The Triumvirate of Bespoke Diverse Hybridized Activation Functions, Adaptive Momentum, and Enhanced Entropic Wavelet Energy Spectrum Discernment for Higher Efficacy Detection of Artificial Intelligence-centric Attacks

Steve Chan
Decision Engineering Analysis Laboratory, VTIRL, VT
Orlando, USA
e-mail: schan@dengineering.org

***Abstract*—Artificial Intelligence-centric Attacks (AIA) involving False Data Injection and False Command Injection have become increasingly sophisticated and have also involved Metamorphic Malware (MM), which leverages numerous transformation techniques to avoid detection. Accordingly, the use of AI defender systems requires continuous learning so as to decrease the advantage held by the high cycles of adaptation of attackers. This paper explored Exponential Linear Unit (ELU) Mish as a hybridized activation function that could avoid the cessation of learning when a substantive portion of the Neural Network (NN) neurons output zero and weights are no longer updated. The involved Robust Convex Relaxation (RCR)-based Convolutional NN capitalized upon its architecture by utilizing a Nonlinear Conjugate Gradient and Nesterov's Accelerated Gradient approach to Adaptive Momentum (AdaM) so as to mitigate against oscillation, facilitate convergence, and achieve a more optimal global minimum. This more resilient foundational architecture could then better support a more accurate and expedient Entropic [Wavelet Energy Spectrum] Discernment (ED). Central to this was the use of Second-Order Cone Programming Relaxations to address certain nonconvex subproblems, which were inadvertently spawned via the utilized RCR framework. The described triumvirate approach constitutes the beginnings of a potential mitigation pathway, which exhibited some promise during the preliminary experimentation.**

Keywords-Artificial Intelligence Attack; Cyber; Smart Grid; Operational Technology; Adaptive Momentum; Second-Order Cone Programming Relaxations; Robust Convex Relaxations.

I. INTRODUCTION

Cyber Physical Power Systems (CPPS) are envisioned to enable Smart Grid (SG) technologies with more optimal monitoring and control. Numerous SG enablers have emerged at the convergence of Information Technology (IT) and Operational Technology (OT), and IT/OT engineers have increasingly utilized REpresentational State Transfer (REST) Application Programming Interfaces (APIs) to operationalize desired SG capabilities. However, the Open Worldwide Application Security Project (OWASP) notes the use of deprecated API versions as well as exposed debug endpoints (API9:2023) and potentially compromised third-party APIs (API10:2023). This attack surface area at the IT/OT nexus is of concern to many. Security firms have noted that while advisories often contain a patch to ameliorate the cited vulnerability, oftentimes, it is difficult to implement due to the downtime risk for the involved OT system. Contemporaneously, the World Economic Forum's (WEF) Global Risk Report notes that attacks on critical infrastructure operations (e.g., OT) are among the top five "currently manifesting risks" [1], and McKinsey & Company notes that these OT cyberattacks have particularly profound negative effects (e.g., outages, explosions, etc.) [2]. Among other attacks, polymorphic and Metamorphic Malware (MM) have beset CPPS, and advances in the area of mitigation have remained fairly nascent, particularly if patching is not an option. Yet, perhaps, of even greater concern for CPPS are Artificial Intelligence Attacks (AIA), which are designed to deceive AI-centric defense systems, such as via False Data Injection (FDI), False Command Injection (FCI), and other forms of insidious attack vectors.

While CPPS/SG defenders have increasingly looked to AI to defend at machine speed, these defending systems may be at a disadvantage due to the adversarial cycles of adaptation and may not yet be sufficiently robust against adversarial AIA. Contemporary defense research tends to center upon improving Machine Learning (ML) approaches to attain higher detection accuracy and computational performance, but efforts to mitigate against AIA remain fairly nascent. This paper delineates a potential mitigation pathway, and the paper is structured as follows. Section I provided a backdrop and introduces the problem space. Section II provides the background by way of describing the operating environment as well as the state of the challenge. Section III provides some theoretical foundations and the posited/utilized approach. Section IV delineates some preliminary experimental forays regarding the posited AIA mitigation approach. Section V concludes with some preliminary reflections, puts forth envisioned future work, and the acknowledgements close the paper.

II. BACKGROUND INFORMATION

MM mitigation efforts have already illuminated the sophistication of modern attack vectors. For example, crypters (i.e., a paradigm, wherein the use of obfuscation and/or encryption is at play) and protectors (i.e., a paradigm, wherein a hybridization of packing — self-extracting archives that unpack in memory upon execution — and encrypting is utilized) are becoming increasingly successful at obfuscating their malicious intent from detectors. Likewise, AIA endeavor to obscure their intent from

 ISBN: 978-1-68558-089-6

Endpoint Detection and Response (EDR) applications and CPPS/SG Detectors (CSD) alike by leveraging Generative Adversarial Networks (GANs), among other approaches, to spawn/facilitate FDI, FCI, and other attack vectors, which are difficult to discern in real-time by the EDR and CSD.

