Letter Report
TLR RES/DE-2024-003

CHARACTERIZING NUCLEAR CYBERSECURITY STATES USING ARTIFICIAL INTELLIGENCE/MACHINE LEARNING - FINAL REPORT

June 2024

S. Chatzidakis, V. Theos, K. Gkouliaras, Z. Dahm, W. Richards, K. Vasili, T. Miller, B. Jowers
Purdue University

J. Lawrence, J. Hollern
Electric Power Research Institute

D. Eskins, K. Cottrell, A. Kim
U.S. Nuclear Regulatory Commission

Division of Engineering
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555–0001

Prepared as part of the Task Order 31310022P0034, "Characterizing Nuclear Cybersecurity States Using Artificial Intelligence/Machine Learning"

This report does not contain or imply legally binding requirements. Nor does this report establish or modify any regulatory guidance or positions of the U.S. Nuclear Regulatory Commission and is not binding on the Commission.

Page intentionally left blank

PURDUE UNIVERSITY | School of Nuclear Engineering

Final Report

Report submitted to NRC as part of the project:

Characterizing Nuclear Cybersecurity States Using Artificial Intelligence/Machine Learning

Principal Investigator

Stylianos Chatzidakis

School of Nuclear Engineering, Purdue University

West Lafayette, IN

Authors		Reviewed		Document No	
S. Chatzidakis, V. Theos, K. Gkouliaras, Z. Dahm, W. Richards, K. Vasili, T. Miller, B. Jowers (Purdue)		S. Chatzidakis (Purdue), J. Lawrence, J. Hollern (EPRI)		PUNE-NRC-006	
Department	NUCL	Version	1.0	Date	05/30/2024

SUMMARY

There is increased interest from the nuclear industry and nuclear engineering community to explore the applicability of artificial intelligence and machine learning (AI/ML) in the nuclear domain. A successful implementation of AI/ML in the nuclear realm could provide tangible benefits to stakeholders. For example, AI/ML could enhance the detection of operational anomalies, predict system failures before they occur, and optimize maintenance schedules, thereby reducing the risk of accidents and ensuring adherence to strict safety standards.

This report summarizes the methodology, implementation, performance evaluation, and lessons learned of an experimental and computational effort to assess the feasibility of artificial intelligence and machine learning (AI/ML) technologies to characterize cyber events in a nuclear system and to test the main hypothesis of this study: *AI/ML can be feasibly and usefully applied to characterize system states resulting from cyber events.*

The report describes the main hypothesis and research questions supporting the main hypothesis. In addition, a set of criteria and assumptions are established for the identification of an initial set of potential use cases and related procedures. These criteria and assumptions ensure that the selected use case is simple yet representative of real events and states (i.e., normal, abnormal, and cybersecurity) in nuclear facilities with characteristics that support event differentiation, flexibility, practical measurement within the resources and time constraints, and data types expected at a cyber-physical environment. A research methodology is presented which was followed throughout the project implementation and includes problem space definition, data types, algorithms, methodology, and project and computing resources.

Nine potential use cases were identified that include domain knowledge, physical manifestations, data artifacts, and combined operational technology (OT) and information technology (IT) data without being overly complex or unrealistic. Out of the nine total use cases, one was implemented that explores events with physical manifestations that change the system state from normal to abnormal including a combination of cyber events that result in modification of objects (e.g., the outputs of plant instrumentation and controls) and modification of relationships between objects (e.g., the network connectivity among plant components). To achieve this, the use case is comprised of eight events resulting in fourteen different system states (1 normal and 13 abnormal) with 67 OT and 11 IT time series multi-variate signals. Two datasets were collected with a total of (i) 13,400,000 OT and 638,000 IT normal data and (ii) 418,080 OT and 19,800 IT abnormal data. From the two datasets, 14 datasets were created for the purpose of AI/ML model training. The objective is to provide insights on the AI/ML application to nuclear systems and their ability to characterize (i) normal and abnormal system states, (ii) cyber events and other abnormal but not cyber related system states, and (iii) various cybersecurity events. The use case is implemented in a real-world cyber-physical system by leveraging PUR-1, a fully digital research reactor located at Purdue University.

After the use case was implemented and data collected, several AI/ML algorithms were selected and evaluated to provide a comprehensive understanding of their limitations and performance-affecting variables. The implementation and performance evaluation of AI/ML algorithms included the design and development of a dedicated classifier architecture, referred to as the *composite classifier*, consisting of a Boolean combination of three simple binary classifiers (Level 1, 2, and 3), each with a different classification objective. Results are presented for each level using both separate and combined OT and IT data. The overall performance of the composite classifier combining all levels is evaluated and discussed.

An extensive search over a broad range of parameters was performed to identify the optimal set of performance-affecting variables. These include window length, window step, training balance ration, test/validation/train split, and scaling, as well as the best performing AI/ML model developed among the

five selected algorithms (Random Forest, Decision Tree, Support Vector Machines (SVM), Logistic Regression, and Naïve Bayes). Overall, Random Forest models outperformed all other models. It was shown that Random Forest is very robust against various balance ratios and window lengths, with only minimal changes in performance when tested with different values for those variables. For these reasons, Random Forest was selected as the best algorithm to use for characterizing cyber events using a classifier trained with separate OT and IT datasets (in Levels 1, 2, and 3) and with combined OT and IT datasets (in a modified Level 1 that can accept OT and IT simultaneously). A significant advantage of Random Forest was its explainability. The high dimensionality and dynamic nature of the problem space makes visualization of the data difficult, and by analyzing the Random Forest model's structure, we were able to identify several erroneous models as well as the signals that introduced incorrect model behavior. This type of model explainability is not currently feasible with SVM, Logistic Regression, Naïve Bayes, or other AI/ML algorithms explored by this project.

