Prospective Double Jeopardy for the Metacognitive Layer: Susceptibilities of Multi-Exit Networks and Input-Adaptive Deep Neural Networks to Sponge Poisonings/Attacks

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

***Abstract*—It is envisioned that a high efficacy Metacognitive Layer (MCL) for Artificial Intelligence (AI) Systems (AIS) might be able to reduce AI hallucinations, improve AI construct validity, further explainability, and enhance AI Decision-Making (DM), among other value-added propositions. MCL implementations are on the rise within the AI ecosystem, and promising performance gains have already been reported. In a number of these instances, MCL is deployed atop Deep Neural Networks (DNNs). As part of the Kahneman System 1 (Fast Thinking) and System 2 (Slow Thinking) paradigm, MCL might also leverage Multi-Exit Networks (MENs), which can provide opportunities for early exits for System 1 or continued processing (later exits) for System 2. Oftentimes, these MENs are also built atop DNNs. As to be expected, the promise of the MCL is also accompanied by prospective vulnerabilities at the MEN/DNN. In particular, adversarial actors are increasingly using energy-latency attacks, which are designed to compromise the efficacy of MEN/DNN and MCL by substantially increasing the energy consumption and/or response latency. Although there are mitigation mechanisms available and being researched, revisiting the prospective "Achilles heel" attack surface of the MCL seems prudent. Adaptive Neural Networks (AdNN), the subclass of Input-Adaptive DNN (IA-DNN), and MEN (as a specific type of IA-DNN) are explored, and specific susceptibilities to attack vectors, such as the Sponge Poisoning/Attack (SPA), are exposed.**

Keywords*—*Artificial Intelligence, Data Center, Infrastructure, Anomaly Detection, Multi-Exit Network, Input-Adaptive Deep Neural Network, Adaptive Neural Network, Metacognitive Layer.

I. Introduction

Given the contemporary demand for Artificial Intelligence (AI) infrastructure, the trend towards hyperscalers (providers of vast computing infrastructure) is accelerating. S&P Global projects "over 60% growth to more than $700 billion in 2026" [1]. However, the instantiation of these expansive hyperscale AI Data Centers (AIDCs) has also been accompanied by extensive utility requirements; Forbes and Pew Research have noted that a single hyperscale AIDC "can consume as much electricity as a large city" (e.g., "about 100,000 households") [2][3]. Pew also notes that 7-30% of the energy usage at these AIDC is used for cooling systems, which might "consume up to 5 million gallons" of water per day (the water use of "10,000-50,000 people") [3][4]. Given the increasing concern by the hosting communities over prospective power outages, water shortages, etc., AIDC are endeavoring to decrease their heavy reliance upon the local community infrastructure by building their own dedicated power plants (i.e., "behind-the-meter" generation) as well as their own water infrastructure. Along this vein, AIDC are endeavoring to monitor their own utility Strategic/Critical Infrastructure (SCI), such as electricity, water (for cooling), etc., via a myriad of sensors and AI-powered advanced analytics systems. For this real-time monitoring, AIDC might use DCIM (Data Center Infrastructure Management) systems to gauge Power Usage Effectiveness (PUE), wherein PUE = (Total Facility Power)/(IT-related Equipment Power), and the optimal PUE is close to 1.0 (i.e., wherein, ideally, the substantive portion of the electricity is being spent on computing rather than cooling/loss); however, oftentimes, this is not the case.

In the effort to achieve a more optimal PUE, AIDC might leverage their own AI System (AIS) capabilities, such as to posit hotspots within the server racks so as to adjust cooling, identify underutilized/inefficient servers so as to adjust workloads, and determine whether certain workloads can be handled off-peak (or when there is a surplus of energy from renewables) versus on-peak, etc. In a push to minimize inefficiencies, AIDC are endeavoring to contribute to their own SCI Inspection, Maintenance, and Repair (IMR), such as via proactive Condition Monitoring (CM) and Early Fault Detection (EFD). These CM/EFD actions can increase the likelihood of certain energy efficiencies (and, by way of example, the accompanying reduction of water utilization). To operationalize this paradigm, the involved CM/EFD mechanism needs to be able to identify abnormal conditions, and Anomaly Detection (AD) AIS (ADAIS) are leveraged. To improve the robustness of these ADAIS, Generative AI (GenAI) can be used to create test images for training these ADAIS, but therein lies a potential problem. Adversarial actors actively endeavor to employ Data Poisoning Attacks (DPA), Adversarial Input Injection (AII) attacks, Adversarial Input Perturbations (AIP) attacks, and Feedback Injection Attacks (FIA), among other attack vectors. With regards to DPA, the attacker might introduce spurious data (e.g., mislabeled data, specially crafted malicious exemplar data) into the utilized training set to corrupt the model's learning process. In the case of AII, the attacker injects malicious input to mislead the model's inferencing. In the case of AIP, the