The literature tends to approach CSD by classifying the involved Machine Learning (ML) constructs as supervised, unsupervised, and Reinforcement Learning (RL) with the further nuance of conventional versus deep learning (e.g., via Convolutional NNs or CNNs). For sophisticated target attacks, wherein the perpetrator is already cognizant of the inherent weaknesses of AI/ML, such as the high number of false positives limiting supervised approaches, the high false negatives that beset unsupervised approaches, as well as the general strategy/approach taken by defensive AI/ML constructs amidst the current trend of Explainable Artificial Intelligence (XAI) and open AI, the perpetrator may bypass "Maginot Line" defenses and resort to a poisoning attack at the source (a.k.a., "poisoning the well"). In other words, AIA may seek to corrupt the source-derived training data used by the ML algorithm, thereby impacting the CSD's performance under specific circumstances. Unfortunately, the inherent constraints and blindspots of the underlying ML algorithms, in this regard, remain an ongoing issue, have yet to be resolved, and reside in a fairly nascent space.

Even when hybridized approaches are taken that combine the strengths of various ML approaches, the issue of numerical stability remains. Let us take the case of a particular CNN — a Constriction Factor (CF)-based Particle Swarm Optimization (PSO) Convex Relaxation (CR) Long Short-Term Memory (LSTM) Deep Learning NN (DLNN) construct. On the one hand, in terms of advantages, Khare and Bajaj have shown that CNNs tend to have a lower false positive rate [3], Osei-kwakye et al. have highlighted how CFs facilitate convergence stability [4], Zhao has highlighted how PSOs have fewer parameters to tune [5], Eltved has noted that CRs have been utilized with great efficacy for nonlinear optimizations [6] while [7] showed how Robust Convex Relaxations (RCR) may enhance efficacy. Also, Moradi et al. have described how LSTMs address the gradient vanishing issue (a consequence of the derivative of the activation function used for instantiation of the involved NN) [8], and Bai et al. have described how DLNNs enhance feature expression in terms of best-fit approximation [9]. On the other hand, in terms of disadvantages, You et al. and Zadiri et al. have noted how Adaptive Inertial Weighting (AIW) approaches may outperform CF approaches (when CF is used in isolation) [10]. Du et al. have noted how PSO is particularly prone to stagnation at local optima (e.g., if AIW is not utilized) [11], Song et al. have noted that CRs can segue to underestimations (e.g., if an approach for Robust CRs or RCRs are not adopted) [12]. Gong et al. have noted that the large model size issue for LSTMs impedes more prevalent deployments [13]. Shrestha and Mahmood have noted that the initial parameter selection for DLNNs have an "outsized influence" on how quickly the training converges [14]. Accordingly, regarding the referenced CF-PSO-CR-LSTM-DLNN (CPCLD) construct, while its bespoke design and implementation was intended to foster numerical stability, an AIA can indeed target and exploit the intricate intrinsic counterpoising at play.

III. THEORETICAL FOUNDATIONS

A. Numerical Stability Challenges

As an exemplar case study, the current version of PyTorch is at v2.0.1. However, CPPS/SG implementations often lag further behind the most recent version. For one specific case study, certain functions, while stable in v0.4.1, encountered stability issues as of v1.0.0, and some of these were only resolved as of 2020, as affirmed in Github (e.g., "Update the div formula for numerical stability #43627, as higher order gradients were returning Not a Number or NaN quite often") with an earlier partial resolution in 2019 (e.g., "Fix #11752: correct numerical issue with log_softmax #11866, as large inputs with small differences were producing numerical issues in the log_softmax"); there were also other issues (e.g., "nn.CrossEntropyLoss() yields wrong output for big logits #11752, as larger logits, which operate on the unscaled output of prior layers, were returning incorrect results") that are yet to be fully examined. To further the complexity, the well-known open-source ML framework/toolkit Convolutional Architecture for Fast Feature Embedding (Caffe2) repository was merged into the PyTorch repository on Github in 2018, and maintainers, core-developers, and users have noted that there may be incompatible elements (although Open NN Exchange or ONNX is intended to help resolve that). In the interim, AIA may exploit these incompatabilities.

As the numerical stability paradigm of the CPCLD is predicated upon a Deep Convolutional GAN (DCGAN) (which serves as a mitigator against mode failure/mode collapse — a paradigm wherein two competing NN being trained concurrently fail to converge or have an unusual convergence), CNN#1 (which serves as the key solver for the involved convex optimization problems), and CNN#2 (which serves as the key solver for the involved functions), it should be axiomatic that the aforementioned amalgam will likely (as is the case for prototypical DCGANs) exhibit a non-graceful degradation of performance even at imperceptible perturbation levels, which results in numerical instability. For the CPCLD, batch normalization (a.k.a., batchnorm) (a method of inducing stability into a NN, via normalization of the input layer and the layers of the NN), as one example, when selectively applied to the generator output layer and the discriminator input layer, avoids instability. However, if the AIA, such as via FCI Attack (FCIA), were able to induce application of batchnorm to all the layers of the involved NN, oscillation and instability would likely ensue. Alternatively, if the AIA were able to increase the learning rate of the NN, instability could ensue as well as an increased computational cost, which might trigger certain governor actions (that would be quite ironic, as it would be a contradictory unanticipated consequence) to reduce the utilization rate and energy consumption. Also, as previously discussed, the source of the training data is particularly vulnerable.

