implicit features extracted through convolutional layers.

## V. Results and Discussion

### *A. Simulation Setup*

The simulations are run on a server with 128GB RAM and two Intel Xeon Silver 4208 processors running at base clock of 2.1GHz each with 11MB of level 3 cache. Each processor has 8 cores and 16 threads which gives a total of 32 simultaneous threads running on the training and testing process. The storage consists has 8TB of SSD storage.

### *B. Simulation Results*

For each dataset, we ran CNN 30 times for statistical significance and calculated the average accuracy with standard deviation to observe its robustness as DL models could be fluctuating due to some internal factors such as weight initialization. On the other hand, we performed only one SVM run because of its deterministic nature. For both algorithms, we tuned their algorithm-specific parameters in the training stage to find the optimal models.

Training an SVM model without the frame approach on a such big data for the purpose of sample-based spoof detection took a considerable amount if time, in line with the intuitive expectation we mentioned earlier. Even for the smaller subsets of the data (1hr - 5hr in length), run time was too long (up to 50 hours for some cases) for an efficient spoof detection task. Even though we observed high accuracy and true negative rates (TNRs) around 99% for some runs, this was mostly because negative cases were extremely dominant in the data which led correct classification of negative cases and increased accuracy. However, for a spoof detection framework this would be misleading and true positives rate (TPR), which carries more importance towards flagging a spoofed reading, needed further inspection. We found that TPRs were as low as 10% for training and 0% for test. The failure of the algorithm on these subsets might be due to the nature of the threat type, RLV, as SVM sees the same feature values in an authentic reading and multiple forged ones at the same time, which may make it harder to separate them.

Next, we took the frame-based approach which not only shrank the size of the data, but also let us detect a spoofed time window, which was a second in our case. For both CNN and $SVM_f$, we built classification models on 6 hours of PMU readings, and tested for the remaining 18 hours to find out the ability of our models to predict for long hours of readings once they were developed. Similar to earlier SVM runs, both algorithms gave high accuracy values around 0.99 as shown in Table I. However, this phenomenon was again mostly because of the class imbalance problem which let algorithms correctly classify overwhelmingly many negative cases as depicted in Figures 5 and 6. Therefore, we needed a deeper look at the performances in terms of TPRs here as well.

TABLE I
ACCURACY VALUES FOR FRAME-BASED APPROACHES

| | 50+ | | | | 10-40 | | | |
|---|---|---|---|---|---|---|---|---|
| | $SVM_f$ | | CNN | | $SVM_f$ | | CNN | |
| a/hr | Train | Test | Train | Test | Train | Test | Train | Test |
| 500 | 0.998 | 0.996 | 0.999 | 0.98 | 0.999 | 0.89 | 0.996 | 0.97 |
| 1000 | 0.998 | 0.998 | 0.998 | 0.98 | 0.997 | 0.96 | 0.994 | 0.95 |
| 1500 | 0.998 | 0.998 | 0.998 | 0.995 | 0.994 | 0.93 | 0.993 | 0.90 |
| 2000 | 0.998 | 0.997 | 0.999 | 0.74 | 0.992 | 0.92 | 0.991 | 0.87 |
| 2500 | 0.998 | 0.997 | 0.999 | 0.96 | 0.990 | 0.90 | 0.989 | 0.91 |
| 3000 | 0.998 | 0.998 | 0.999 | 0.98 | 0.987 | 0.89 | 0.986 | 0.84 |

In 50+ attack design, we observed higher TPRs for $SVM_f$ than that of CNN except for the 2000 attacks/hour as shown in Figure 5. $SVM_f$ had an increasing trend for TPR, whereas CNN showed some fluctuations, which we also represented with error bars at each attack count. As we increased the attack counts, both algorithms converged around 80% accuracy, but it was the standard deviation of each frame which turned out to be zero giving a slight advantage to $SVM_f$ over CNN in terms of more informative underlying features.