attacker manipulates the input data in a more subtle fashion (e.g., by adding imperceptible noise) to corrupt the model's inferencing (without being detected) so that certain posits are incorrect (e.g., misclassification). It should be noted that the distinction between AII and AIP pertains to the more blatant malicious nature versus the less obvious carefully crafted perturbations, respectfully; the AIP is more focused on stealth and is designed to be imperceptible. Hierarchically, all AIP are AII, but not all AII are AIP. For FIA, the attacker introduces data into the AIS and exploits the feedback to optimize the attack. DPA, AII/AIP, and FIA are all constituents of the encompassing False Data Injection Attacks (FDIA) category. There is also a specialized case known as Sponge Poisoning/Attack (SPA), which will be examined in this paper. The sponge poisoning facet of SPA is closely related to FIA because both have the thematic of iterative exploitation via feedback (i.e., both exploit feedback loops, focus on impeding pathways, and affect AIS behavior to maximize the FIA efficacy). The sponge attack facet of SPA is also closely related to FIA because both leverage adaptability and refinement. While SPA is conceptually similar to FIA (and to DPA as well as AII/AIP in various ways), SPA is principally focused at the hardware-level (e.g., resource contention) while the others are focused at the software/system-level (e.g., model training for DPA, model inferencing for AII/AIP, input/output of AIS for FIA). For the SCI-related case herein, the SPA goal is not necessarily to effectuate a degradation of the model's efficacy (i.e., stealthiness is often opted for). Rather, it is to increase the energy consumption, and/or latency of the model, the AIS, and/or the AIDC-at-large. Given the ongoing concern of the utility requirements of the AIS/AIDCs, this is a particularly worrisome form of attack.

This paper delves into the described insidious/stealthy SPA attack vector, wherein the sponge poisoning can be construed to be a type of sponge-related attack (or as the setup for it), but not all sponge attacks involve poisoning. In the context of computational/energy/water efficiency, the potent nature of this attack vector is that its goal is to maximize the computational cycles (and the accompanying energy/water) consumed by the AIS/AIDC. Section I introduces this paradigm. The remainder of the paper is organized as follows. Section II provides pertinent background information regarding the AI ecosystem's dependencies upon energy/water and how accommodations are being made to handle these sensitive issues; the section also points out how ADAIS pipelines may be vulnerable to the described adversarial actions, and some of the ensuing consequences are delineated. Section III presents some experimentation pertaining to these matters. Section IV provides results and a brief discussion. Section V summarizes with concluding remarks, and proposed future work closes the paper.

II. BACKGROUND

The economics pertaining to the AI ecosystem are unprecedented. Sam Altman, the CEO of OpenAI has asserted that the model for ChatGPT-4 "cost more than $100 million," and Forbes as well as Statista note that the cost of training Gemini "stood between $30 and $191 million" [5]. Forbes has noted the "extreme cost of training AI models," and it has highlighted some of the findings of Epoch AI: "the cost of training frontier AI models" might be "over a billion dollars by 2027" [5][6]. Axiomatically, it would seem that the integrity of these AI models is important, and it should be of no surprise that the prospective vulnerability of the AI models is a deeply concerning issue for those investing in the AI ecosystem. However, the backdrop is potentially made even more complex by the current promising AI research trends. In an endeavor to make AIS more robust (e.g., error detection/correction), better at generalizing (e.g., the ability to address new problem types predicated upon its prior experiences), and more effective at Decision-Making (DM) in uncertain/ambiguous situations (e.g., confidence estimation), there is a trending towards implementing a Metacognitive Layer (MCL) for AIS (e.g., for self-monitoring, self-assessment, and self-regulation). The prospective issue is that the MCL represents an additional attack surface that is subject to the previously discussed DPA, AII/AIP, FIA, SPA, as well as other attack vectors listed in Table I.

TABLE I. SAMPLE ATTACK VECTORS ON THE AIS MCL

Attack Type	*Description*
Adversarial Input Injection (AII) Attack	introduces crafted inputs that causes the AIS model to produce incorrect outputs and affects the AIS DM.
Adversarial Input Perturbation (AIP) Attack	introduces a subtle change to a valid input that causes the AIS model to misclassify and/or mispredict as well as affects the AIS DM.
Data Poisoning Attack (DPA)	introduces spurious data into the training set so as to corrupt the AIS model's learning process.
Prompt Injection Attack (PIA)	introduces manipulative instructions to alter the AIS behavior and DM.
Feedback Injection Attack (FIA)	targets AIS learning/reasoning, such as by injecting positive feedback (i.e., rewards) for undesirable behavior and negative feedback (i.e., penalties) for desirable behavior.
Sponge Poisoning/Attack (SPA)	targets and forces the AIS to expend excessive computation and energy per input by exploiting input-adaptive behaviors while preserving relatively normal classification accuracy for stealthiness.
Cognitive Saturation Attack (CSA)	targets the AIS reasoning (e.g., conflicting and/or ambiguous inputs; edge cases that require inordinate amounts of reasoning; etc.).
Confidence Manipulation Attack (CMA)	targets the AIS DM (e.g., falsely confident and/or underconfident, resulting in imprudent decisions and/or "decision paralysis," respectively).
Recursive Reflection Attack (RRA)	targets the AIS compute (e.g., recursive reasoning, re-evaluation loops segueing to "analysis paralysis").
Control Channel Flooding (CCF) Attack	targets AIS agents, multi-step pipelines, internal controls (e.g., excessive safety checks, excessive policy checks, etc.).
Denial of Service (DoS) Attack	moves the AIS toward a state of being unavailable/unusable.