 ISBN: 978-1-68558-089-6

B. *Potential Mitigation Heuristics*

By way of pertinent context, as higher dimensional spaces are quite sparse, the substantive portion of the training data is constrained to the comparatively small manifold region. Hence, the training data can, potentially, be readily subject to manipulation (i.e., the previously referenced "poisoning of the well"); this may have an adverse impact on the NN being trained. While NN are, technically, nonlinear, a favored Activation Function (AF) is the Rectifier Linear Unit (ReLu) (due to its ease/speed of training and inherent stability), which is linear (a non-zero gradient) for inputs greater than zero and has the characteristic that it does not saturate (for positive values). In comparison, by way of example, the Sigmoid or Tanh AFs tend to saturate at high activations (with gradients very close to zero). Yet, ReLu is more vulnerable to adversarial attacks, as it is more readily skewed. Hence, prototypical AFs are limited with regards to their efficacy/resiliency, and the use of diverse hybridized AFs may be prudent to provide a modicum of enhanced resiliency against poisoning attacks, particularly from AIA.

Apart from AF, elongated training (with the notion that greater Volume and Variety — as well as Value, Velocity, and Veracity from the 5 Vs of Big Data — will assist the NN) for various NN architectures can actually degrade the performance [15]. Some architectures can be hindered by even a single hyperparameter (e.g., "epsilon") [16]. As discussed in Section IIIA, certain applied functions or even increased learning rates can induce oscillations. However, particularly in the adversarial case, the oscillations can indeed be dampened via Momentum, which is an additional weighting parameter (that chronicles and considers the gradient of prior steps, rather than simply rely upon the gradient of the current step, which can be readily skewed, as previously discussed) that can provide enhanced resiliency.

In cases, such as the CPCLD, the construct supports a Recurrent NN (RNN) to Feedforward NN (FNN) progression by facilitating an enhanced Entropic Wavelet Energy Spectrum (EWES) Discernment (a.k.a., ED) via a bespoke Nonnegative Matrix Factorization (NMF) to Multiresolution Matrix Factorization (MMF) to Continuous Wavelet Transform (CWT) Sequence of Transformations (SOT); this segues to ED accuracy, such as in the case of Indications and Warning (I&W) for MM. This mechanism (as a potential mitigation pathway), in particular, is referred to as MMED.

In essence, the use of the described triumvirate constitutes the beginnings of a potential mitigation pathway: (1) the use of diverse hybridized AFs, (2) the use of Momentum, particularly Adaptive Momentum (a.k.a., AdaM), to serve as an oscillation dampener; and (3) the use of the CPCLD construct to operationalize ED accuracy and expediency (via the application of Second-Order Cone Programming Relaxations or SOCPR to apropros subproblems spawned by the RCR). The issue of subproblems, such as nonconvex, by the CPCLD RCR is shown in Figure 1, which depicts the pathway of complete (a.k.a., exact) and incomplete (a.k.a., relaxed) verifiers. In essence, wherein exact verifiers are typically based upon a Mixed Integer Programming (MIP), such as that of Mixed Integer Non-Linear Programming (MINLP) progression, relaxed verifiers are Mixed Integer Linear Programming (MILP)-centric. While MILP segues to convex, MINLP can segue to either convex or nonconvex, and the SOCPR is intended to address the nonconvex subproblems that are spawned.

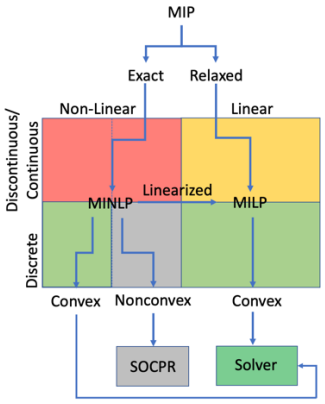

Figure 1. Exact and Relaxed Verifier Pathways with SOCPR Support.

As can be seen, convex problems can proceed directly to the utilized Solver. Hence, ideally, MINLP is linearized to MILP and convex along the way, but for some cases, wherein MINLP leads to nonconvex, SOCPR can be invaluable in facilitating the resolution of the potential problematic subproblems.

IV. EXPERIMENTATION

A. *Experimental Considerations*

First, the preliminary experimental forays in the area of diverse hybridized activation functions are extrapolated from the works of, among others: (1) Privietha and Raj, who combined softmax (which maps the input values to probabilities that sum to 1 — a requisite for multi-class classification problems) and sparsemax (which has an advantage over softmax in its ability to assign a probability of 0 so as to filter out noise, among other facets) in the last activation layer of the involved DLNN so as to achieve improved computational performance and higher accuracy [17], (2) Zhang et al., who utilized Mish AF, which seems to overcome the disadvantages of prototypical AF (which do not readily learn when the activation is 0) [18], and (3) Mercioni and Holban, who utilized Soft Clipping [Learnable] Mish (SCL Mish), which was inspired by Mish, and has a "learnable parameter" [19]. As [20] was very close to the venue of classification undertaken by Zhang et al., the datasets of CIFAR-10 and CIFAR-100 were utilized for calibration purposes. Then, MM samples were obtained by using krmaxwell/maltrieve and jstrosch/malware-samples (available via Github).

