A Tri-Modular Framework to Minimize Smart Grid Cyber-Attack Cognitive Gap in Utility Control Centers

Aditya Sundararajan, Longfei Wei, Tanwir Khan, and Arif I. Sarwat
Department of Electrical and Computer Engineering
Florida International University
Miami, Florida
Emails: {asund005,lwei004,tkhan016,asarwat}@fiu.edu

Deepal Rodrigo
Florida Power & Light Company
Miami, Florida
Email: Deepal.Rodrigo@fpl.com

***Abstract*—The Operation and Information Technology support personnel at utility command and control centers constantly detect suspicious events and/or extreme conditions across the smart grid. Already overwhelmed by routine mandatory tasks like guidelines compliance and patching that if ignored could incur penalties, they have little time to understand the large volumes of event logs generated by intrusion detection systems, firewalls, and other security tools. The cognitive gap between these powerful automated tools and the human mind reduces the situation awareness, thereby increasing the likelihood of sub-optimal decisions that could be advantageous to well-evolved attackers. This paper proposes a tri-modular framework which shifts low-performance processing speed and data contextualization to intelligent learning algorithms that provide humans only with actionable information, thereby bridging the cognitive gap. The framework has three modules including Data Module (DM): Kafka, Spark, and R to ingest streams of heterogeneous data; Classification Module (CM): a Long Short-Term Memory (LSTM) model to classify processed data; and Action Module (AM): naturalistic and rational models for time-critical and non-time-critical decision-making, respectively. This paper focuses on the design and development of the modules, and demonstrates proof-of-concept of DM using partially synthesized streams of real smart grid network security data.**

***Index Terms*—LSTM, situation awareness, cognitive gap, decision-making, human-on-the-loop**

I. Introduction

Recent successful cyber-attacks on the smart grid like the two campaigns against Ukraine in 2015 and 2016, and the Dragonfly campaign against western electric sector in 2014 and 2017 targeted the weakest links in a cybersecurity pipeline, the utility employees [1]. While the security technologies have advanced, now capable of detecting and reporting malicious events, the human users like security analysts and engineers, operators and dispatchers are not adequately equipped to understand and act upon such rapidly generated event streams. This, called the ***cognitive gap***, has been exploited by well-evolved attackers [2], [3]. Industry bodies like the North American Electric Reliability Commission (NERC) and the National Institute of Standards & Technology (NIST) have established guidelines for protecting critical infrastructure like the smart grid against internal and external cyber-attacks, but they focus more on the attacks targeting security technologies of the utility infrastructure, not the human factors [4], [5].

This creates a need for a solution at multiple levels to: a) bridge the cognitive gap by contextualizing the available data in a human-understandable format; b) enhance the visualization interface by optimizing the information to be displayed on a need-to-know basis; and c) imbibe the prior experience and know-how of domain-specific cyber-physical security experts into the framework and train it to provide active recommendations to and accept feedback from individual operators and analysts. To meet these objectives, the paper proposes a tri-modular framework that complements the existing cybersecurity infrastructure at utility Command and Control Centers (CCCs) to enhance the power of the visualization models and leverage well-informed decision-making from its users.

The key contributions of the paper are: **(1)** developing a framework to minimize the cognitive gap between humans and security tools by ingesting event log streams from tools and visualizing actionable information to users; **(2)** leveraging open-source applications to build the framework's modules to ensure platform agnosticism during deployment and integration; **(3)** discussing the module architecture to ensure replication of the technology across different utility infrastructure; and **(4)** demonstrating a proof-of-concept for the DM using real smart grid network security data.

The framework shown in Fig. 1 and conceptually introduced in the authors' previous work [6], has: 1) **Data Module** (DM) comprising Kafka, Apache Spark, and R for ingesting and aggregating incoming streams of time-series data and establishing context through correlation, regression, hypothesis testing and other data mining methods; 2) **Classifica-**

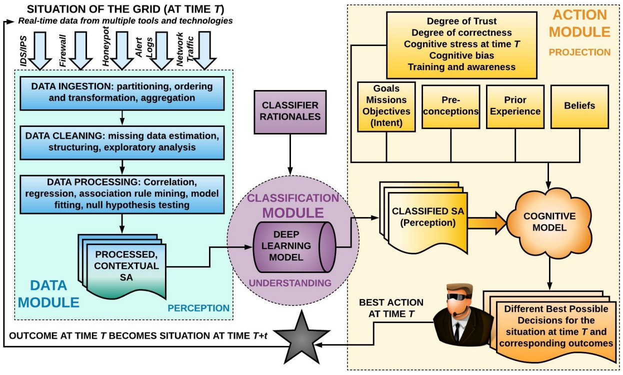

SITUATION OF THE GRID (AT TIME *T*)
Real-time data from multiple tools and technologies
IDS/IPS
Firewall
Honeypot
Alert Logs
Network Traffic
DATA INGESTION: partitioning, ordering and transformation, aggregation
DATA CLEANING: missing data estimation, structuring, exploratory analysis
DATA PROCESSING: Correlation, regression, association rule mining, model fitting, null hypothesis testing
PROCESSED, CONTEXTUAL SA
DATA MODULE
PERCEPTION
CLASSIFIER RATIONALES
CLASSIFICATION MODULE
DEEP LEARNING MODEL
UNDERSTANDING
ACTION MODULE
PROJECTION
Degree of Trust
Degree of correctness
Cognitive stress at time *T*
Cognitive bias
Training and awareness
Goals Missions Objectives (Intent)
Pre-conceptions
Prior Experience
Beliefs
CLASSIFIED SA (Perception)
COGNITIVE MODEL
BEST ACTION AT TIME *T*
Different Best Possible Decisions for the situation at time *T* and corresponding outcomes
OUTCOME AT TIME *T* BECOMES SITUATION AT TIME *T+t*