Because of its advantages, a Random Forest model was used as the classifier for all the levels of the final experimental composite classifier. Each Random Forest model was trained with the optimal set of performance-affecting variables: (i) window length 20 seconds, (ii) window step 1 second, (iii) standard scaling, (iv) data training/validation/testing split 60/20/20, and (v) training balance ratio 20. Once trained, each Random Forest model in each level was tested with a balance ratio of 30. The performance in Levels 2 and 3 achieved an F1 score of 100% and the performance at Level 1 an F1 score of 99.7% with six false negatives and zero false positives. Additional tests were performed to further explore the robustness of Random Forest and the different classifier levels for identifying abnormal events. These tests were done with new data that was not part of the original experimental dataset and included additional falsified signals and denial of service (DoS) attacks. The Level 1 Random Forest model was able to correctly identify that the event was abnormal with an F1 score of 86% and 361 false negatives. Level 2's model correctly identified the DoS with zero false negatives or positives and an F1 score of 100%. The Random Forest model in Level 3 showed poor performance with an F1 score of 34% and 5,981 false positives.

For comparison with a different data selection and training approach, a modified level 1 classifier that combined OT and IT data was tested. Random Forest and Decision Tree again outperformed the other algorithms with an F1 scores of 100% with zero false negatives and zero false positives. The F1 scores of SVM, Logistic Regression and Naïve Bayes ranged from 60 to 80%. Although this experiment demonstrated that an approach using combined OT and IT can effectively classify normal vs abnormal events, efficiently differentiating among various events types may be difficult and would likely require a more complex multi-class classifier or the use of a composite classifier approach (as used in this project). Joint OT and IT signals datasets can add other complexities. Typically, OT and IT signals may require additional steps in the dataset construction process, be collected using different monitoring tools/software, have differences in sampling frequency or resolution, be disjoint in time, and be difficult to simultaneously collect during certain events. For example, because a DoS attack can impact network performance, data collection using that network can be impacted or even prevented.

The composite classifier approach was tested for this project and showed improved performance with minimal misclassifications when compared with the individual binary classifiers of Levels 1, 2, and 3 alone. This approach demonstrates that multi-state classification of normal, abnormal, and cyber events can be achieved by combining multiple simpler binary classifiers within a composite multi-layer architecture, and such an approach can potentially outperform the individual classifiers.

Finally, key observations, insights, and challenges are described in Section 6 and potential future research areas are identified in Section 7.

This work was performed to fulfill Task 6 Milestone, “Final Report,” within contract 31310022P0034 “Characterizing Nuclear Cybersecurity States Using Artificial Intelligence/Machine Learning” with the U.S. NRC.

ACKNOWLEDGMENTS

This research was sponsored by the U.S. Nuclear Regulatory Commission and was carried out at Purdue University under contract 31310022P0034.

This report was developed with significant contributions, expert input, and guidance from Jeremy Lawrence and Jason Hollern at the Electric Power Research Institute, and Doug Eskins, Anya Kim, and Kaitlyn Cottrell at the U.S. NRC.

Table of Contents

Table of Figures 7
List of Tables 8
Abbreviations 9
1. INTRODUCTION 10
1.1 Background 10
1.2 Objectives 10
1.3 Definitions 11
1.4 Scope 11
2. MAIN HYPOTHESIS AND RESEARCH QUESTIONS 12
2.1 Main Hypothesis 12
2.2 Research Questions 12
2.3 Research Plan 13
3. ASSUMPTIONS 15
4. USE CASE OVERVIEW 17
4.1 System Description and Boundaries 17
4.2 Representing Plant States and Events 18
4.3 Use Case Progression 19
4.4 Data Collection and Dataset Creation 23
4.5 Data Artifacts 26
5. AI/ML TECHNOLOGIES AND IMPLEMENTATION 29
5.1 Basis for Identification and Selection of Potential AI/ML Technologies 29
5.2 Implementation Methodology 29
6. OBSERVATIONS–INSIGHTS 42
7. FUTURE WORK 47
REFERENCES 48
APPENDIX A - GLOSSARY 49