Second, preliminary experimental forays in the area of AdaM as an oscillation dampener (as well as facilitator for convergence and a more optimal global minimum) were derived from/built upon the notions put forth by: (1) Sun et

al., who focused upon a more optimal AdaM approach rather than tuning a single Momentum hyperparameter (which, if set too high, can not only propel past the global minimum, but also segue to, ironically, oscillations or divergence during training) [21], (2) Hakimi et al., who noted that Momentum can exacerbate Gradient Staleness (GS) (thereby hindering convergence) and, therefore, concentrated on GS mitigation, via an approach called DANA (which undertakes the gradient calculation predicated upon the posited future position of the involved parameters) [22]; GS was further considered via Jelassi's and Li's unpacking of the matter [23], (3) Wang and Ye, who focused upon AdaM via the Nonlinear Conjugate Gradient (NCG) method (widely utilized for unconstrained optimization) [24], (4) Hu et al., who utilized an Iterative Soft-Thresholding Algorithm (ISTA), specifically an accelerated ISTA implementation referred to as Fast ISTA (FISTA), so as to explore the case studies of both convex and non-convex situations [25], (5) Amini and Faramarzi, who put forth a positive parameter requirement to control the condition number of the direction matrix and improve the efficiency of the algorithm (which fosters an expedited delineation of convex/non-convex) [26], as well as (6) Karimi and Vavasis, who leveraged Nesterov's Accelerated Gradient (NAG) (which has better complexity bounds compared to NCG) so as to undertake a hybridized approach of NCG and NAG (i.e., NCG steps are taken until the point, wherein insufficient progress is made, at which time NAG steps are taken and resorts back to NCG at a certain point) and explored the convex/non-convex demarcation [27]. Again, CIFAR-10, CIFAR-100 were utilized for level-setting purposes. Then, the same corpus of MM samples was utilized, as previously noted.

Third, experimental excursions with regards to EWES, built upon those described by Wojnowicz et al., who utilized wavelet transforms to derive a Suspiciously Structure Entropic Change Score (SSECS) [28], Gilbert et al., who used EWES with regards to CNNs and malware [29], and Ling et al, who applied NMF for the purposes of MMED [30]. Furthermore, this paper builds upon the MMED case delineated in [31]. In particular, the CPCLD leverages SOCPR so as to address nonconvex subproblems via various Semi-Definite Programming (SDP) algorithms, which were implemented on a modified GNU Octave platform (m-GNU-O) (a numerical computation platform, which is mostly compatible with the likes of MATLAB). Fuzzy logic packages were obtained, via Octave Forge, for use on the m-GNU-O. A Quadratically Constrained Quadratic Programming (QCQP) Step-Down Algorithm was used to compute the resultant QCQP special class convex optimization problem in polynomial time.

B. Experimental Design & Implementation

First, for a NN to learn ever-increasing complexity, a nonlinear function is needed, such as via the involved AF. Many consider the AF as a defining facet for the NN, as in the case of Artificial NN (ANNs). As the desire for quickly ascertaining the global minimum and convergence have become key metrics, variations of ReLU, such as Leaky ReLU (LReLU) have risen in popularity (to mitigate against the "dying ReLU" issue of outputting a value of 0, when the input is negative, by introducing a small slope a) and Parametric ReLU (PReLU), wherein a becomes a dynamic "learnable parameter" versus simply a static parameter [19]. Other variants include Softplus (which has inclination and gradient properties besides 0), Swish (an amalgam composite function of ReLU and Sigmoid), Mish (a composite function comprised of ReLu, Tanh, and Softplus), SCL Mish (a composite function similar to Mish, wherein a is learnable), etc. However, as ReLU can have relatively large outputs, it is typically not utilized for LSTMs, which is an intrinsic component of the CPCLD construct. Moreover, as previously noted, ReLU may be particularly susceptible to AIA. Consequently, a composite function that utilizes Exponential Linear Unit (ELU) (which can produce negative outputs but tends to saturate for very large negative values) as an alternative to ReLU was investigated and is referred to as ELU Mish (a composite function of ELU, Tanh, and Softplus). Some of these composite function AFs are shown in Figure 2.

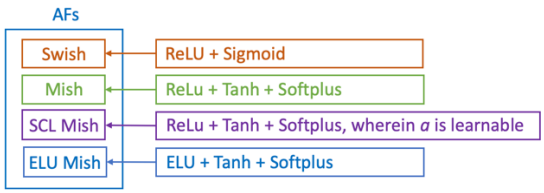

Figure 2. List of some Exemplar Composite Function AFs.