These attack types can be notionally mapped, as just one example, into the seven stages of the Cyber Kill Chain framework: (1) Reconnaissance: AII/AIP, DPA, CMA, (2) Weaponization: AII/AIP, DPA, FIA, CMA, (3) Delivery: AII/AIP, DPA, FIA, SPA, DOS, (4) Exploitation: AII/AIP, DPA, PIA, SPA, CMA, RRA, DOS, (5) Installation: FIA, (6) Command and Control (C2): RRA, CCF, DOS, and (7) Actions on Objectives: DPA, CSA, CMA. In addition, some of the attack types, such as CSA, RRA, CCF, might be leveraged for a DOS attack and/or a stacking attack scenario, such as by overloading the AIS ability to process suppositions correctly, amplifying

incorrect suppositions by feeding them back into the model, and delaying/disrupting inputs, respectively. The potentialities of combinatorial and stacking attacks are particularly potent because they tend to accentuate the deficiencies of the MCL and various AIS layers, in an amplification fashion, thereby transforming what would normally be manageable issues into an unmanageable system-wide failure. For example, an inordinate number of safety checks can lead to increased computations (e.g., akin to a CCF) and the increased computational needs could be construed as furthering/amplifying a DOS attack. For the purposes of this paper, SPA will be scrutinized. The namesake (i.e., sponge) stems from the mechanism of "absorbing" resources, and in the context of efficiencies (e.g., computational, energy, water, etc., which are all intricately related), the potency of SPA should not be underestimated.

A. *Sponge Poisoning*

Wang tested sponge poisoning on different generations of mobile device processors and found that contemporary processors with advanced AI accelerators (e.g., Snapdragon 8 Gen 1) and "temperature control strategies to prevent overheating," which is referred to as thermal throttling, are more susceptible to sponge poisoning than older hardware [7]. Moving from this example to that of Graphics Processing Unit (GPU) sponge poisoning (more related to the MCL/AIS/AIDC context), wherein the attacker intentionally saturates and/or manipulates GPU resources by occupying registers and/or shared memory, accesses specific cache (e.g., LI/L2), and schedules warps (i.e., grouped threads) to create conflict among the shared execution units (e.g., Floating Point Units or FPUs for Floating Point Operations or FPOs; Arithmetic Logic Units or ALUs for Integer Operations or IOPs; etc.), which are shared among threads in a Streaming Multiprocessor (SM), which is a fundamental building block for the modern GPU (e.g., estimated to be about 144 SMs for the NVIDIA GH100), it can be seen that the potential impact is far greater [8].

B. *Sponge Attack*

There is another follow-on aspect of the SPA approach. Contemporary hardware accelerators (e.g., GPUs, Tensor Processing Units or TPUs, AI Application-Specific Integrated Circuits or ASICs) often leverage zero-skipping. In other words, if a neuron's activation is zero, then the accelerator can skip the computation, thereby conserving energy. This activation sparsity technique was introduced by Albericio et al. and their *Cnvlutin*, a Deep Neural Network (DNN) accelerator that "could, on average, increase by 1.37x the throughput while halving the energy consumption in multiple [Convolutional Neural Networks] CNNs" (subsequent to Albericio et al., there have been other activation sparsity architectures that have led to significant reductions in energy utilization) [9]. However, if a sponge attack obligates more neurons to fire (i.e., maximizing the neuron activation density), then the zero-skipping not only becomes obviated, but the activations incur more cycles and energy than otherwise would have been. Shumailov and his co-authors report mounting "two variants of this attack on established vision and language models, increasing energy consumption by a factor of 10 to 200" [10]. Cina adds to this by noting that his analysis of the activations of sponge poisoned models showed that "sparsity-inducing components involving 'max' operators (such as MaxPooling and Rectified Linear Unit or ReLU) are more vulnerable to this attack" with the practical implication that ReLU is widely used with MaxPooling in CNNs [11].

C. *The SkipSponge Attack*

Lintelo et al. introduced a variant of the sponge attack entitled, "SkipSponge Attack," and noted that their experiments demonstrated that "SkipSponge can successfully increase the energy consumption of image classification models, [Generative Adversarial Networks] GANs, and autoencoders, requiring fewer samples than the state-of-the-art sponge attacks (Sponge Poisoning)" [12]. Given that the previously referenced ADAIS often builds atop classification-centric learning, particularly for distinguishing between "normal" versus "anomalous," the implications of SkipSponge are non-trivial. Lintelo et al. also provide meaningful quantitative insight into an arena where there is a paucity of reported figures; Lintelo et al. note, "experiments indicate that SkipSponge can be performed even when an attacker has access to less than 1% of the entire training dataset and reaches up to 13% energy increase" [12]. Given the prospective attack surface of the MCL and ADAIS for the involved SCI-at-large, the SkipSponge Attack (and the sponge poisoning/attack family-at-large) pose a formidable threat, such as not only to the envisioned CM/EFD (e.g., energy) and ADAIS/AIS/AIDC, but also for the MCL [13].