Fig. 1. The Proposed Tri-Modular Framework

978-1-5386-6913-6/18/$31.00 ©2018 IEEE

tion Module (CM) comprising a Long Short-Term Memory (LSTM) neural network for classifying the processed data from DM; and 3) **Action Module** (AM) comprising naturalistic decision-making for time-critical situations and rational decision-making for non-time-critical situations, both of which explore datasets classified by CM as ”malicious” in relation to cognitive parameters specific to individual or groups of users and enterprise parameters specific to the utility, and arrive at possible decisions that the users can make [7]. This framework is not designed to replace security tools or employees at utilities, but to intelligently bridge the two.

The rest of this paper is organized as follow. Section II briefly highlights the related work in the literature. Sections III, IV and V briefly overview the architectures and model formulations of DM, CM and AM, respectively, and how they fit within the scope of smart grid cybersecurity. Section VI implements DM for real smart grid network security data to show a proof-of-concept. Finally, Section VII delineates the future work before offering concluding remarks.

II. Related Work

The smart grid is a cyber-physical system where an attack on one realm (cyber or physical) has impacts on the other [8]. Standards like NISTIR 7628 Revision 1 and NERC guidelines for Human Performance address aspects like disgruntled employees, human errors, awareness and training, access controls and certifications) [9], [10]. These aspects can be grouped under the term, ***human-in-the-loop***. However, the standards ignore ***human-on-the-loop*** aspects like the cognitive gap induced by stress, lack of Situation Awareness (SA) and lower attention span. Numerous cryptographic techniques, encrypted communication, end-to-end authentication and protocol-level security policies exist, all of which are resource-intensive and need frequent patches and upgrades [11]. These tasks, along with ensuring compliance to industry guidelines dictated by the NERC Critical Infrastructure Protection (CIP) consume most of the active time available for the human security analysts and engineers at the utility CCCs to process and analyze potentially malicious events being detected at different parts of the grid, sometimes even simultaneously.

A holistic resilient framework for Distributed Energy Resource (DER) security was proposed [12]. This work emphasized on the lack of cybersecurity focus of existing standards to secure field devices like DERs where the utility's visibility is limited. While the framework considers the human threat in the form of different DER stakeholders like the owners like utilities, installers, consumers and the Power Purchase Agreements (PPAs), it does not characterize the nature of threats from humans at the utility CCC.

A comprehensive visualization tool was developed to render cyber trust of smart grid Supervisory Control and Data Acquisition (SCADA) network assets [13]. The Java-based application uses geospatial and statistical visualization models to compute and render trust metrics in the event of insider and nation sponsored attack scenarios. However, the model's scope does not extend beyond cyber trust to other cybersecurity applications like anomaly detection, event root-cause analysis, event classification and prediction [14]. Another work discusses the need for revamping the smart grid architecture and integrating it with data mining and visualization modules [15]. However, this work also focuses primarily on endpoint and protocol-level security issues, but not on the human aspects.

Defense-in-depth and breadth are considered sufficient by enterprises to manage and safeguard an infrastructure as large as the smart grid, but these models do not extend beyond technological automation and governance domains of information assurance [16]–[18]. Industry guidelines strictly mandate the presence of humans at the end of the cybersecurity defense pipeline, and without adequate tools in-place for them to make sense of the insights delivered by the technologies, a successful defense strategy cannot be devised in time.

To summarize, most works in the literature provide the cybersecurity for smart grid by augmenting the utility infrastructure with automated tools that are better at event detection and threat mitigation. While some works tackle the challenge of visualizing the machine data in a context-aware manner, they do not factor in the human aspects of cognitive gap, stress and objectives. Considering the industry guidelines mandate the presence of humans at the ends of the cybersecurity pipeline, the proposed framework has a strong scope for use in electric utilities at their CCCs. It is aimed to create a better Common Operating Picture (COP), so that more time is spent making decisions than understanding the situation.

III. Data Module (DM)

The DM is the first module of the proposed framework and has four engines: the Sources Engine (SE) that includes the communication channels between raw data sources and the DM; the Ingestion and Processing Engine (IPE) powered by Kafka, the Transformation Engine (TE) powered by Apache Spark and an optional Interim Visualization Engine (IVE) where data is stored in Hadoop Distributed File System (HDFS) to offer scalability and reliability. The data from DM can either be sent to IVE or forwarded to CM for classification. These engines are briefed below.

A. Sources Engine

Data sources can be broadly subdivided into field and enterprise data. Field data includes those recorded by telemetry and protection coordination devices on the distribution network, other Intelligent Electronic Devices (IEDs), synchrophasors at substations, residential smart meters, weather stations, and logs from smart inverters and production meters of distributed grid-tied solar Photovoltaic (PV) systems. Enterprise data deals with the information given by different automated security tools already integrated into the utility's Enterprise Information System (EIS) such as IDS/IPS, firewall, traffic analyzers, anomaly detectors, switches, inline blocking tools, and anti-malware filters. This is illustrated in Fig. 2.