Table of Figures

Figure 1. PUR-1 facility (left) and remote monitoring workstation (right). 18
Figure 2. Layer numbering convention with example entry points for cyber events. 18
Figure 3. Schematic representation of modes, states, and objects. 19
Figure 4. Schematic representation of system states and events implemented in the use case. 20
Figure 5. Use case progression, events and resulting states, and collected datasets. 22
Figure 6. Variation of power levels and pool temperature during normal operation (normal state). 24
Figure 7. Variation of signals during reactor trip (normal state) 24
Figure 8. Snapshot of signals during abnormal states trip unavailable (left) and FDI #1 (right). 25
Figure 9. Snapshot of signals during abnormal states FDI #2 (left) and FDI #3 (right). 26
Figure 10. Network traffic during high intensity and low intensity DoS attacks. 26
Figure 11. Example of signals with different types of fluctuations. 27
Figure 12. Example of a signals with null values, outliers, and an instrument related artifact. 28
Figure 13. Implemented composite classifier. 30
Figure 14. AI/ML implementation methodology. 31
Figure 15. Effect of train/validation/test split for two different training balance ratios 35
Figure 16. Example of effect of balance in Logistic Regression model performance (Figure 16a, left). Example of effect of changing the amount of abnormal data with constant balance ratio for Random Forest (Figure 16b, right). 36
Figure 17. Effect of window length in AI/ML model performance for two different balance ratios. 36
Figure 18. Computational time as a function of AI/ML model and training balance ratio (left). Relative increase in computational time for Random Forest as a function of training balance ratio and window length (right). 37
Figure 19. Confusion matrix for Level 1 (L1), Level 2 (L2) and Level 3 (L3) for separate OT and IT. 38
Figure 20. Confusion matrix for Level 1 (L1), Level 2 (L2) and Level 3 (L3) with out-of-training data. 39
Figure 21. Confusion matrix for Random Forest using combined OT and IT. 40
Figure 22. Confusion matrix of the composite classifier. 41

List of Tables

Table I. Composite classifier output truth table. 31
Table II. Parameters used for identifying the best performing AI/ML model. 34
Table III. Performance metrics for separate OT and IT. 38
Table IV. Performance metrics for separate OT and IT and out-of-training data. 38
Table V. Performance metrics for combined OT and IT 39

Abbreviations

AI	Artificial Intelligence
BR	Balance Ratio
DoS	Denial of Service
EPRI	Electric Power Research Institute
FDI	False Data Injection
IT	Information Technology
ML	Machine Learning
NIST	National Institute of Standards and Technology
NRC	Nuclear Regulatory Commission
OT	Operational Technology
PI	Principal Investigator
PUR-1	Purdue University Reactor One
ROC	Receiver Operating Characteristic
SVM	Support Vector Machine

CHARACTERIZING NUCLEAR CYBERSECURITY STATES USING ARTIFICIAL INTELLIGENCE/MACHINE LEARNING

FINAL REPORT

1. INTRODUCTION

The project aims to experimentally assess the feasibility of using artificial intelligence and machine learning (AI/ML) technologies to characterize (i) normal and abnormal system states, (ii) cyber events and other abnormal system states, and (iii) various cybersecurity events. The project comprised five tasks: research plan development (Ref. 1), identification of representative use case (Ref. 2), identification of AI/ML technologies (Ref. 3), use case implementation (Ref. 4), and performance analysis and gap analysis (Ref. 5). This report summarizes the methodology, implementation, performance evaluation, and lessons learned of the project. Details on each project task can be found in individual task reports labeled TLR-RES/DE-2024-003 (a) through (e).

1.1 Background

While humans excel at complex identification and decision-making problems, recent advancements in computational capacity coupled with advancements in AI/ML have led to faster, more reliable transient identification and online monitoring systems - the progress of self-driving vehicles serves as a clear example. Of course, an AI/ML nuclear reactor state or event characterization system poses a significant engineering challenge: it would need to be guaranteed to properly characterize a large number of possible operational scenarios and events without jeopardizing the safety of the reactor. Even though the deployment of such systems is many years off, developing the technical basis, ideally using a simplified, low-risk nuclear reactor such as PUR-1, would be an important first step toward understanding the feasibility and practicality of AI/ML architectures for future applications to nuclear plants.

There has been a large body of research over the years by the nuclear community on the development of online monitoring tools, fault detection, diagnostics, prognostics, etc. Applications of AI/ML to reactor problems have focused on identifying transients, predicting parameters such as optimal fuel loading pattern, fuel temperatures, etc. Most successful studies have relied on widely used AI/ML tools such as Support Vector Machines (SVM), Fuzzy Logic, Bayesian Statistical Learning, and Artificial Neural Networks. The algorithms are typically trained using data developed from system codes such as Monte Carlo N-Particle (MCNP) or RELAP5. However, application and demonstration of such approaches to actual nuclear environments with real data and noise artifacts has been limited.

In this project, we aim to leverage this publicly available knowledge base and implement AI/ML algorithms proposed for use in monitoring and control systems for nuclear and industrial applications. In this project, it is not our intention to develop new algorithms but to identify and use what is known to work and demonstrate applicability to characterization of nuclear states and cyber events in PUR-1, a fully digital research reactor located at Purdue University.

1.2 Objectives

The project goal is to experimentally assess AI/ML tools and explore their ability to characterize nuclear states under prototypic conditions in PUR-1. To accomplish this goal, we performed the following:

(i) We identified and implemented a use case that is simple yet representative of real events and states (i.e., normal, abnormal, and cybersecurity) in nuclear facilities and with characteristics that support

event differentiation, flexibility, practical measurement within the resources and time constraints, and data types expected at a cyber-physical environment.

(ii) We identified and used AI/ML algorithms that were trained on real-time OT and IT data collected from PUR-1.

(iii) We evaluated the performance of the AI/ML models to characterize different system states.