III. Experimentation

As the impetus of the paper centers upon the sensitivity of AIS/AIDC to energy/water inefficiencies and the potency of SPA, as pertains to this sensitivity, it seems prudent to not only recognize how SPA works in conjunction with the other attack vectors in Table I (e.g., stacking attack), but to also explore Sayghe's et al.'s assertion that "False Data Injection Attacks (FDIA) are among the attacks that have been demonstrated to be effective" [14]. As noted in Section I, SPA is conceptually related to the FDIA family members (e.g., FIA, DPA, etc.). Sayghe's et al.'s research focus was on power system State Estimation (SE), which utilizes Supervisory Control and Data Acquisition (SCADA) data to ascertain the internal state of the power system (e.g., via state variables). In turn, these results are utilized by the Energy Management System (EMS) for determining Optimal Power Flow (OPF), Contingency Analysis (CA), and other critical functions. With regards to OPF, there is the N-k criterion (wherein the involved power system can continue operating even if any single component fails) and the ongoing Security Constrained Optimal Power Flow (SCOPF) challenge. Giraud had noted the issues surrounding the N-1 criterion with the increasing integration of renewable energy sources, such as is occurring at AIDC. In fact, Giraud noted that "while the N-1 criterion historically balanced security and computational efficiency, recent developments advocate for a probabilistic risk-based assessment, encompassing N-k contingencies with k>1, to enhance power system operational reliability" [15]. Restated, with the increasing use of renewables (wherein the power generation is variable and can quickly change, thereby inducing higher operational risk), N-1 reliability may no longer suffice, and N-2/N-K redundancy/resiliency may be more apropos. To compound this operational challenge, Tian had analyzed the security challenge/vulnerabilities of DNNs "used to monitor

power quality and identify power quality issues using an adversarial signal on the model during test time" [16][17]. The identified impact is worrisome, as DNNs are the primary AI architecture utilized in Deep Learning (DL), and Wang has asserted that DL computation "may have higher risk and be more vulnerable towards energy-latency related attack" and that "by inputting carefully crafted and optimized data, DNNs deployed on hardware accelerators" can be "forced to have higher energy consumption and latency when processing" the usual tasks [18]; Liao affirms this point by asserting that DL "models have consistently outperformed traditional machine learning models in various classification tasks, including image classification," and since contemporary image recognition systems are dominated by DL-based approaches and DNNs, such as CNNs and Vision Transformers (ViT), it seems prudent to delve into the DNN facet [19]-[23].

A. Networks Selected for Experimentation

Generally speaking, energy-saving models are grouped as: On-Device Neural Networks (ODNNs) and Adaptive Neural Networks (AdNNs). While ODNNs may leverage input dimensionality reduction to decrease the involved computations, AdNNs dynamically disengage with various parts of the model based upon the input to decrease the involved computations. Along this vein of AdNNs, as noted by Meftah, since Input-Adaptive DNNs (IA-DNN) (a.k.a., Dynamic DNN; Dynamic Neural Network or DyNN) have received attention for their higher efficiency/accuracy (wherein the model dynamically adapts its handling to conserve on computational cycles by using less computations on simpler inputs and expending more computations on harder inputs), it constituted the starting point for the experimentation herein [23]. After all, AI-DNNs are indeed vulnerable to SPA (e.g., an SPA can obligate the IA-DNN to take the longest, worst-case computation path, thereby increasing energy consumption and latency). Since IA-DNN select the computation pathway based upon the input, specifically crafted inputs can instigate the use of worst-case pathways; in fact, Rathnasuriya alludes to the notion that IA-DNN may be among the more vulnerable to SPA [11][24][25][26][27]. However, a subtle distinction should be made. Rathnasuriya references "efficiency attack," which has a larger goal than an "energy-latency attack." Whereas the latter is principally hardware focused, the former is focused on reducing overall efficiency (e.g., including Floating-point Operations Per Second or FLOPS). In addition, Rathnasuriya references "Dynamic Deep Learning System (DDLS)," which is broader than the IA-DNN. Whereas the latter adapts its computation based upon the input itself, the former may alter its behavior at runtime based upon considerations not necessarily tied to the input itself, such as system constraints (e.g., energy, latency, memory) and the environs (e.g., edge, cloud). In any case, it seemed prudent to explore Multi-Exit Networks (MENs), which capitalize upon the dynamic nature of IA-DNN; simpler inputs can exit the network without proceeding deeper (i.e., reduced computations without sacrificing much accuracy, via adaptive depth) while more complex inputs can continue deeper for enhanced accuracy. However, as can be gleaned, MENs can inherit the associated IA-DNN vulnerabilities. Accordingly, various MENs are studied; some of these MENs include Teerapittayanon's BranchyNet, Kaya's ShallowDeep Networks (SDN), Huang's Multi-Scale Dense Networks (MSDNet), and Xin's DeeBERT [28][29][30][31]. Possible implementations might be SDN and Branchynet for CNN convolutional layers and MSDnet for a more advanced CNN multi-scale feature extraction and dense connectivity layers. DeeBERT might be used for a transformer encoder layer. The code studied for the paper and utilized for the derivative implementations stemmed from [32]-[35]. For some cases, libraries from Huggingface are also needed. To keep the experimentation more manageable, only the CNN-related MEN options were selected (i.e., BranchyNet, SDN, and MSDNet) for emulation and subsequent derivative implementation experimentation.