In order to avoid zero standard deviations and make the attacks more stealth, we ran both algorithms on 10-40 attack scenario. In this case, CNN continued to learn and extract features while SVM still relied on standard deviations as features. As shown in Figure 6, both methods performed poorly for very small number of attacks, but we noticed a huge drop in $SVM_f$'s TPR performance which did not exceed 36% as attack counts increased. On the other hand, CNN mostly showed a consistently increasing performance in terms of TPR that reached 76%.

## VI. Conclusion and Future Work

The preliminary result showed that CNN and frame-based SVM algorithm can be used to detect RLV attacks. It is also evident that CNN outperforms the frame-based SVM detection performance when the attack size becomes larger. In the future work we plan to apply the same approaches on other types of the attacks including, but not limited to TDA and MA. The main advantage is that for these type of the attack there is no need to perform complex state estimation functions to detect if attack occurred. Furthermore the detailed information and theory that stands behind the power grid is not needed. These approaches can definitely with modifications be applied to different fields where similar FDI attacks may occur. We also showed that time-series component is completely not considered as important for attack detection. One potential research direction is to explore the time-series component using CNN and LSTM neural networks. In our work we spoofed only one of the PMUs data and labeled the whole row along with correct readings of other PMUs as spoofed. One potential direction is to investigate how CNN and frame-based SVM perform when multiple PMUs data are spoofed.

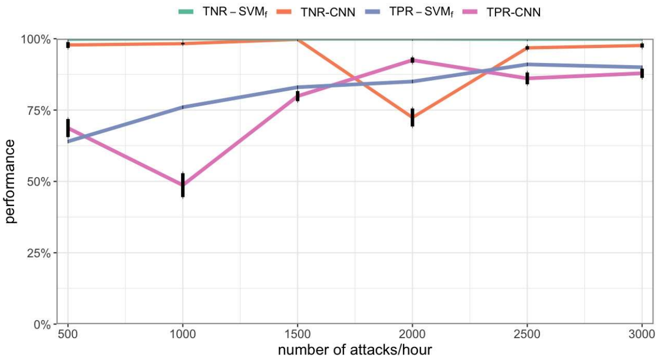

Fig. 5. Performances of $SVM_f$ vs. CNN in 50+ attack scenario.

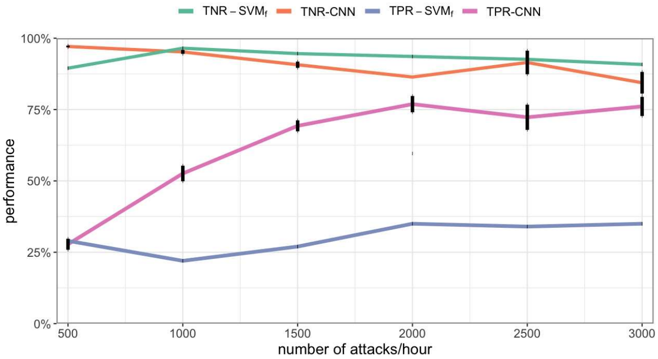

Fig. 6. Performances of $SVM_f$ vs. CNN in 10-40 attack scenario.

## References

[1] H. Farhangi, "the path of the smart grid," *IEEE Power and Energy Magazine*.

[2] M. L. Tuballa and M. L. Abundo, "A review of the development of smart grid technologies," *Renewable and Sustainable Energy Reviews*, vol. 59, pp. 710–725, 2016.

[3] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid—the new and improved power grid: A survey," *IEEE communications surveys & tutorials*, vol. 14, no. 4, pp. 944–980, 2011.

[4] S. Systems, P. McDaniel, and S. McLaughlin, "Security and Privacy Challenges in the Smart Grid," *Security & Privacy, IEEE*, vol. 7, no. 3, pp. 75–77, 2009.

[5] H. Khurana, M. Hadley, N. Lu, and D. A. Frincke, "Smart-grid security issues," *IEEE Security and Privacy*, vol. 8, pp. 81–85, 2010.

[6] A. R. Metke and R. L. Ekl, "Security Technology for Smart Grid Networks," *IEEE Trans. on Smart Grid*, no. 1, pp. 99–107, jun.

[7] W. Wang and Z. Lu, "Cyber security in the Smart Grid: Survey and challenges," *Computer Networks*, no. 5, pp. 1344–1371.