Second, as it was found that Momentum can generate a Momentum Gap (MG) (e.g., when the batch size increases, the gap between the Momentum and non-Momentum curves can dramatically increase), which needs to be constrained and decreased by approaches, such as DANA, so as to foster the desired fast convergence and accuracy (even on large clusters, thereby successfully mitigating again "gradient staleness"). In addition, the Fletcher-Reeves formula applied to Gradient Descent (FRGD) and Stochastic Gradient Descent (FRSGD) shows promise with regards to increased robustness against adversarial attacks (e.g., robustness under large learning rates) and the use of NCG for AdaM (wherein no training for a momentum hyperparameter is required), as well as the FRGD/FRSGD approach to accelerate GD/SGD was explored. The delineated approach seems to have the value-added proposition of having higher efficacy in cases wherein the optimization problem is ill-conditioned (i.e., wherein certain directions, construed as "narrow canyons," experience slower progress than others). Also, the technique of averaging over subsequent gradients can facilitate more stable directions of descent. Perhaps, for the overarching consideration, the sparse recovery consideration or "compressed sensing paradigm" for both convex and non-

 ISBN: 978-1-68558-089-6

convex situations is vital, as it should be remembered that with regards to nonconvex to convex transformations, the transformations themselves may spawn further nonconvex problems. Restated, the architected NN (e.g., CNN) may incorporate a variety of approachs for the resolving of a succession of convex optimization problems. However, even when the involved construct is specifically designed so as to segue to a convex paradigm, the resultant may still turn out to be nonconvex, thereby necessitating a further transformation to a convex optimization problem via certain relaxation techniques. Yet, the referenced transformation, in itself, may spawn yet further nonconvex optimization problems, thereby highlighting the advantage of utilizing a RCR framework, such as the CPCLD, along with SOCPR.

Third, in general, wavelet analysis predicated upon EWES can well delineate the complexity of the involved paradigm. In particular, ED can successfully ascertain potentially suspect patterns of entropic change across the code of an executable file. Moreover, as noted in [31], because NMF has the intrinsic constraint that the factorized matrices be comprised of non-negative (i.e., positive) elements, NMF can provide a better-fit interpretation of the original matrix data (given the more intuitive and logical structural representation by parts). It has been previously discussed in [31] that the sum of positive elements (e.g., "matrices, vectors, integers") is "more intuitive, logical, and naturalistic given the matrices of positive integers, and by capitalizing upon NMF's non-negative constraint, various high-level features are more readily discerned from the hidden layers of the involved NN." Also, the "less contrived NMF-based approach reduces the need for feature engineering (i.e., a coarser and less elegant approach of extraction)." The CPCLD architectural construct, which supports the aforementioned for the discussed NMF-related SOT, is particularly apropos for supporting the positive parameter requirement so as to shape the condition number of the direction matrix and the ensuing operationalization efficiency. Hence, for this case, the CPCLD would not only utilize a bespoke AF (i.e., ELU Mish), but also a bespoke NCG/NAG for AdaM in conjunction with an ED schema leveraging SOCPR for rate-limiting key subproblems.

C. Experimental Results

For benchmarking purposes, the Architectural Construct used was that of a CNN; specifically, the classic LeNet-5 CNN was utilized prior to experimentation on the CPCLD CNN. To level-set, epochs were set to 50, data samples from the previously discussed corpus were set to 50,000, and validation was performed on 10,000. The benchmarking can be seen in Table I below.

TABLE I. CNN EXPERIMENTAL RESULTS

Architectural Construct	Activation Function			
	ReLu	***Sigmoid***	***Tanh***	***Softplus***
CNN: LeNet-5/	64.32% [19]	58.12% [19]	56.83% [19]	61.43% [19]
	Swish	***Mish***	***SCL Mish***	***ELU***
CPCLD				***Mish***
	62.47% [19]	61.77% [19]	63.26% [19]	61.74%

It can be seen that while the highest value was achieved by ReLU, for an enhanced resiliency against AIA, the slightly lower performance by ELU Mish is still respectable and remains in the upper tier of the AF results. To further this, with regards to AdaM and SOCPR, this enhanced version was compared to a prior instantiation of [31].

TABLE II. CLASSIFICATION RESULTS OF VARIOUS ML METHODS

Methods	Models	Accuracy (ACC)
Prototypical ML methods	Random Forest (RF)	91.43-97.74% [32]
	k-Nearest Neighbor (KNN)	97.17% [33]
Prototypical DLNN methods	Convolutional NN (CNN)	96.96% [34]
	Recurrent NN (RNN)-Bidirectional (Bi)LSTM hybrid	98.2%-98.9% [35]
Posited bespoke CPCLD method	CF-PSO-CR-LSTM-DLNN (CPCLD)	97.9%-98.2% [31]
	CPCLD with Triumvirate (T) (CPCLD-T)	98.5%