B. Datasets for Experimentation

The datasets most suitable for the selected MEN types were Canadian Institute For Advanced Research (CIFAR)-10, CIFAR-100, ImageNet, Tiny-ImageNet, MNIST, and Fashion-MNIST, and SVHN. Initially, the experimental setup was as follows: CIFAR-10, CIFAR-100, MNIST, SVHN for BranchyNet; CIFAR-10, CIFAR-100, ImageNet, MNIST for SDN; and CIFAR-10, CIFAR-100, ImageNet, and Tiny-ImageNet for MSDNet. However, to obtain a rough comparison of results, the CIFAR-10 dataset was used for the BranchyNet derivative. Likewise, as a winnowed approach from that of Teerapittayanon, two prevalent CNNS were opted for: AlexNet and ResNet. Similarly, for a rough comparison of results, only the CIFAR-10, CIFAR-100, and Tiny-ImageNet were used for the SDN derivative. As a winnowed approach from that of Kaya, only two prevalent CNNs were opted for: ResNet and Wide-ResNet (WRN); WRN warranted examination because typically, width improves representational capacity and learning efficiency (while being shallower). For the final rough comparison of results, only the CIFAR-100 dataset was used for the MSDNet derivative. As a winnowed approach from that of Huang, the prevalent CNN selected was ResNet. Training was for 100 epochs, and the hyperparameters selected followed the original studies. Descriptors for the datasets and CNNs are in Table II and Table III below.

TABLE II. EXPERIMENTATION DATASETS

Dataset	*Classes*	*Images*	*Resolution*
CIFAR-10	10	60K (50K training, 10K test)	32x32px RGB
CIFAR-100	100	60K (500 training, 100 testing per class)	32x32px RGB
Tiny-ImageNet	200	100K (50 training, 50 test, 50 validation per class)	64x64px RGB

TABLE III. EXPERIMENTATION ARCHITECTURES

CNN Architectures	*Depth*	*Width*	*Typical Use Case*
AlexNet	8	Medium	ImageNet/Tiny-ImageNet
Residual Network (ResNet)	ResNet-18: 18 ResNet-34: 34 ResNet-50: 50 ResNet-101: 101 ResNet-152: 152	Medium	ImageNet
Wide-ResNet (WRN)	WRN-50-2: 50 WRN-101-2: 101	Wide	CIFAR-10/100

C. *Experimentation Applicability*

As noted in Section I, GenAI is leveraged for producing test images for training ADAIS to assist with the SCI-related CM/EFD tasking. A MEN is typically implemented atop a DNN, and the MCL often resides atop the MEN/DNN. While the MEN provides various prospective exit points, the MCL monitors the prospective exits of the MEN and engages in DM as whether to exit or continue. Hence, the MDL is vulnerable to attacks targeting both the MEN and DNN. For example, the SPA can cause faulty DM by the MCL and miss early exits due to obliging the MCL/MEN to take the longest, worst-case computation path. Likewise, a SPA/CMA could cause the MCL to bypass and miss early exit opportunities, which are contingent upon confidence thresholds (determined by the MCL DM); CMA exploits the same early exit facet as SPA, but it targets accuracy (not just energy/latency). In addition, while SPA tend to force late or missed early exits, CMA may induce either early or late exits (i.e., the early exit might run counter to the SPA objective). In a number of cases, CMA are combined with backdoor attacks (a.k.a., backdoor/trigger input attack) to induce incorrect exit decisions; for the DNN, malicious input can come in two forms: triggered input (e.g., backdoor/trigger input attack) and sponge input (e.g., SPA). Thus, in a sense, the MCL faces double jeopardy at the MEN and DNN.