This research aims to establish knowledge and insights to inform future regulatory bases for the nuclear domain application of AI/ML, especially with respect to nuclear powerplant cybersecurity.

1.3 Definitions

Definitions specific to the project scope are listed in Appendix A to facilitate a more precise formalism and clearer representation of the terms used throughout this project's problem space. Effort has been made to align these definitions with the more generic NIST cybersecurity definitions (Ref. 6) and U.S. NRC AI Strategic Plan (Ref. 7) definitions.

1.4 Scope

The scope of this project includes characterization of known system states that could potentially occur in a nuclear environment using open-source AI/ML algorithms. The scope does not include development of new AI/ML algorithms, use of proprietary AI/ML algorithms or deep learning, or evaluation of system vulnerabilities or their consequences.

No assessment is performed to evaluate the likelihood or risk of a use case or a potential attack progression. It is emphasized that nuclear systems have multiple defense-in-depth mechanisms and even though some use cases may be plausible they may be unlikely to occur in a nuclear system.

2. MAIN HYPOTHESIS AND RESEARCH QUESTIONS

This section describes the main project hypothesis and research questions.

2.1 Main Hypothesis

The overarching goal of this project is to test the following hypothesis:

Hypothesis: *AI/ML can be feasibly and usefully applied to characterize system states resulting from cyber events.*

Basis: Recent advances in AI/ML have significantly expanded its ability to process and analyze large volumes of data, recognize patterns, and make predictions or decisions based on data. AI/ML algorithms have been used in the cybersecurity and other engineering domains to detect anomalies or deviations from normal system behavior, which could indicate a cyberattack or system compromise. There is increased interest from the nuclear industry and nuclear engineering community to explore the applicability of AI/ML in the nuclear domain. A successful implementation of AI/ML in the nuclear realm could provide tangible benefits to stakeholders. For example, AI/ML could enhance the detection of operational anomalies, predict system failures before they occur, and optimize maintenance schedules, thereby reducing the risk of accidents and ensuring adherence to strict safety standards. From a regulatory standpoint, the application of AI/ML in the nuclear domain might require a new approach to nuclear safety and compliance monitoring.

2.2 Research Questions

The following research questions need to be addressed to test the main project hypothesis:

Q1: *Can abnormal events, such as equipment malfunctions and cyberattacks, result in distinguishable system states?*

Justification: There are various types of abnormal events (e.g., malfunctions or cyber events) with differing intensities and complexities which can lead to different physical or non-physical manifestations in a cyber-physical system. This question aims to explore whether these events can result in distinguishable system states that can then be classified using an AI/ML system.

Q2: *Can system states be represented with IT and OT time series data?*

Justification: A cyber-physical system generates both IT and OT time series data. This question aims to explore whether IT and OT data can used to represent system states and provide novel insights into the characterization of states resulting from a cyberattack.

Q3: *How should the data be collected and processed for input suitable to an AI/ML system?*

Justification: IT and OT data generated in a cyber-physical system include data with different characteristics and noise artifacts that are not necessarily suitable for input to an AI/ML system. This question aims to explore how data with different characteristics should be collected and processed to a format suitable for input to an AI/ML system.

Q4: *Can an AI/ML system trained with data extracted from normal and abnormal system states be used to characterize those states?*

Justification: This question aims to investigate whether an AI/ML system trained with data extracted from normal and abnormal system states can characterize those states, what classification architecture would be suitable for such a task, and what training, validation, and testing methodology would need to be followed.

Q5: *Can an AI/ML system differentiate between cyber events and other non-cyber events?*

Justification: Abnormal system states that result from unacceptable system conditions due to cyber events or causes other than cyber events (e.g., equipment malfunction, corrosion buildup, leakage, or wear) may have similar physical manifestations. This question aims to explore whether an AI/ML system can differentiate between cyber events and between a cyber event and other non-cyber events.

Q6: *Can AI/ML models be combined to perform multi-state classification?*

Justification: It is unlikely that a single AI/ML model would be able to characterize all system states. Different AI/ML models may need to be trained and the combined to perform multi-state classification. This question aims to explore whether this is feasible.

Q7: *What are the limitations of such an AI/ML system, its performance, and what variables can affect its performance?*

Justification: To support the main hypothesis, a comprehensive understanding of the limitations and performance-affecting variables of an AI/ML system is fundamental to ensure their effective, safe, and reliable application to the characterization of cyber events. This necessitates the exploration of the limitations of the AI/ML system used for characterization of cyber events, whether the AI/ML system can identify those events, its performance, and what variables can affect its performance.

2.3 Research Plan

To test the main hypothesis and answer the supporting research questions, the following research plan was developed and followed throughout the project:

Identification of representative use case: Identify a use case and related procedures that are representative of real events (i.e., normal, abnormal, and cybersecurity) in nuclear facilities. Describe assumptions, event progression, the experimental system, and experimental procedures in detail including components/systems and computational models. The use case will be selected with characteristics (i.e., events or data) that support event differentiation, flexibility, and practical measurement within the resources and time constraints, and data types expected within a cyber-physical environment. A use case that can generate novel insights and includes physical process, communication and monitoring data without being overly complex or unrealistic will be preferred to the extent possible.