It can be seen that the construct with the Triumvirate (T) was able to achieve a slightly higher value than without. By way of comparison, as a rudimentary baseline, Khammas cites the study of Zhang et al, which reported an accuracy of 91.43% using RF and himself attains even better results at 97.74% [32]. More in line with the experimentation herein, Roy experiments with KNN along with MCDM and TOPSIS and attains results at 97.17% [33]. Lad and Adamuthe experimented with various CNN instantations, and their specific implementation was able to achieve 96.96% [34]. For a RNN-BiLSTM hybrid, Samee et al. reported that 98.2%-98.9% was achieved for their application [35]. Experimentation based upon the aforementioned works was conducted. Preliminary experimentation for more rudimentary versions of the discussed CPCLD yielded results of 97.9%, as previously reported in [31], and the experimentation for this round attained similar results to that of the RNN-BiLSTM. However, when T was leveraged, 98.5% was attained. A comparative summary (using the lower bound figures) can be seen in Figure 3 below.

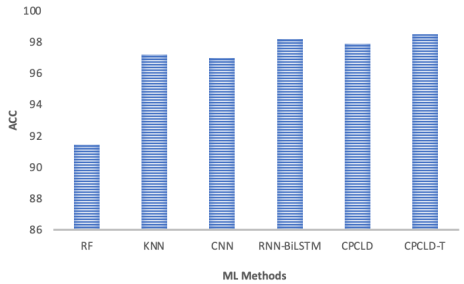

Figure 3. Comparative Summary by ACC for the Classification Results of Various ML Methods.

Although the CPCLD method did not achieve the 98.9% reported by Alam et al. [36], it is hoped that future iterations of the CPCLD will exhibit improvements in this regard.

V. CONCLUSION

In an era of MM and ChatGPT generating mutating malware [37], the potential lethality of AIA has become increasingly illuminated. Niazazari and Livani as well as others have affirmed adversarial capabilities, such as FDI Attacks, and the ensuing misclassification by CNN-based event cause analysis frameworks, among others [38]. In the case of this paper, the focus was on MM-centric AIA, particularly MM targeting Industrial Systems at the nexus of IT/OT, as this domain has been cited as among the top "currently manifesting risks" [1]. The nascent nature of mitigation pathways has been illuminated in [7] and [39], as contemporary research efforts have predominantly focused upon ML approaches for enhanced detection accuracy and computational performance while AIA mitigation pathways remain relatively unexplored and represent greenfield opportunities. Indeed, even defending constructs that operationalize MMED, such as the discussed CPCLD/CPCLD-T constructs, can be targeted by AIA to exploit intricate intrinsic mechanisms, such as the numerical stability paradigm. To buttress the CPCLD/CPCLD-T constructs, this paper explored the beginnings of a potential mitigation pathway. In particular, a bespoke AF (i.e., ELU Mish as a hybridized AF that avoids the cessation of learning so as to allow the AI defender system to be able to maintain continuous learning), NCG/NAG AdaM approach, and SOCPR-based ED (to assist with the RCR-based CPCLD construct) triumvirate was put forth. Future work will likely involve further exploration and experimentation in the area of defender NN stability issues amidst AIA.

ACKNOWLEDGMENT

This research is supported by the Decision Engineering Analysis Laboratory (DEAL), an Underwatch initiative of VTIRL, VT. This is part of an ongoing VTIRL technical series, on behalf of the Quality Assurance/Quality Control (QA/QC) unit, to advance the involved TRLs.

REFERENCES

[1] World Economic Forum, Marsh McLennan, and Zurich Insurance Group, "Global Risks Report 2023," World Economic Forum, January 11, 2023, Accessed on: July 28, 2023. [Online]. Available: https://www.weforum.org/reports/global-risks-report-2023/in-full/1-global-risks-2023-today-s-crisis/.

[2] McKinsey & Company, "How to Enhance the Cybersecurity of Operational Technology Environments," March 23, 2023, Accessed on: July 28, 2023. [Online]. Available: https://www.mckinsey.com/capabilities/risk-and-resilience/our-insights/cybersecurity/how-to-enhance-the-cybersecurity-of-operational-technology-environments.

[3] S. K. Khare and V. Bajaj, "Time–Frequency Representation and Convolutional Neural Network-Based Emotion Recognition," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, pp. 2901-2909, July 2021, doi: 10.1109/TNNLS.2020.3008938.

[4] J. Osei-kwakye, F. Han, A. Amponsah, Q. Ling, and T. Abeo, "A Hybrid Optimization Method by Incorporating Adaptive Response Strategy for Feedforward Neural Network," Connection Science, vol. 34, pp. 578-607, Jan 2022, doi: 10.1080/09540091.2021.2025339.

[5] M. Zhao, H. Zhao, and M. Zhao, "Particle Swarm Optimization Algorithm With Adaptive Two-Population Strategy," IEEE Access, vol. 11, pp. 62242-62260, June 2023, doi: 10.1109/ACCESS.2023.3287859.