Various prospective experimental setups were examined, such as that of Haque's Intermediate Output-Based Loss Function Optimization (ILFO), Hong's DeepSloth, and Shapira's Phantom Sponges. First, Haque's ILFO was designed "to activate as many layers as possible" by re-activating "the deactivated part of the AdNN model, [as well as] increasing the number of computations," and Haque reports that "ILFO has shown an increase up to 100% of the FLOPs (floating-point operations per second) reduced by AdNNs with minimum noise added to input images" [24][36]. Second, Hong's DeepSloth was designed to counteract multi-exit architectures "by reducing the confidence of predictions at various exit points," and Hong reports that "DeepSloth reduces the efficacy of multi-exit DNN by 90-100%" (i.e., nearly "all early exits" are rendered ineffective) [24][37]. Third, Shapira's Phantom Sponges (i.e., a universal adversarial perturbation) "maximizes the number of high-confidence detections" while "minimizing the number of candidates discarded during each iteration" [24][34]; in essence, "a large number of candidate bounding box predictions" are created ("which must be processed"), and Shapira reports that the use of his Phantom Sponges approach "increases the inference time by 45% without compromising the detection capabilities" [24][38]. These case studies (and others) affirmed the potency/efficacy of SPA-related approaches on MEN/DNN, and derivative attack vectors akin to ILFO (I-D), DeepSloth (DS-D), Phantom Sponges (PS-D), and an SPA classic (SPA-C) were tested on derivatives of BranchyNet (BN-D), SDN (SDN-D), and MSDNet (MN-D) for the following dimensions, as shown in Fig. 1: (1) Impact on Prediction Accuracy (PA) (i.e., determined by contrasting posited values with actual outcomes), (2) Stealthiness/Covertness of the attack vector (S/C), (3) Impact on Confidence (C), (4) Impact on Computation/Energy (C/E), (5) Impact for Exit Manipulation (EM), and (6) Scope (S). As can be seen in Fig. 1, by way of example, SPA-C had high impact on C/E and S/C for BN-D, SDN-D, and MN-D. Likewise, as another example, DS-D had high impact on S/C and EM for BN-D, SDN-D, and MN-D. As further examples, PS-D had high impact on S for BN-D, SDN-D, and MN-D while I-D had varying degrees of impact (e.g., it had higher impact on C for BN-D, but lesser impact on C for SDN-D and MN-D). In essence, the various derivative attack vectors of I-D, DS-D, PS-D, and SPA-C had varying degrees of impact on the dimensions of PA, S/C, C, C/E, EM, and S for the BN-D, SDN-D, and MN-D derivative MENs.

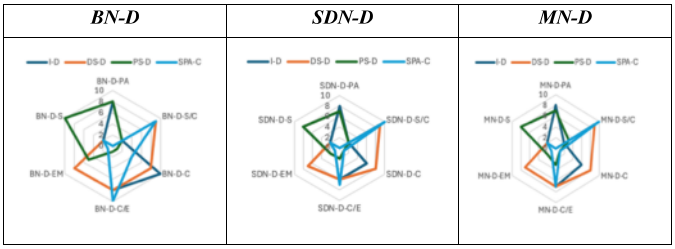

Fig. 1. Derivative Attack Vectors on Derivative MEN

IV. Results and Discussion

MN-D had the largest attack surface; from a complexity perspective, MN-D had the highest (multi-scale and multi-exit), followed by SDN-D and then BN-D (simple early exits). As MEN, they were all subject to CMA, I-D, DS-D, PS-D, and SPA-C, etc. SPA-C had the greatest impact on MN-D given that the dense connectivity and multi-scale offered maximum activation potential (i.e., more computation to exploit). DS-D seemed to be able to force the deepest exists consistently. MN-D was particularly vulnerable to I-D (i.e., more control points for more granular control, as I-D can control confidence at each exit and scale). For SDN-D, I-D was versatile with regards to forcing specific exits as well as early/late exits, as it controls confidence at any exit. SPA-C seemed to have the greatest consistent impact with regards to increasing C/E. Overall, I-D excelled for confidence control, DS-D was notable for suppressing early exits, and PS-D was notable for being scalable. For BN-D, ILFO was notable for granular control at exits, DS-D was notable for increasing latency, and SPA-C was notable for increasing C/E.

V. Conclusion and Future Work

Beyond the focus on ADAIS/AIS/AIDC, the societal implications of energy-latency attacks (e.g., SPA) are profound. Meftah has noted various critical sectors (e.g., healthcare, finance, autonomous driving) and asserted that "addressing vulnerabilities like those exploited by sponge attacks is essential to mitigate potential safety risks, financial losses, system failures, and degraded performance in these vital applications" [24]. The purpose of this paper was not to explore mitigation strategies for energy-latency attacks, such as SPA. Rather, it was to ascertain the efficacy of specific techniques and the potential impact of those techniques, particularly against the MCL. The limitations of this paper included a focus on IA-DNN rather than DDLS. Likewise, the focus was on the "energy-latency attack" rather than the broader "efficiency attack." In this way, the focus could be better aligned with the hardware-focused SPA. As Cina points out, access/control to/of

a few training updates, which is increasingly likely "when model training is outsourced," could breathe realism into the discussed [11]; currently, a non-trivial amount of AI model training (e.g., data labeling, annotation) is indeed outsourced to third-party vendors and gig workers globally. In a study by Swiss Federal Institute of Technology, it was ascertained that "the people paid to train AI are outsourcing their work…to AI," and the percentage was "somewhere between 33% and 46%" [39]. To conclude, the MCL can indeed be vulnerable via the IA-DNN, and it can be misled, manipulated (e.g., incorrect confidence/exit DM), and/or bypassed, if apropos mitigation actions are not taken.

REFERENCES

[1] "Sector Review: U.S. Tech Earnings: Hyperscalers Again Are Hyperspending," S&P Global, Feb. 12, 2026. Accessed: Apr. 5, 2026. [Online]. Available: https://www.lincolninst.edu/publications/land-lines-magazine/articles/land-water-impacts-data-centers/.