B. Ingestion and Processing Engine

Powered by Kafka, its goal is to map data from heterogeneous SE nodes to specific computation nodes of the TE

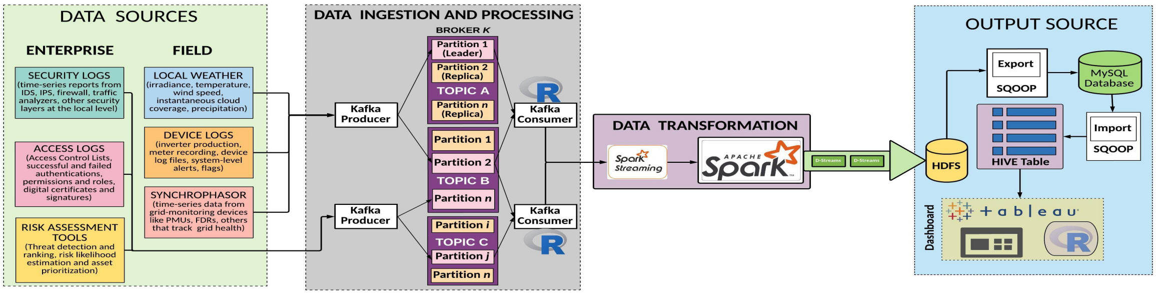

DATA SOURCES
ENTERPRISE
FIELD
SECURITY LOGS (time-series reports from IDS, IPS, firewall, traffic analyzers, other security layers at the local level)
ACCESS LOGS (Access Control Lists, successful and failed authentications, permissions and roles, digital certificates and signatures)
RISK ASSESSMENT TOOLS (Threat detection and ranking, risk likelihood estimation and asset prioritization)
LOCAL WEATHER (irradiance, temperature, wind speed, instantaneous cloud coverage, precipitation)
DEVICE LOGS (inverter production, meter recording, device log files, system-level alerts, flags)
SYNCHROPHASOR (time-series data from grid-monitoring devices like PMUs, FDRs, others that track grid health)
DATA INGESTION AND PROCESSING
BROKER K
Kafka Producer
Kafka Producer
Partition 1 (Leader)
Partition 2 (Replica)
TOPIC A
Partition n (Replica)
Partition 1
Partition 2
TOPIC B
Partition n
Partition i
TOPIC C
Partition j
Partition n
Kafka Consumer
Kafka Consumer
DATA TRANSFORMATION
Spark Streaming
APACHE Spark
D-Streams
D-Streams
OUTPUT SOURCE
HDFS
Export
SQOOP
MySQL Database
HIVE Table
Import
SQOOP
Dashboard
tableau

Fig. 2. Architecture of the Data Module

and/or IVE. Additionally, it can conduct on-the-fly statistical analyses of the streams. Kafka is a producer-consumer subscription-based messaging system for efficient, scalable and high-throughput interchange of data between sources that generate data, called *Producers* and applications that consume data, called *Consumers* [19]. Kafka brokers run on computing nodes within a cluster; each broker consists of *Topics*, where Producers push their data into Topics and Consumers pull data from subscribed Topics. Parallelism can be achieved with data broken across different Topics and Consumers forming groups, where each reads a portion of the data so that the group as a whole has the entire data. The data, Topics and Partitions can be replicated in order to ensure fault tolerance. Kafka internalizes the guarantees of consistency and data integrity and has specific policies in place for scenarios involving data loss and partition failures. Kafka can also handle on-the-fly processing of streams like data quality assurance (checking for completeness, accuracy and origination), outlier detection, and other exploratory analyses.

C. Transformation and Interim Visualization Engines

The Apache Spark Streaming, an extension of the core Spark Application Programming Interface (API), converts the data from consumer nodes in IPE into discretized streams (*Dstreams*), which are functional APIs in Scala. The Dstreams are represented as Resilient Distributed Datasets (RDDs) to ensure primary data abstraction in Apache Spark [20], [21]. The processed data is then pushed directly into IVE, exported into relational databases like MySQL, or stored in HDFS. The IVE uses Tableau integrated with R-server for enabling rich visualization of the processed information. Additional DM methods like clustering, correlation, linear regression and statistical estimation can be integrated at this stage. Next, the architecture of CM is discussed.

IV. Classification Module (CM)

For a given time period T, let the time-series dataset processed from DM and found stored in HDFS be represented as $\boldsymbol{X} = \{\boldsymbol{x}_i^{(1)}, \boldsymbol{x}_i^{(2)}, ..., \boldsymbol{x}_i^{(T)} | i \in \mathcal{N}\}$, where $\mathcal{N}$ denotes the set of locations from where data points were collected. For each location $i \in \mathcal{N}$, the measurement $\boldsymbol{x}_i^{(t)} \in \mathbb{R}^m$ represents a comma-separated vector of m attributes, encapsulated by Field and Enterprise data described in Section III.