Identification of AI/ML technologies: Provide a description of AI/ML technologies, the technical approach, and the methodology/reasoning behind the determination of a suitable AI/ML technology for a given use case. The description will include a discussion of the strengths and weaknesses of the technologies and approach relative to the research objectives and use case problem space (i.e., resources, time constraints, and data types available in a cyber-physical system) and include such factors as applicability, capability, cost, and risk where appropriate.

Technical approach: Develop a technical approach for: (i) performing experiments related to the use case, (ii) representing plant events including cyber events and/or the system states that result from those events, (iii) characterizing, measuring, and visualizing system states including those that result from cyber events and other system events, and (iv) using AI/ML technology and system state information to identify, characterize, and distinguish types of system events including cyber events.

Use case implementation: Implement and test the use case. Apply the identified and selected AI/ML tools to characterize various system states. The AI/ML tools will be trained using data representative of the use

case. Once training is performed, the AI/ML tools will be tested under previously unseen conditions. The obtained data will be used for performance evaluation and gap analysis.

Performance evaluation and gap analysis: Evaluate the performance of the AI/ML tools and extract insights and fundamental knowledge about nuclear applications of AI/ML to cybersecurity and the AI/ML use case. Various performance metrics (e.g., error quantification, receiver operating characteristics, false positives vs. false negatives, confusion matrix, or computing cost) will be used to quantify performance. Use the performance evaluation to answer the following questions: (i) Can AI/ML be practically applied as a tool to identify, characterize, and distinguish among nuclear plant states resulting from normal operations and events including cyber events? (ii) What methods can be used to evaluate a AI/ML technique for a given use case? (iii) What affects the accuracy and reliability of the tool and how can such effects be determined or measured? (iv) What data is needed to train and maintain the tool? Provide a discussion on lessons learned and areas for improvement (e.g., classes of events for which AI/ML implementation may not be potentially applicable given current level of associated resources and/or current state of technology).

Project summary, observations, and insights: Summarize the key activities of the project. Provide observations and insights concerning the project goals, use case, and approach. Provide recommendations for future work.

3. ASSUMPTIONS

The following assumptions were used to guide the project implementation. The basis for each assumption and the approach followed to address these assumptions are discussed below:

A1: *It is assumed that there is no prior knowledge or results on the performance of the AI/ML algorithms for the chosen use case.*

Basis: Data obtained from a nuclear system would be unique and it is assumed that pre-trained AI/ML models or models trained on one system would not be transferable to another system. Experience with PUR-1 has shown that data artifacts are very sensitive to system configuration, which would be very difficult to reproduce in another system. In addition, even though there is prior work on AI/ML for anomaly detection, specific applications to nuclear environments have been limited and the results are not expected to be easily generalizable. This assumption allows for an unbiased approach in developing and selecting AI/ML models, ensuring that the implementation is guided solely by the data and observed outcomes rather than preconceived notions about algorithm performance.

Approach: In this work, five AI/ML algorithms (Random Forest, Decision Tree, Support Vector Machines, Logistic Regression, and Naïve Bayes) were trained, and the resulting AI/ML models were tested on the created datasets. The AI/ML models with best performance were then selected for use in a composite classifier designed for characterizing cyber events. The goal is not only to identify the best AI/ML models but also to provide insights on model robustness and identify which parameters and what factors are most significant in the selection process.

A2: *It is assumed that the normal to abnormal data ratio will be imbalanced.*

Basis: This assumption reflects real-world scenarios in nuclear systems where abnormal events are expected to be much less frequent than normal events.

Approach: In this work, additional performance metrics, beyond the commonly used accuracy, which account for data imbalance, were used to provide a holistic assessment of AI/ML model performance. The performance metrics used are F1-score, precision, and recall. The true positives, true negatives, false positives, and false negatives are represented using confusion matrices. These metrics are defined in Section 2. It is noted that abnormal events are classified as “positive” per conventional practice in the computer science domain.

A3: *It is assumed that the exact balance ratio is not known.*

Basis: In an ideal scenario, training and testing should be performed with the true balance ratio (i.e., true proportion of classes). However, in practice, it can be challenging to determine or know beforehand the exact distribution of classes. Even if the exact balance ratio is known, it might be difficult to collect adequate data. This assumption would favor selection of a robust model that can handle varying degrees of imbalance to the extent practical.

Approach: In this work, since the exact balance ratio is unknown, tests were performed with varying balance ratios to identify potential tradeoffs and other effects in AI/ML model performance. The balance ratio for training the AI/ML algorithms varied from 1 (equal size of normal and abnormal data) to 30 (normal data is 30 times more than abnormal data). While testing it was kept constant to 30 to represent a highly imbalanced ratio to the extent practical and within project constraints. Additional tests were performed with a constant training balance ratio of 30 but with varying abnormal data size (i.e., starting from very limited

abnormal data to a size that is adequate for reasonable AI/ML training). The goal is to provide insights and identify potential performance issues when training is performed using incorrect balance ratios.

A4: *It is assumed that misclassifying abnormal events (i.e., false negatives) caries higher weight than false positives.*

Basis: The cost of missing an abnormal event is expected to be greater than the cost of incorrectly flagging a normal event. For example, not detecting a cyber event could be costlier than investigating a false alarm (note: cost herein does not necessarily mean economic cost). This assumption would favor selection of models with as few false negatives as possible.