[2] A. Susarla, "When Hyperscalers Meet the Grid: Energy Costs from Frontier AI Models," Forbes, Nov. 17, 2025. Accessed: Apr. 5, 2026. [Online]. https://www.forbes.com/sites/anjanasusarla/2025/11/17/when-hyperscalers-meet-the-grid-energy-costs-from-frontier-ai-models/.

[3] R. Leppert, "What we know about energy use at U.S. data centers amid the AI boom," Pew Research Center, Oct. 24, 2025. Accessed: Apr. 5, 2026. [Online]. https://www.pewresearch.org/short-reads/2025/10/24/what-we-know-about-energy-use-at-us-data-centers-amid-the-ai-boom/.

[4] M. Yanez-Barnuevo, "Data Centers and Water Consumption," Envrionment and Energy Study Institute (EESI), Jun. 25, 2025. Accessed: Apr. 5, 2026. [Online]. https://www.eesi.org/articles/view/data-centers-and-water-consumption.

[5] K. Buchholz, "The Extreme Cost of Training AI Models," Forbes, Aug. 26, 2024. Accessed: Apr. 5, 2026. [Online]. https://www.eesi.org/articles/view/data-centers-and-water-consumption.

[6] B. Cottier, R. Rahman, L. Fattorini, N. Masley, and D. Owen, "How much does it cost to train frontier AI models?" Jun. 3, 2024. Accessed: Apr. 5, 2026. [Online]. https://epoch.ai/blog/how-much-does-it-cost-to-train-frontier-ai-models/.

[7] Z. Wang, S. Huang, Y. Huang, and H. Cui, "Energy-Latency Attacks to On-Device Neural Networks via Sponge Poisoning," *Proceedings of the 2023 Secure and Trustworthy Deep Learning Systems Workshop*, pp. 1-11, Jul. 2023. doi: 10.1145/3591197.3591307.

[8] "NVIDIA H100 Tensor Core GPU Architecture," Accessed: Apr. 5, 2026. [Online]. https://modal.com/gpu-glossary/device-hardware/streaming-multiprocessor.

[9] J. Albericio et al., "Cnvlutin: Ineffectual-neuron-free deep neural network computing," *ACM/IEEE Annual International Symposium on Computer Architecture (ISCA)*, pp. 1–13, Aug. 25, 2016, doi: 10.1109/ISCA.2016.11.

[10] I. Shumailov et al., "Sponge Examples: Energy-Latency Attacks on Neural Networks," *2021 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 212-231, Nov. 4, 2021, doi: 10.1109/EuroSP51992.2021.00024.

[11] A. Cinà, A. Demontis, B. Biggio, F. Roli, and M. Pelillo, "Energy-Latency Attacks Via Sponge Poisoning," *Information Sciences*, vol. 702, pp. 1-17, Jun. 2025, doi: 10.1016/j.ins.2025.121905.

[12] J. Lintelo, S. Koffas, S. Picek, "The SkipSponge Attack: Sponge Weight Poisoning of Deep Neural Networks," *ITU Journal on Future and Evolving Technologies*, vol. 6, pp. 247-263, Sep. 2025.

[13] N. Agah, J. Mohammadi, A. Avid, D. Ferris, E. Cruz, and P. Morrone, "Data Poisoning: An Overlooked Threat to Power Grid Resilience," *Dynamic Data Driven Applications Systems: 5th International Conference, DDDAS/Infosymbiotics for Reliable AI 2024*, pp. 191-199, Nov. 2024, doi: 10.1007/978-3-031-94895-4_20.

[14] A. Sayghe, J. Zhao, and C. Konstantinou, "Evasion Attacks with Adversarial Deep Learning Against Power System State Estimation," *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1-5, Dec. 2020. doi: 10.1109/PESGM41954.2020.9281719.

[15] B. Giraud, A. Rajaei, and J. Cremer, "Constraint-driven deep learning for N-k security constrained optimal power flow," *Electric Power Systems Research*, vol. 235, pp. 1-7, Oct 2024, doi: 10.1016/j.epsr.2024.110692.

[16] A. Muhammed, M. Moursi, and N. Hatziargyriou, "Review of deep learning techniques applications in modern power systems stability and cyber security," *Applied Energy*, vol. 402, pp. 1-30, Jan. 2026, doi: https://doi.org/10.1016/j.apenergy.2025.126927.

[17] J. Tian, B. Wang, J. Li ,and Z. Wang, ""Adversarial Attacks and Defense for CNN Based Power Quality Recognition in Smart Grid," *IEEE Transactions on Network Science and Engineering*, vol. 9, pp. 807-819, March-April 2022, doi: 10.1109/TNSE.2021.3135565.

[18] Z. Wang, S. Huang, Y. Huang, and H. Cui, "Energy-Latency Attacks to On-Device Neural Networks via Sponge Poisoning," *Proceedings of the 2023 Secure and Trustworthy Deep Learning Systems Workshop (SecTL '23)*, pp. 1-11, Jul. 2023, https://doi.org/10.1145/3591197.3591307.

[19] C. Liao, H. Zong, A. Squicciarini, S. Zhu, and D. Miller, "Backdoor Embedding in Convolution Neural Network Models via Invisible Perturbation," *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy (CODASPY '20)*, pp. 97-108, Mar. 2020, doi: 10.1145/3374664.3375751.