[8] J. Liu, Y. Xiao, S. Li, W. Liang, and C. L. P. Chen, "Cyber Security and Privacy Issues in Smart Grids," *Communications Surveys Tutorials, IEEE*, vol. 14, no. 4, pp. 981–997, 2012.

[9] X. Li, X. Liang, R. Lu, X. Shen, X. Lin, and H. Zhu, "Securing smart grid: cyber attacks, countermeasures, and challenges," *Communications Magazine, IEEE*, vol. 50, no. 8, pp. 38–45, 2012.

[10] Y. Xiao, *Security and Privacy in Smart Grids*. Taylor & Francis, 2013. [Online]. Available: http://books.google.com/books?id=QQ2oY0IrRM8C

[11] A. Huseinović, S. Mrdović, K. Bicakci, and S. Uludag, "A survey of denial-of-service attacks and solutions in the smart grid," *IEEE Access*, vol. 8, pp. 177 447–177 470, 2020.

[12] "Surviving a Catastrophic Power Outage," The President's National Infrastructure Advisory Council (NIAC), Tech. Rep., Dec 2018. [Online]. Available: https://www.dhs.gov/sites/default/files/publications/NIAC%20Catastrophic%20Power%20Outage%20Study_508%20FINAL.pdf

[13] N. Kshetri and J. Voas, "Hacking Power Grids: A Current Problem," *Computer*, no. 12, pp. 91–95, dec.

[14] "The Global Risks Report 2018 13th Edition," World Economic Forum, Tech. Rep., 2018. [Online]. Available: http://wef.ch/risks2018

[15] M. P. Barrett, "Framework for Improving Critical Infrastructure Cybersecurity Version 1.1," National Institute of Standards and Technology (NIST), Tech. Rep., apr 2018. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf

[16] V. Terzija, G. Valverde, P. Regulski, V. Madani, J. Fitch, S. Skok, M. M. Begovic, and A. Phadke, "Wide-Area Monitoring, Protection, and Control of Future Electric Power Networks," *Proceedings of the IEEE*, no. 1, pp. 80–93, jan.

[17] S. M. S. Hussain, S. M. Farooq, and T. S. Ustun, "A security mechanism for ieee c37.118.2 pmu communication," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 1, pp. 1053–1061, 2022.

[18] C. Tu, X. He, X. Liu, and P. Li, "Cyber-attacks in pmu-based power network and countermeasures," *IEEE Access*, vol. 6, pp. 65 594–65 603, 2018.

[19] R. Khan, K. Mclaughlin, D. Laverty, and S. Sezer, "Design and implementation of security gateway for synchrophasor based real-time control and monitoring in smart grid," *IEEE Access*, vol. 5, pp. 11 626–11 644, 2017.

[20] Y. Wang, T. T. Gamage, and C. H. Hauser, "Security implications of transport layer protocols in power grid synchrophasor data communication," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 807–816, 2016.

[21] M. N. Aman, K. Javed, B. Sikdar, and K. C. Chua, "Detecting data tampering attacks in synchrophasor networks using time hopping," in *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*. IEEE, 2016, pp. 1–6.

[22] J. Jiang, X. Liu, S. Wallace, E. Cotilla-Sanchez, R. Bass, and X. Zhao, "Defending against adversarial attacks in transmission- and distribution-level pmu data," 2020.

[23] J. Jiang, "Defending against adversarial attacks in electric power systems: A machine learning approach," *Washington State University*, no. 1, 2019.

[24] "Packet-data anomaly detection in pmu-based state estimator using convolutional neural network," *International Journal of Electrical Power & Energy Systems*, vol. 107, pp. 690 – 702, 2019.

[25] R. Ma, S. Basumallik, and S. Eftekharnejad, "A pmu-based data-driven approach for classifying power system events considering cyberattacks," *IEEE Systems Journal*, vol. 14, no. 3, pp. 3558–3569, 2020.

[26] J. Landford, R. Meier, R. Barella, S. Wallace, X. Zhao, E. Cotilla-Sanchez, and R. B. Bass, "Fast sequence component analysis for attack detection in smart grid," in *2016 5th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*, 2016, pp. 1–8.