Approach: In this work, the dependence of false negatives on various input parameters was evaluated. The models with the best F1-score were selected for use in the Level classifiers. It is noted that this approach is not unique and that models that optimize other metrics or minimize the number of false negatives (but reducing the F1 score) could be selected and applied.

A5: *It is assumed that there is capability to obtain adequate abnormal data but much less than the amount of normal data that can be collected.*

Basis: This assumption takes into account the practical limitations of data collection, given the cost and safety risks associated with collecting abnormal data in nuclear systems and the difficulty in producing adequate synthetic data representing abnormal events.

Approach: In this work, the size of normal data was varied as it was much easier to obtain normal data than abnormal.

A6: *It is assumed that amount of normal data available to develop models will vary based on application.*

Basis: This assumption is representative of real-world nuclear systems where the amount and quality of normal data can be highly dependent on the signals, systems, and states represented. For example, transient-state data may be rarer than steady-state operational data.

Approach: Exploring different training set sizes and balance ratios develops a better understanding of potential impacts on model performance, such as the ability to generalize. It can also provide insights on the minimum amount of data needed for effective model training. In this work, tests were performed with different training sets, ranging from 50% to 70% of the total datapoints, and balance ratios with various abnormal data sizes to evaluate any effects on the performance of the AI/ML models.

4. USE CASE OVERVIEW

The use case selected is suitable for testing the main hypothesis while leveraging domain knowledge, physical manifestations, OT data, and IT data without being overly complex or unrealistic. The use case explores events that change the system state from normal to abnormal including a combination of cyber events that result in the modification of objects (e.g., false data injection) and the modification of relationships between objects (e.g., denial of service). The use case is implemented in a real-world cyber-physical system by leveraging PUR-1, a fully digital research reactor located at Purdue University.

The use case starts by initiating a manual trip during reactor operations at power. Normally, once a manual trip is initiated, the magnets holding the control rods are deactivated, all the control rods are inserted in the reactor core, and the large negative reactivity inserted by the rods rapidly shuts down the reactor. However, in the use case, a failure of the actuation of the reactor trip is simulated, representing either a system malfunction event or the unauthorized action of a malicious actor (i.e., a cyber event). This base cyber event is implemented both in isolation and in combination with additional cyber events. The additional cyber events explored are false data injection (FDI) and denial of service (DoS) cyberattacks. The FDI attack aims to modify system objects (e.g., the outputs of plant instrumentation and controls). The DoS attack aims to modify the relationship between objects (e.g., the network connectivity among plant components).

Why this use case?

The use case meets the following criteria:

- The use case is simple and at the same time representative of a cyber-physical system.
- The use case allows for differentiation between system states.
- The use case includes both OT and IT data.
- The use case represents physical process manifestations.
- The use case includes events that may be result in (i) the modification of objects, (ii) the modification of object relationships, or (iii) a combination of (i) and (ii).

More details on these criteria are available in Section 2.2 of the Task 4 report (Ref. 4).

4.1 System Description and Boundaries

The use case was implemented entirely in PUR-1. The PUR-1 facility, cyber-physical architecture and boundaries are briefly described below.

PUR-1 facility

PUR-1 is a pool-type research reactor with a fully digital instrumentation and control system that allows for remote monitoring and the collection of more than 2,000 parameters. The reactor core is surrounded by graphite moderators and located at the bottom of a cylindrical pool. The reactor has a rectangular parallelepiped core and consists of 16 assemblies. Twelve out of sixteen are fuel assemblies, the rest are two shim safety assemblies, a fission chamber assembly, and a regulating rod assembly. Fuel assemblies consist of 12 or 13 fuel plates, including two or one dummy plate(s), respectively. The shim safety assemblies contain eight fuel plates and control rods which are made of borated stainless steel. The regulating rod is made of hollow stainless steel filled with water. Each fuel plate consists of low enriched uranium and Al cladding. The PUR-1 facility and a remote monitoring station used for data collection are shown in Figure 1.

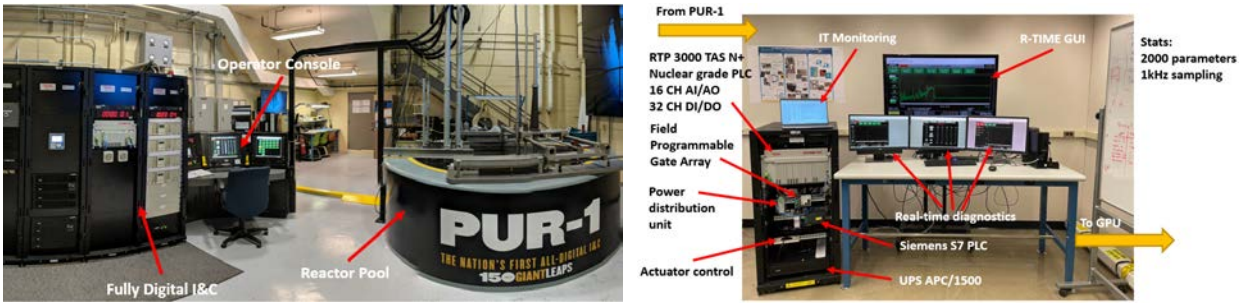

Figure 1. PUR-1 facility (left) and remote monitoring workstation (right).