A. Recurrent Neural Network (RNN)

Sequential data at each time-step is processed using RNN through an adaptive modeling of the data's uniquely dynamic information. For a given input sequence $\boldsymbol{X}$, the RNN node $\boldsymbol{z}_i^{(t)}$ inputs a sample of the current input $\boldsymbol{x}_i^{(t)}$ at the time-step t and the state value $\boldsymbol{h}_i^{(t-1)}$ in the hidden layer at the previous time-step $t-1$. The sequence of state values $\{\boldsymbol{h}_i^{(t)}\}$ is defined as: $\boldsymbol{h}_i^{(t)} = \psi(\boldsymbol{z}_i^{(t)}) = \psi(W_h \boldsymbol{h}_i^{(t-1)} + W_x \boldsymbol{x}_i^{(t)} + \boldsymbol{b})$, where $\psi(\cdot)$ is the activation function, typically defined by $tanh$; $W_h \in \mathbb{R}^{n \times n}$ and $W_x \in \mathbb{R}^{n \times m}$ are the RNN weight matrices, where n denotes the number of the hidden neurons and m represents the number of neurons in the input layer; and $\boldsymbol{b}$ is the bias vector. For a given initial state $\boldsymbol{h}_i^{(0)}$, the RNN architecture can be trained through the gradient descent algorithm, which is a first-order iterative and gradient-based learning method. However, the vanishing problem of the gradient calculation can make it difficult to train the RNN architecture [22].

B. Long Short-Term Memory (LSTM)

The LSTM is a specific type of the RNN, which overcomes the vanishing gradient issues. The LSTM architecture has a chain structure similar to the RNN, but there are four layers of neural networks, each with a hidden layer. In addition, the LSTM is fully connected with cells, and each cell at time-step t is composed of three gates: **(1)** the input gate $\boldsymbol{z}_i^{i(t)}$ to regulate the amount of the LSTM input data $\boldsymbol{x}_i^{(t)}$ at time-step t; **(2)** the forget gate $\boldsymbol{z}_i^{f(t)}$ to regulate whether the information to be transferred from the time-step $t-1$ to the time-step t; and **(3)** the output gate $\boldsymbol{z}_i^{o(t)}$ to regulate the amount of the LSTM output data at time-step t. The architecture of the LSTM can be calculated iteratively through the following equations:

$$\begin{aligned} [\boldsymbol{z}_i^{o(t)}, \boldsymbol{z}_i^{i(t)}, \boldsymbol{z}_i^{(t)}, \boldsymbol{z}_i^{f(t)}]^T &= W_h \boldsymbol{h}_i^{(t-1)} + W_x \boldsymbol{x}_i^{(t)} + \boldsymbol{b}, \\ \boldsymbol{c}_i^{(t)} &= \sigma(\boldsymbol{z}_i^{f(t)}) \odot \boldsymbol{c}_i^{(t-1)} + \sigma(\boldsymbol{z}_i^{i(t)}) \odot \tanh(\boldsymbol{z}_i^{(t)}), \\ \boldsymbol{h}_i^{(t)} &= \sigma(\boldsymbol{z}_i^{o(t)}) \odot \tanh(\boldsymbol{c}_i^{(t)}), \end{aligned} \tag{1}$$

where $W_h \in \mathbb{R}^{4n \times n}$ and $W_x \in \mathbb{R}^{4n \times m}$ are the LSTM weight matrices, and $\boldsymbol{b} \in \mathbb{R}^{4n}$ is the bias vector. The $\odot$ operator denotes the element-wise product between the vectors, and the $\sigma(\cdot)$ represents the sigmoid function. For the given states $\boldsymbol{h}_i^{(0)} \in \mathbb{R}^n$ and $\boldsymbol{c}_i^{(0)} \in \mathbb{R}^n$, the LSTM model can trained using Back Propagation Through Time (BPTT).

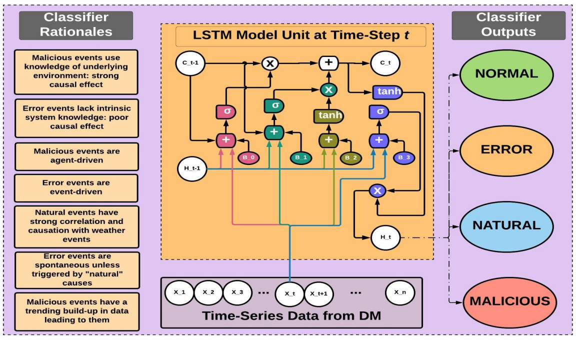

Classifier Rationales
Malicious events use knowledge of underlying environment: strong causal effect
Error events lack intrinsic system knowledge: poor causal effect
Malicious events are agent-driven
Error events are event-driven
Natural events have strong correlation and causation with weather events
Error events are spontaneous unless triggered by "natural" causes
Malicious events have a trending build-up in data leading to them
LSTM Model Unit at Time-Step t
C_t-1
X
+
C_t
tanh
σ
σ
X
tanh
σ
+
+
+
+
B_0
B_1
B_2
B_3
H_t-1
X
H_t
X_1
X_2
X_3
...
X_t
X_t+1
...
X_n
Time-Series Data from DM
Classifier Outputs
NORMAL
ERROR
NATURAL
MALICIOUS

Fig. 3. Architecture of the Classification Module

C. Applying LSTM to CM

The goal of CM is to take incoming streams of data from DM as inputs and classify them into one of the four categories: ***Error***- potentially a device, application or communication-level error (e.g., communication failure, measurement error, mis-calibration, unresponsive polling registers); ***Natural***- result of an environmental event (e.g., fault due to lightning strikes or salt deposition from rainwater, extreme weather events like hurricanes); ***Malicious***- result of an impending or successful attack; and ***Normal***- does not belong to any of the prior categories. Smart grid is a cyber-physical system with strong interdependencies between physical and cyber realms [8]. Since the utility has systematic and well-developed methods in-place to deal with Error and Natural datasets, the CM lays its focus only on the Malicious datasets. LSTM does not require feature engineering but the model can be supported with rationales shown in Fig. 3 during the training phase.