[27] X. Liu, S. Wallace, X. Zhao, E. Cotilla-Sanchez, and R. B. Bass, "Episodic detection of spoofed data in synchrophasor measurement streams," in *2019 Tenth International Green and Sustainable Computing Conference (IGSC)*, 2019, pp. 1–8.

[28] B. Chen, S. i. Yim, H. Kim, A. Kondabathini, and R. Nuqui, "Cybersecurity of wide area monitoring, protection, and control systems for hvdc applications," *IEEE Transactions on Power Systems*, vol. 36, no. 1, pp. 592–602, 2021.

[29] J. Wang, D. Shi, Y. Li, J. Chen, H. Ding, and X. Duan, "Distributed framework for detecting pmu data manipulation attacks with deep autoencoders," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4401–4410, 2019.

[30] E. Klinginsmith, R. Barella, X. Zhao, and S. Wallace, "Unsupervised clustering on pmu data for event characterization on smart grid," in *2016 5th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*, 2016, pp. 1–8.

[31] A. Shahsavari, M. Farajollahi, E. M. Stewart, E. Cortez, and H. Mohsenian-Rad, "Situational awareness in distribution grid using micro-pmu data: A machine learning approach," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6167–6177, 2019.

[32] P. Donner, A. S. Leger, and R. Blaine, "Unsupervised machine learning for anomaly detection in synchrophasor network traffic," in *2019 North American Power Symposium (NAPS)*, 2019, pp. 1–6.

[33] F. L. Grando, A. E. Lazzaretti, M. Moreto, and H. S. Lopes, "Fault classification in power distribution systems using pmu data and machine learning," in *2019 20th International Conference on Intelligent System Application to Power Systems (ISAP)*, 2019, pp. 1–6.

[34] X. Zheng, B. Wang, D. Kalathil, and L. Xie, "Generative adversarial networks-based synthetic pmu data creation for improved event classification," *IEEE Open Access Journal of Power and Energy*, vol. 8, pp. 68–76, 2021.

[35] "Pmu-based voltage stability prediction using least square support vector machine with online learning," *Electric Power Systems Research*, vol. 160, pp. 234–242, 2018.

[36] EPFL. Epfl campus pmu dataset. [Online]. Available: https://bigdata.seas.gwu.edu/data-set-20-epfl-campus-pmu-data-set/

[37] M. Pignati, M. Popovic, S. Barreto, R. Cherkaoui, G. Dario Flores, J.-Y. Le Boudec, M. Mohiuddin, M. Paolone, P. Romano, S. Sarri, T. Tesfay, D.-C. Tomozei, and L. Zanni, "Real-time state estimation of the epfl-campus medium-voltage grid by using pmus," in *2015 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, 2015, pp. 1–5.

[38] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*, 2017, pp. 1–6.

[39] U. Michelucci, *Advanced Applied Deep Learning*. Apress, 2019. [Online]. Available: https://doi.org/10.1007/978-1-4842-4976-5

[40] F. Chollet, *Deep learning with Python*. Shelter Island, NY: Manning Publications Co, 2021.

[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[42] C. Zheng, D.-W. Sun, and L. Zheng, "Recent developments and applications of image features for food quality evaluation and inspection–a review," *Trends in Food Science & Technology*, vol. 17, no. 12, pp. 642–655, 2006.

[43] D. Nguyen, H. Yoon, T. Pham, and K. Park, "Spoof detection for finger-vein recognition system using nir camera," *Sensors*, vol. 17, p. 2261, 10 2017.

[44] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[45] S. S. Keerthi and C.-J. Lin, "Asymptotic behaviors of support vector machines with gaussian kernel," *Neural computation*, vol. 15, no. 7, pp. 1667–1689, 2003.

[46] C. Ding, L.-F. Yuan, S.-H. Guo, H. Lin, and W. Chen, "Identification of mycobacterial membrane proteins and their types using over-represented tripeptide compositions," *Journal of proteomics*, vol. 77, pp. 321–328, 2012.