Cyber-physical system architecture and boundaries

The cyber-physical system architecture includes five layers with various objects (e.g., equipment, components, or controllers), and relationships between objects (e.g., network connections with communication protocols and different signal flow requirements). Layers 0 to 3 are physically located in PUR-1 while Layer 4 is the outside network (also known as "business" or "remote") and is in a separate building. Layers 3 and 4 are connected via Ethernet/TCP-IP. IT security measures (e.g., firewalls or data diodes) are implemented between Layers 3 and 4. Physical OT signals and measurements for training, validation, and testing of AI/ML models are generated from Layers 0 to 3, while IT data is generated in Layers 3 and 4. All data collection, computations, and analysis (including model training, validation, and testing) takes place at Layer 4. Finally, an "attacker" PC is connected via an Ethernet switch to Layer 4 for introducing cyber events (e.g. DoS attack). An example of the Layering convention can be seen in Figure 2, along with possible entry points for cyber events.

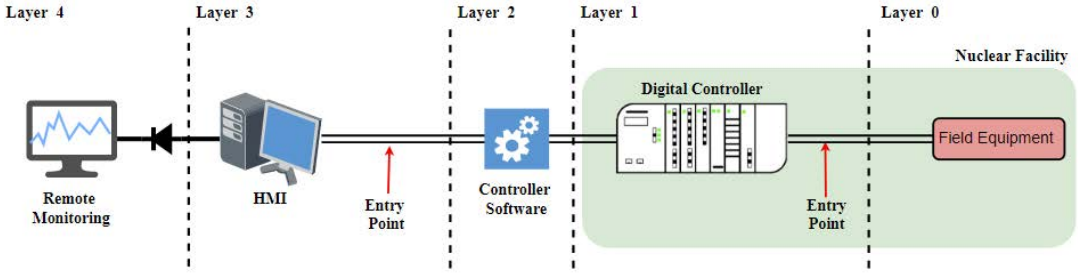

Figure 2. Layer numbering convention with example entry points for cyber events.

4.2 Representing Plant States and Events

It was found early in the project that a large number of possible events and states could render the use case complex and difficult to follow and describe in a coherent manner. A formalism is introduced to facilitate representation, in a simple way, of events and the system states that result from those events. This also facilitates representation of datasets collected and the states they correspond to.

In this formalism, there are four main components: mode, state, objects, and relationships between objects. This is schematically represented in Figure 3. The mode describes the condition (operational or not) of a system. For example, a nuclear system can have multiple modes, (e.g., startup, power operation, standby, refueling, or hot shutdown). A mode may be inclusive of multiple possible nuclear system states. A nuclear system state maps to a specific set of plant object and object relationship conditions. Any event that affects an object or a relationship between objects can result in state change. It is noted that if an event is intentional

and unauthorized then it is categorized as cyber event. Finally, objects can be any system component, equipment, instrument, data, or data connection. Objects can be interconnected and communicate with other objects. Any two or more objects can be interconnected and a change in one object may affect another object depending on the relationship between them. Relationships between objects can be one-way or two-way. For example, a plant controller (object) may communicate over a two-way Ethernet connection (relationship) to send or receive information from or to an actuator (object).

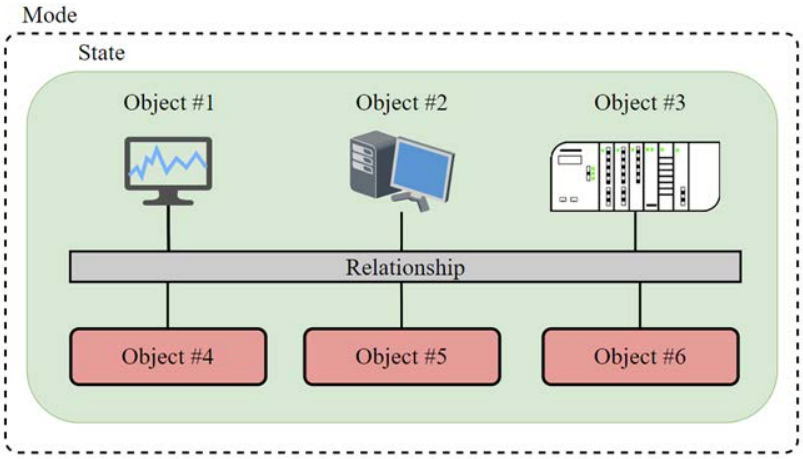

Figure 3. Schematic representation of modes, states, and objects.

4.3 Use Case Progression

The following assumptions were considered when implementing the use case:

- It is assumed that an adversary has access down to Layer 1 (but not Layer 0).
- It is assumed that an adversary has no knowledge about the AI/ML system.
- It is assumed that the AI/ML models are located on a workstation in Layer 4 and that AI/ML model training and data processing takes place in a workstation located in Layer 4.
- It is assumed that an AI/ML model receives the same data and information as the reactor operators (Layer 3).

More details on these criteria are available in Section 2.5 of the Task 4 report (Ref. 4).

Use case progression

The use case starts with the operator attempting to trip the reactor by pressing the SCRAM button on the console. During normal operation, pressing the SCRAM button cuts the magnet current to the control rods. Within less than a second, the rods are fully inserted and power drops to very low levels. From that point, power is due to heat generated from decaying fission products. Following reactor trip, servo drives and several switches that indicate location and condition of control rods are activated.