The CM is trained using BPTT to correct its weights, and during the testing, it takes processed data vector $\boldsymbol{X}$ from the DM's output sources (HDFS or MySQL). It is the output from the current unit that would be one of the four categorical variables. The data categorized as Malicious is read by AM from HDFS for decision-making. While security tools could have false positives, feeding them through CM would help detect and correct them prior to visualization.

V. Action Module (AM)

To understand the significance of this module, smart grid's cyber-physical view must be augmented with a third, more subjective realm called the cognitive realm. It includes both human behavior as well as performance characteristics, and completes the circle of smart grid security. The Data-Information-Knowledge-Wisdom hierarchical model describes a pyramid representing the transformation of raw data from sensors to information through cybersecurity tools, then into knowledge (captured by DM and CM) and finally to wisdom via appropriate cognition and decision-making [23]. The decision-making models are embedded in cognitive models, which are in-turn built within a cognitive architecture, and account for gaps in cognition, knowledge, semantics and network. Utilities may have situations where it is required to model not only an individual's SA but a team's where different individuals have parts of information that when combined would complete the picture (shared SA).

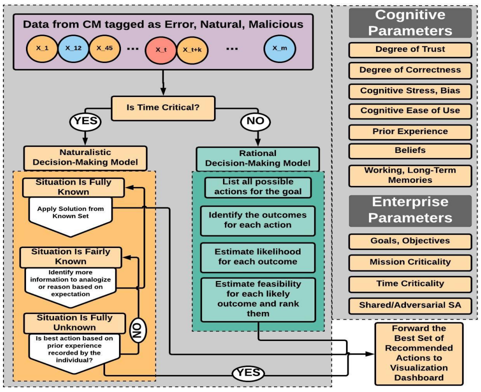

Data from CM tagged as Error, Natural, Malicious
X_1
X_12
X_45
...
X_t
X_t+k
...
X_m
Is Time Critical?
YES
NO
Naturalistic Decision-Making Model
Situation Is Fully Known
Apply Solution from Known Set
Situation Is Fairly Known
Identify more information to analogize or reason based on expectation
Situation Is Fully Unknown
Is best action based on prior experience recorded by the individual?
NO
YES
Rational Decision-Making Model
List all possible actions for the goal
Identify the outcomes for each action
Estimate likelihood for each outcome
Estimate feasibility for each likely outcome and rank them
Cognitive Parameters
Degree of Trust
Degree of Correctness
Cognitive Stress, Bias
Cognitive Ease of Use
Prior Experience
Beliefs
Working, Long-Term Memories
Enterprise Parameters
Goals, Objectives
Mission Criticality
Time Criticality
Shared/Adversarial SA
Forward the Best Set of Recommended Actions to Visualization Dashboard

Fig. 4. Architecture of the Action Module

As shown in Fig. 4, the parameters that contribute to individual or shared SA can be grouped into two sets: *cognitive* parameters- the traits that define the mental model of and differentiate the human users summarized in Table II. It captures how different users might respond to the same situation differently; and *enterprise* parameters- the traits which define the utility's predefined expectations from the users. These expectations might differ across organizational units, job profiles, and can be disrupted when the time or mission is critical. These parameters are also fostered by the utility policies and governance rules. These two sets of parameters enrich the decision-making model, making it unique to each individual or team, thereby leveraging the maximum potential of DM and CM and catering it best to the needs of the user(s).

A. Naturalistic Decision Making (NDM)

Its main objective is to describe how people make decisions in real-world settings under time-critical situations, where cognitive parameters like degrees of trust and correctness, cognitive stress, and prior experience are considered [24]. Specifically, Recognition-Primed Decision (RPD) is one popular model to describe how people make effective decisions using their experience, which can be categorized into two parts including *the situation recognition* and *the solution generation*. For situation recognition, the module acquires the most important features from the current situation and then compares them with corresponding features saved in its working or short-term memory based on the past experiences. The situation can be confirmed if the totally same features can be founded in the memory parts. The corresponding solution is then recommended. If it is a partial match, the module seeks more information or reassesses the situation until it secures a match. However, if the situation is completely unfamiliar (no match), the module checks for the availability of a best action in its long-term memory that could have been recorded in distant past. If there is no match there, it elicits more information until it either discovers a match in long or short-term memory and recommends the associated decision.

Based on the identified situations in the module's memory, the current situation evaluation and assessment will first produce the most relevant cues, which can be implemented to

TABLE I
THE DIFFERENT ALGORITHMS OF THE TRI-MODULAR FRAMEWORK.

Module Name	Algorithm(s) Included	Tool(s)	Evaluation Methods	Performance Description

Data (DM)	Multiple imputation;
	Linear correlation;
	Linear/polynomial regression;
	Maximum Likelihood Estimator
		Kafka;
		Apache Spark;
		R;
		Tableau
			Null Hypothesis Testing;
			Pearson Correlation;
			PDF Comparison;
			Little's Test;
			Cohen's Distance Test
				$H_0 : \mu_o = \mu_i$
				(mean equals for original and imputed data);
				$\rho_{o,i} = cov(X, Y)/(\delta(X)\delta(Y))$;
				$Pr[a \leq X \leq b] = \int_a^b f_X(x)dx$;
				$t = (\overline{X} - \mu)/(\delta/\sqrt{n})$;
				$d = (\mu_o - \mu_i)/SD$
				(SD is standard deviation for samples)