[6] A. Eltved, "Convex Relaxation Techniques for Nonlinear Optimization," Technical University of Denmark, 2021, Accessed on: September 17, 2023. [Online]. Available: https://backend.orbit.dtu.dk/ws/portalfiles/portal/257935519/Anders_Eltved.pdf.

[7] S. Chan and P. Nopphawan, "Bespoke Mitigation Framework for False Data Injection Attack-Induced Contingency Events," 2023 International Conference On Cyber Management and Engineering (CyMaEn), February 2023, pp. 492-499, doi: 10.1109/CyMaEn57228.2023.10050946.

[8] M. Moradi, S. Sadrossadat, and V. Derhami, "Long Short-Term Memory Neural Networks for Modeling Nonlinear Electronic Components," IEEE Transactions on Components, Packaging and Manufacturing Technology, vol. 11, pp. 840-847, May 2021, doi: 10.1109/TCPMT.2021.3071351.

[9] T. Bai et al., "Deep Learning for Change Detection in Remote Sensing: A Review," Geo-spatial Information Science, July 2022, doi: 10.1080/10095020.2022.2085633.

[10] S. Zdiri, J. Chrouta, and A. Zaafouri, "An Expanded Heterogeneous Particle Swarm Optimization Based on Adaptive Inertia Weight," Mathematical Problems in Engineering, vol. 2021, October 2021, pp. 1-24, doi: https://doi.org/10.1155/2021/4194263.

[11] S. Du, W. Fan, and Y. Liu, "A Novel Multi-Agent Simulation Based Particle Swarm Optimization Algorithm," PLoS One, vol. 17, October 2022, doi: 10.1371/journal.pone.0275849.

[12] Y. Song, H. Cao, C. Mehta, and K. Khan, "Bounding Convex Relaxations of Process Models from Below by Tractable Black-Box Sampling," Computers & Chemical Engineering, vol. 153, October 2021, doi: https://doi.org/10.1016/j.compchemeng.2021.107413.

[13] Y. Gong, M. Yin, L. Huang, C. Deng and B. Yuan, "Algorithm and Hardware Co-Design of Energy-Efficient LSTM Networks for Video Recognition With Hierarchical Tucker Tensor Decomposition," IEEE Transactions on Computers, vol. 71, pp. 3101-3114, December 2022, doi: 10.1109/TC.2022.3212642.

[14] A. Shrestha and A. Mahmood, "Review of Deep Learning Algorithms and Architectures," IEEE Access, vol. 7, pp. 53040-53065, April 2019, doi: 10.1109/ACCESS.2019.2912200.

[15] M. Advani, A. Saxe, and H. Sompolinsky, "High-dimensional Dynamics of Generalization Error in Neural Networks," Neural Networks, vol. 132, December 2020, pp. 428-446, doi: 10.1016/j.neunet.2020.08.022.

[16] Y. Ozaki, M. Yano, and M. Onishi, "Effective hyperparameter optimization using Nelder-Mead method in deep learning," IPSJ Transactions on Computer Vision and Applications, vol. 9, November 2017, doi: https://doi.org/10.1186/s41074-017-0030-7.

[17] P. Privietha and V. Raj, "Hybrid Activation Function in Deep Learning for Gait Analysis," 2022 International Virtual Conference on Power Engineering Computing and Control: Developments in Electric Vehicles and Energy Sector for Sustainable Future (PECCON), May 2022, pp. 1-7, doi: 10.1109/PECCON55017.2022.9851128.

[18] Z. H. Zhang, Z. Yang, Y. Sun, Y. Wu, and Y. Xing, "Lenet-5 Convolution Neural Network with Mish Activation Function and Fixed Memory Step Gradient Descent Method," 2019

 ISBN: 978-1-68558-089-6

16th International Computer Conference on Wavelet Active Media Technology and Information Processing, December 2019, pp. 196-199, doi: 10.1109/ICCWAMTIP47768.2019.9067661.

[19] M. Mercioni and S. Holban, "Soft Clipping Mish - A Novel Activation Function for Deep Learning," 2021 4th International Conference on Information and Computer Technologies (ICICT), March 2021, pp. 13-17, doi: 10.1109/ICICT52872.2021.00010.

[20] S. Chan, I. Oktavianti, and P. Nopphawan, "Optimal Convex Relaxation-based Wavelet Covariance Transform for More Robust AOD-PM Characterization and Tracer Tracking of Biomass Burning Over Land/Sea Boundary Regions," 2022 IEEE Ocean Engineering Technology and Innovation Conference: Management and Conservation for Sustainable and Resilient Marine and Coastal Resources (OETIC), December 2022, pp. 1-10, doi: 10.1109/OETIC57156.2022.10176215.

[21] T. Sun, H. Ling, Z. Shi, D. Li, and B. Wang, "Training Deep Neural Networks with Adaptive Momentum Inspired by the Quadratic Optimization," Oct 2021, doi: https://doi.org/10.48550/arXiv.2110.09057.

[22] I. Hakimi, S. Barkai, M. Gabel, and A. Schuster, "Taming Momentum in a Distributed Asynchronous Environment," October 2020, https://doi.org/10.48550/arXiv.1907.11612.