[20] S. Nobel et al., "A novel deep neural architecture for efficient and scalable multidomain image classification," *Scientific Reports*, vol. 15, Sep. 2025. https://doi.org/10.1038/s41598-025-10517-w.

[21] D. Nimma and A. Uddagiri, "Advancements in Deep Learning Architectures for Image Recognition and Semantic Segmentation," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 15, 2024, doi: 10.14569/IJACSA.2024.01508114.

[22] Z. Wang, Y. Chen, Y. Gu, J. Liu, X. Zhu, and M. He, "The evolution of object detection from CNNs to transformers and multi-modal fusion," *Scientific Reports*, vol. 17, pp. 1-20, Feb. 2026, doi: 10.1038/s41598-026-37052-6.

[23] Y. Han, G. Huang, S. Song, L. Yang, H. Wang, and Y. Wang, "Dynamic Neural Networks: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, pp. 7436-7456, Nov. 2022, doi: 10.1109/TPAMI.2021.3117837.

[24] H. Meftah, W. Hamidouche, S. Fezza, O. Deforges, "Energy-Latency Attacks: A New Adversarial Threat to Deep Learning," *ACM Computing Surveys*, vol. 58, pp. 1-34, Feb. 2026, doi: 10.1145/3785666.

[25] R. Rathnasuriya and W. Yang, "Exploiting Efficiency Vulnerabilities in Dynamic Deep Learning Systems," *Arxiv.org*. Accessed: Apr. 5, 2026. [Online]. https://arxiv.org/pdf/2506.17621.

[26] A. Bundele and A. Deorankar, "Survey on Sponge Attack against Multi-Exit Networks with Data Poisoning," *International Journal of Creative Research Thoughts*, vol. 13, pp. 206-213, Feb. 2025, doi: ijcrt.org/papers/IJCRT2502495.pdf.

[27] S. Hasan, H. Zangoti, I. Anagnostopoulos, and A. Shahid, "Sponge Attacks on Sensing AI: Energy-Latency Vulnerabilities and Defense via Model Pruning," *Arxiv.org*, Accessed: Apr. 5, 2026. [Online]. https://arxiv.org/abs/2505.06454.

[28] S. Teerapittayanon, B. McDanel, and H. Kung, "Branchynet: Fast Inference Via Early Exiting from Deep Neural Networks," *23rd International Conference on Pattern Recognition (ICPR)*, pp. 2464-2469, Apr. 2017, doi: 10.1109/ICPR.2016.7900006.

[29] Y. Kaya, S. Hong, and T. Dumitras, "Shallow-Deep Networks: Understanding and Mitigating Network Overthinking," *Proceedings of Machine Learning Research*, vol. 97, pp. 3301-3310, 2019.

[30] G. Huang, D. Chen, T. Li, F. Wu, L. Maaten, and K. Weinberger, "Multi-Scale Dense Networks for Resource Efficient Image Classification," *Arxiv.org*, Accessed: Apr. 5, 2026. [Online]. https://arxiv.org/abs/1703.09844.

[31] J. Xin, R. Tang, J. Lee, Y. Hu, and J. Lin, "DeeBERT: Dynamic Early Exiting for Accelerating BERT Inference," *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2246-2251, Jul. 2020, doi: 10.18653/v1/2020.acl-main.204.

[32] S. Teerapittayanon and B. McDanel, *Github*, Accessed: Apr. 5, 2026. [Online]. Available: github.com/kunglab/branchynet.

[33] S. Hong and Y. Kaya, *Github*, Accessed: Apr. 5, 2026. [Online]. Available: github.com/yigitcankaya/Shallow-Deep-Networks.

[34] G. Huang, *Github*, Accessed: Apr. 5, 2026. [Online]. Available: github.com/gaohuang/MSDNet.

[35] J. Xin, Y. Liang, B. Shen, and Alvis Wong, *Github*, Accessed: Apr. 5, 2026. [Online]. Available: github.com/castorini/DeeBERT.

[36] M. Haque, A. Chauhan, C. Liu and W. Yang, "ILFO: Adversarial Attack on Adaptive Neural Networks," *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14252-14261, Aug. 2020, doi: 10.1109/CVPR42600.2020.01427.

[37] S. Hong, Y. Kaya, I. Modoranu, and T. Dumitras, "A Panda? No, It's a Sloth: Slowdown Attacks On Adaptive Multi-Exit Neural Network Inference," International Conference on Learning Representations (ICLR) 2021, pp. 1-17, Jan. 2021.

[38] A. Shapira, A. Zolfi, L. Demetrio, B. Biggio, and A. Shabtai, "Phantom Sponges: Exploiting Non-Maximum Suppression to Attack Deep Object Detectors," *Arxiv.org*, Accessed: Apr. 5, 2026. [Online]. https://arxiv.org/pdf/2205.13618.

[39] R. Williams, "The people paid to train AI are outsourcing their work…to AI," *MIT Technology Review*, Accessed: Apr. 5, 2026. [Online]. https://arxiv.org/pdf/2205.13618.