Classify (CM)	LSTM	TensorFlow	MSE & MAPE; Confusion Matrix	$M = (100/n) \times \sum_{t=1}^{n} \lvert(A_t - F_t)/(A_t)\rvert$; True Positives (TP) and True Negatives (TN)

Action (AM)	NDM and RDM	MATLAB;
		Compendium;
		FreeMind
			Precision;
			Timeliness and Recall
				$\lvert\{relevant\} \bigcap \{retrieved\}\rvert / \lvert\{retrieved\}\rvert$;
				$\lvert\{relevant\} \bigcap \{retrieved\}\rvert / \lvert\{relevant\}\rvert$

summary the situation in high-level. The expectancy can also be derived to measure the accuracy of the current situation evaluation [25]. In addition, the expectancy derived in the current situation will be compared with the expectancies stored in the long or short-term memory. The current situation will be classified into a false if the derived expectancy is less than the stored expectancies. Therefore, the more information is needed for the current situation evaluation. Finally, the module implements mental simulations to experiment actions derived from the recognized situation. Due to time-criticality, they might not consider all cognitive and enterprise parameters.

B. Rational Decision Making (RDM)

It is used for generating optimal actions based on current situation when timeliness of the decision is not critical [26]. Note that timeliness in this case is only relaxed in comparison to NDM but not eliminated. It consists of: 1) *monitoring process*, which collects the data, in this case, the Malicious data from HDFS, and 2) *decision process*, which converts current expectations based on collected measurements into an action selection using the stochastic control theory.

To understand the course of an optimal decision process, we define the deadline respect to go-trials as D_t, a cost function on each trial to be c_c per unit time, a penalty for choosing to respond on a stop-signal trial as c_p. If the trial termination time is denoted by τ with $\tau = D_t$ when no response is taken before D_t, and $\tau < D_y$ otherwise. The optimal decision policy intends to minimize the average loss:

$$\begin{aligned} L_\pi =& c_c(\tau) + c_p r P(\tau < D_t | s = 1) + (1 - r)P(\tau = D_t | s = 0) \\ &+ (1 - r)P(\tau < D_t, \delta \neq d | s = 0) \end{aligned} \tag{2}$$

Since minimizing L_π over the policy space directly is computationally intractable, the dynamic programming provides an iterative relationship in terms of the value function (defined as cost here) where a ranges over all possible actions:

$$V^t(b^t) = \min_a [\int p(b^{t+1} | b^t; 1) V^{t+1}(b^{t+1}) db^{t+1}], \tag{3}$$

C. Applying NDM and RDM to AM

As defined earlier, NDM and RDM heavily rely on two sets of parameters: **a) Cognitive:** it includes the *degree of trust* ($\in [0, 1]$)- influenced by data accuracy, completeness, and availability; the *degree of correctness* ($\in [0, 1]$)- quantifies data consistency and plausibility; *stress*- predetermined tasks that users must perform in a given time window; *bias*- partiality that users might show to address specific tasks before others; *ease of use*- user's level of comfort in interacting with the modules; *prior experience*- a catalog documenting responses to different situations in the past; *belief*- user's personal judgment and evaluation of specific tasks; and *memory*- most frequently accessed actions in the short-term and archived actions in long-term; and **b) Enterprise:** it includes *goals* and *objectives* defined for the users, the criticality of events to the *mission*, the *timeliness* of response warranted, and the shared and adversarial *natures of SA*. Unlike the content displayed by IVE which is standardized, the information from AM will be subjective, trained to improve the performance of users.

VI. RESULTS AND DISCUSSION

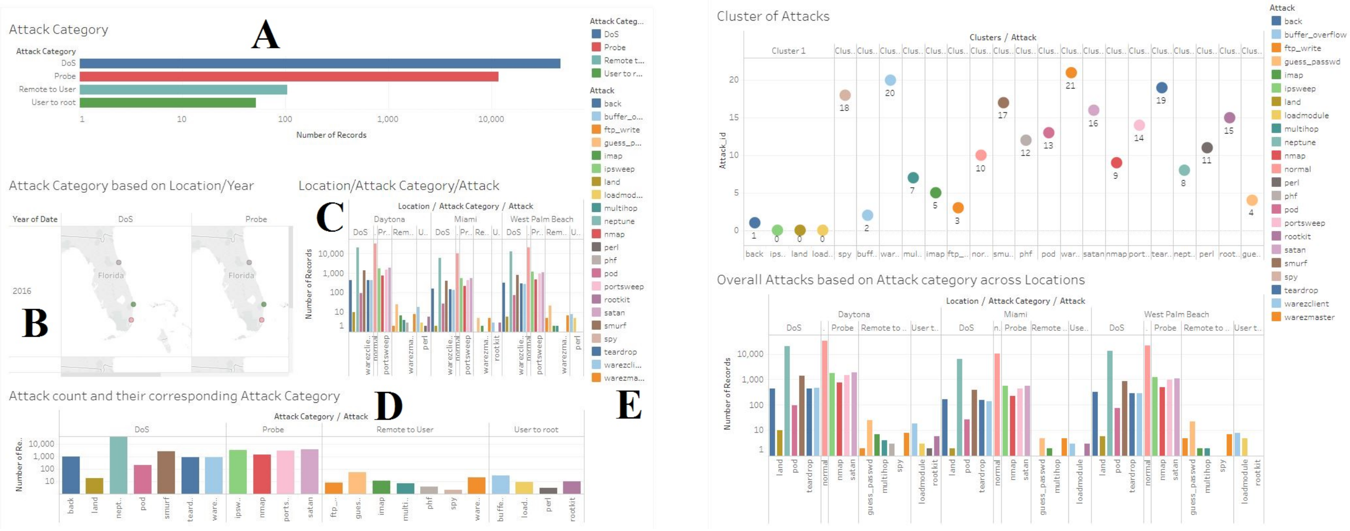
It would be key to first consider the different users in a utility CCC who would benefit from the proposed framework (Table II). In this section, a proof for the framework is shown by configuring and integrating the four engines of DM. The input data is the logs captured by the IDS of an electric utility's enterprise network. A utility CCC replica located at the authors' facility, equipped with high-end processing capabilities was used. The DM can be implemented in different ways as summarized in Table III. While a full-fledged infrastructure at a utility CCC might employ multi-node clusters with requirement to process time-series data on the fly, this paper takes the first step by implementing DM on a single-node cluster with batch processing capabilities.