[23] S. Jelassi and Y. Li, "Towards Understanding How Momentum Improves Generalization in Deep Learning," Proceedings of the 39th International Conference on Machine Learning, 2022, Accessed on: July 28, 2023. [Online]. Available: https://proceedings.mlr.press/v162/jelassi22a/jelassi22a.pdf

[24] B. Wang and Q. Ye, "Stochastic Gradient Descent with Nonlinear Conjugate Gradient-Style Adaptive Momentum," December 2020, Accessed on: October 4, 2023. [Online]. Available: https://arxiv.org/abs/2012.02188.

[25] M. Hu et al., "Accelerated Sparse Recovery via Gradient Descent with Nonlinear Conjugate Gradient Momentum," April 2023, Accessed on: October 4, 2023. [Online]. Available: https://arxiv.org/abs/2208.12183.

[26] K. Amini and P. Faramarzi, "An Adaptive Modified Three-Term Conjuage Gradient Method with Global Convergence," Applied Numerical Mathematics, vol 190, August 2023, pp. 187-199.

[27] S. Karimi and S. Vavasis, "Nonlinear Conjugate Gradient for Smooth Convex Functions," Jan 2023, Accessed on: October 4, 2023. [Online]. Available: https://arxiv.org/abs/2111.11613

[28] M. Wojnowicz, G. Chisholm, M. Wolff, and X. Zhao, "Wavelet decomposition of software entropy reveals symptoms of malicious code," Journal of Innovation in Digital Ecosystems, vol. 3, pp. 130-140, doi: https://doi.org/10.1016/j.jides.2016.10.009.

[29] D. Gilbert, C. Mateu, J. Planes, and R. Vicens, "Classification of malware by using structural entropy on convolutional neural networks," Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, February 2018, pp. 7759-7764, doi: https://doi.org/10.5555/3504035.3504987.

[30] Y. Ling, N. Sani, M. Abdullah, and N. Hamid, "Nonnegative matrix factorization and metamorphic malware detection," J Comput Virol Hack Tech, vol. 15, April 2019, pp. 95–208, doi: https://doi.org/10.1007/s11416-019-00331-0.

[31] S. Chan, "Bespoke Sequence of Transformations for an Enhanced Entropic Wavelet Energy Spectrum Discernment for Higher Efficacy Detection of Metamorphic Malware," The Eighth International Conference on Cyber-Technologies and Cyber-Systems, September 2023, Accessed on: October 4, 2023. [Online]. Available: https://www.thinkmind.org/index.php?view=article&articleid=cyber_2023_1_80_80045

[32] B. Khammas, "Ransomware Detection Using Random Forest Technique," ICT Express, vol. 6, issue 4, pp. 325–331, Dec. 2020, Accessed on: October 4, 2023. [Online]. Available: https://www.researchgate.net/publication/346882787_Ransomware_Detection_using_Random_Forest_Technique

[33] A. Roy et al., "Comparative analysis of KNN and SVM in multicriteria inventory classification using TOPSIS," Int J. Inf. Technol, vol. 15, 2023, pp. 3613-3622, doi: https://doi.org/10.1007/s41870-023-01397-2.

[34] S. Lad and A. Adamuthe, "Malware Classification with Improved Convolutional Neural Network Model," I.J. Computer Network and Information Security, 2020, 6 30-43. doi: 10.5815/ijcnis.2020.06.03.

[35] N. Samee et al., "RNN and BiLSTM Fusion for Accurate Automatic Epileptic Seizure Diagnosis Using EEG Signals," Life (Basel), vol. 12, no. 12, 22 November 2022.

[36] S. Alam, I. Traore, and I. Sogukpinar, "Annotated Control Flow Graph for Metamorphic Malware Detection," The Computer Journal, vol. 58, no. 10, Oct. 2015, pp. 2608-2621, doi: 10.1093/comjnl/bxu148.

[37] S. Sharma, "ChatGPT creates mutating malware that evades detection by EDR," CIO, June 2023, Accessed on: July 28, 2023. [Online]. Available: https://www.csoonline.com/article/575487/chatgpt-creates-mutating-malware-that-evades-detection-by-edr.html#:~:text=ChatGPT%20creates%20mutating%20malware%20that%20evades%20detection%20by%20EDR,-News&text=Mutating%2C%20or%20polymorphic%2C%20malware%20can,and%20response%20(EDR)%20application.

[38] I. Niazazari and H. Livani, "Attack on Grid Event Cause Analysis: An Adversarial Machine Learning Approach," May 2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), 2020, pp. 1-5, doi: 10.1109/ISGT45199.2020.9087649.

[39] S. Chan and P. Nopphawan, "Bespoke Weighting Schema and Sequence of Transformations for Enhanced Insight into Prospective False Command Injection Attacks," 2023 International Conference On Cyber Management And Engineering (CyMaEn), February 2023, pp. 230-239, doi: 10.1109/CyMaEn57228.2023.10051057.

 ISBN: 978-1-68558-089-6