A. Batch Data Preparation

Data generated by the IDS of an electric utility was taken from an online repository. The original data was obtained as

TABLE II
The Different Potential Users of the Tri-Modular Framework.

Category	User Title	User Role	Framework's Benefit	Module
OT	Operator	Initiating incident response at physical realm, system alarm diagnosis, grid voltage management, preventive maintenance scheduling, system monitoring during storms, interact with reliability coordinators and system operators	Better coordinated response from IT teams in the event of suspicious activities on the grid, well-informed insight into the cause behind high-priority or mission-critical incidents	DM, CM
OT	Dispatcher	Crew dispatch and tracking to restore outages, updating Outage Management System, update member account and meter information, coordinate with operators to schedule tickets prior to crew repairs	Improvised outage log data management and processing for easier analysis and decision-making	DM
IT	Security Analyst	Vulnerability assessment of software, hardware and network, recommendation of solutions and best practices, incident diagnosis, security policy compliance	Prioritization of detected vulnerabilities and recommended solutions to recover damaged data or assets	DM, AM
IT	Security Engineer	Monitoring logs, forensic analysis, incident detection and response, investigation of new technologies to enhance security	Can leverage functionalities to determine context across heterogeneous datasets that will expedite monitoring and analysis	DM
IT	Security Architect	Design of security infrastructure and its components	Lower interoperability challenges helps in adapting design to utility needs	All
IT	Security Administrator	Installation and management of security systems of the enterprise	Little to no new security systems need to be managed or installed	All
IT	Security Specialist	Any of the above, protection against malware, record-keeping of prior incidents, attack vectors and threat actors	The framework assists them on conducting such tasks at a faster pace, thereby reducing their stress	AM



(a)

(b)

Fig. 5. Results of the DM dashboard: **(a)** Main dashboard showing the distribution of various data-points sorted by attack categories, the locations where these attacks were observed, and the time of year when they were observed; **(b)** A zoomed-in view of results from a clustering analysis conducted on incoming data points, along with the frequency of occurrence of such clusters across locations

TABLE III
Mapping of Implementation Scenarios

Cluster Environment	Data Flow	Scenario Preference
Single-node	Batch time-series	Previous work
Single-node	Stream time-series	Current work
Multi-node	Stream time-series	Envisioned Goal

raw comma separated text files with the following attributes: the number of records with similar signatures, the type of attack category, the source and destination bytes, the duration and flag, and the communication protocol used (*tcp* or *udp*). To demonstrate better processing capabilities, a few other attributes were synthesized and incorporated into the dataset. The introduction of these synthesized values was randomized to ensure no bias. The total number of records was divided into equal sets of three, each comprising $41,991$ records. Specific location tags were associated with each set: Daytona, Miami and West Palm Beach. In each set, a vector of hourly timestamps from Jan 01 to Dec 31, 2017 were assigned randomly to the records ensuring every timestamp was utilized at least once. Finally, using the documentation available in the literature for each attack type, an attack category was defined. For example, attacks like *portsweep* and *satan* were grouped as *Probe* attacks, while *neptune* and *smurf* attacks were considered as *Denial of Service (DoS)*.

B. Discussion

Figure 5 comprises two renditions of DM's IVE realized using Tableau and R-server. The primary view of the dashboard at a time-instant is shown in Fig. 5a, which has five major divisions, A through E. Division A represents a histogram bar-chart of the different primary attack categories found in the dataset. This gives the users a quick idea about which attack category is more prominent in the environment at that instant. Division B shows the distribution of these attack categories across three locations for 2016 and 2017. Division C displays the frequency of occurrence of attack types within each category for each location. Division D identifies the magnitude of prevalence among different attack types for each category. Division E keeps the legend of all color-coded attack types and categories active in the view. It can be seen that the organization of information is structured to first acquaint the users' minds with a bigger picture and then drill down to finer details only on a need-to-know basis which is enabled by mouse clicks and hovers. Upon clicking and panning, more complex analyses can be performed. Figure 5b appears when zoomed into Division C, where clustering analysis is conducted to group data into different attack categories.

VII. Future Work and Conclusion

This paper described a detailed architecture of the framework's three modules, DM, CM and AM, detailing how they can be realized and which human users within the utility CCC would be the potential users. This framework reduces the cognitive gap through its three modules and increases the situation awareness. Significant future work stems from this work. Most utilities employ a comprehensive Syslog based event data collection. JSON-based Elasticsearch-Logstash-Kibana (ELK) stack will be considered to modify DM's IVE. Using Syslog as the raw data bank, it can be configured to forward information to Hadoop for management. ELK stack can also be integrated easily with Hadoop and supports R and Python. While the core architecture of DM is realized, its implementation in a multi-node cluster with stream processing features will be developed. The CM and AM will also be developed and integrated with the modified DM to realize the full framework. Besides Enterprise data, some Field data will be used to explore the capability of DM to handle heterogeneous sources.

Acknowledgments

The material published is a result of the research supported jointly by the U.S. Department of Energy under the Award number DE-OE0000779 and the National Science Foundation under the Award numbers CNS-1553494 and CMMI-1745829.

References

[1] T. Rueters, "Cyberattack that crippled ukrainian power grid was highly coordinated," *CBC News*, vol. 11, 2016.

[2] R. Graf, F. Skopik, and K. Whitebloom, "A decision support model for situational awareness in national cyber operations centers," in *International Conference on Cyber Situational Awareness, Data Analytics and Assessment*, 2016.

[3] T. Pleskac, M. Dougherty, J. Busemeyer, and J. Rieskamp, "Cognitive decision theory: Developing models of real-world decision behavior," *Proceedings of the Annual Meeting of the Cognitive Science Society*, 2007.

[4] N.I.S.T, "National institute of standards and technology (nist) framework for improving critical infrastructure cybersecurity," *NIST*, January 2017.

[5] N.E.R.C., "North american electric reliability corporation (nerc) criticial infrastructure protection compliance standards," *NERC*, 2017. [Online]. Available: http://www.nerc.com/pa/Stand/Pages/CIPStandards.aspx

[6] A. Sundararajan, T. Khan, H. Aburub, A. Sarwat, and S. Rahman, "A tri-modular human-on-the-loop framework for intelligent smart grid cyber-attack visualization," in *IEEE Southeast Conference*. IEEE, April 2018.

[7] M. D. R. Azuma and C. Furmanski, "A review of time critical decision making models and human cognitive processes," in *IEEE Aerospace Conference*, 2006.

[8] U. Ozgur, H. Nair, A. Sundararajan, K. Akkaya, and A. Sarwat, "An efficient mqtt framework for control and protection of networked cyber-physical systems," in *IEEE Conference on Communications and Network Security*. IEEE, 2017.

[9] NERC, "Improving human performance: From individual to organization and sustaining the results," in *North American Electric Reliability Corporation (NERC) Technical Presentation*. NERC, 2012.

[10] NIST, "Nistir guidelines for smart grid cybersecurity revision 1," 2014.

[11] A. Kott, M. Lange, and J. Ludwig, "Approaches to modeling the impact of cyber attacks on a mission," 2017.

[12] J. Qi, A. Hahn, X. Lu, J. Wang, and C. Liu, "Cybersecurity for distributed energy resources and smart inverters," *IET Cyber-Physical Systems: Theory & Applications*, 2016.

[13] W. Matuszak, L. DiPippo, and Y. Sun, "Cybersave - situational awareness visualization for cyber security of smart grid systems," in *Proceedings of the Tenth Workshop on Visualization for Cyber Security*, 2013.

[14] A. Kott, C. Wang, and R. Erbacher, "Advances in information security," *Cyber Defense and Situation Awareness*, 2014.

[15] Y. Zhou, P. Li, Y. Xiao, A. Masood, Q. Yu, and B. Sheng, "Smart grid data mining and visualization," in *International Conference on Progress in Informatics and Computing (PIC)*, 2016.

[16] I.N.L., "Control systems cyber security: Defense in depth strategies," *Idaho National Laboratory (INL) Control Systems Security Center Technical Report*, 2006.

[17] P. Small, "Defense in depth: An impractical strategy for cyber world," *SANS Institute InfoSec Reading Room Report*, 2011.

[18] S.A.N.S., "Defense in depth," *SANS Institute InfoSec Reading Room Report*, 2001.

[19] K. Goodhope, J. Koshy, J. Kreps, N. Narkhede, R. Park, J. Rao, and V. Y. Ye, "Building linkedin's real-time activity data pipeline," *IEEE Data Eng. Bull.*, vol. 35, pp. 33–45, 2012.

[20] P. M. Grulich, "Scalable real-time processing with spark streaming: implementation and design of a car information system," *CoRR*, vol. abs/1709.05197, 2017.

[21] M. Zaharia, T. Das, H. Li, T. Hunter, S. Shenker, and I. Stoica, "Discretized streams: Fault-tolerant streaming computation at scale," in *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*, 2013.

[22] H. Zhao, S. Sin, and B. Jin, "Sequential fault diagnosis based on lstm neural network," *IEEE Access*, 2018.

[23] N. Staggers and R. Nelson, "Data, information, knowledge, wisdom," *Routledge International Handbook of Advanced Quantitative Methods in Nursing Research*, 2015.

[24] G. Klein, "Naturalistic decision making," *Human Factors*, 2008.

[25] Y. Hu, R. Li, and Y. Zhang, "Predicting pilot behavior during midair encounters using recognition primed decision model," *Information Sciences-Informatics and Computer Science, Intelligent Systems, Applications: An International Journal*, 2018.

[26] R. R. P. Shenoy and A. Yu, "A rational decision-making framework for inhibitory control," *Neural Information Processing Systems*, 2010.