Eight events were experimentally simulated. Specifically:

1. Manual trip with trip available (normal state)
2. Manual trip with trip unavailable due to cyber related cause
3. Manual trip with trip unavailable due to other non-cyber related cause
4. False Data Injection of one signal (FDI #1)
5. False Data Injection of two signals (FDI #2)

6. False Data Injection of three signals (FDI #3)
7. High intensity Denial of Service (High DoS)
8. Low intensity Denial of Service (Low DoS)

The DoS attacks were performed using ethical penetration testing software by sending continuous information packets and increasing the network latency which affects the capacity of the system to fulfill requests and could render the system unresponsive to any request. The FDI attacks are implemented by simulating the modification of normal operational data sent to a remote location.

Figure 4 shows the system states (normal and abnormal) scoped for consideration in this project and the events or combinations of events that may result in these states. Normal systems states are those plant states for which the performance of all plant systems is acceptable and include, as a subset, normal states that result from any event where trip remains available, and system behavior is acceptable. Abnormal system states include plant states in which trip is unavailable due to an isolated event, or a combination of cyber or non-cyber events, e.g., DoS, FDI, or malfunction. States resulting from FDI's of increasing complexity were implemented by increasing the number of falsified signals. As a result, FDI #1 is a subset of FDI #2 and FD I#3, while FDI #2 is a subset of FDI #3. Events that occurred simultaneously are shown as overlapping.

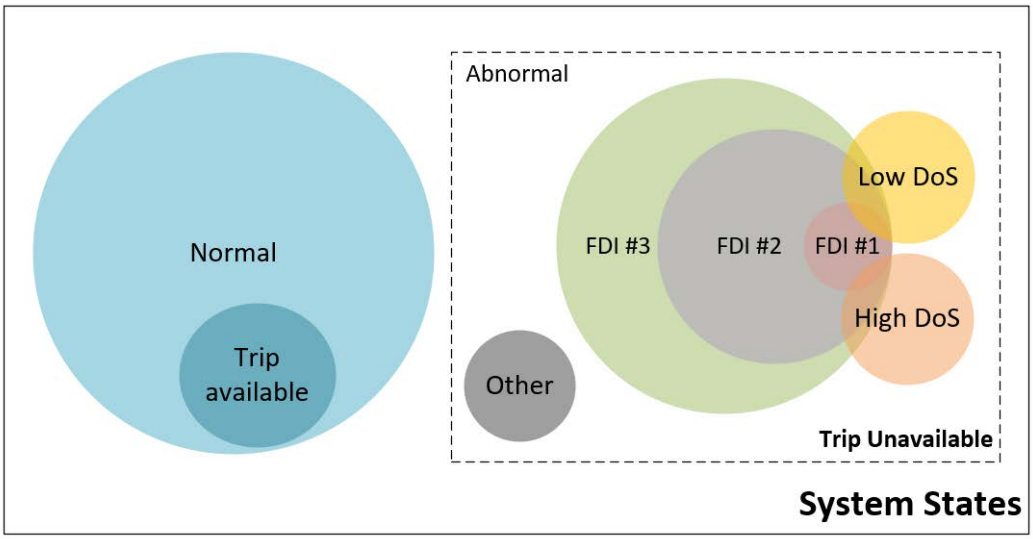

Figure 4. Schematic representation of system states and events implemented in the use case.

These eight events, in various combinations, resulted in the following 14 system states (1 normal and 13 abnormal states). States 2 through 14 represent abnormal states. The use case progression, events and resulting states, collected datasets, including affected objects, are shown in Figure 5. The top row shows the events that take place, and the bottom row shows the objects affected.

1. Normal operation
2. Trip unavailable due to cyber related cause
3. Trip unavailable combined with low intensity Denial of Service (Low DoS)
4. Trip unavailable combined with high intensity Denial of Service (High DoS)
5. Trip unavailable combined with False Data Injection of one signal (FDI #1)
6. Trip unavailable combined with False Data Injection of one signal and low intensity Denial of Service (FDI #1 + Low DoS)

7. Trip unavailable combined with False Data Injection of one signal and high intensity Denial of Service (FDI #1 + High DoS)
8. Trip unavailable combined with False Data Injection of two signals (FDI #2)
9. Trip unavailable combined with False Data Injection of two signals and low intensity Denial of Service (FDI #2 + Low DoS)
10. Trip unavailable combined with False Data Injection of two signals and high intensity Denial of Service (FDI #2 + High DoS)
11. Trip unavailable combined with False Data Injection of three signals (FDI #3)
12. Trip unavailable combined with False Data Injection of three signals and low intensity Denial of Service (FDI #3 + Low DoS)
13. Trip unavailable combined with False Data Injection of three signals and high intensity Denial of Service (FDI #3 + High DoS)
14. Trip unavailable due to other cause (e.g., malfunction)

It is noted that, because the number of states resulting from various possible event combinations can grow exponentially, the researchers selected these 14 states to balance research utility with feasibility, e.g., the capability to explore interesting research questions within the constraints of project resources and time. Other event and state combinations may be useful for future work and this project provides a conceptual framework for identifying and constructing those combinations.

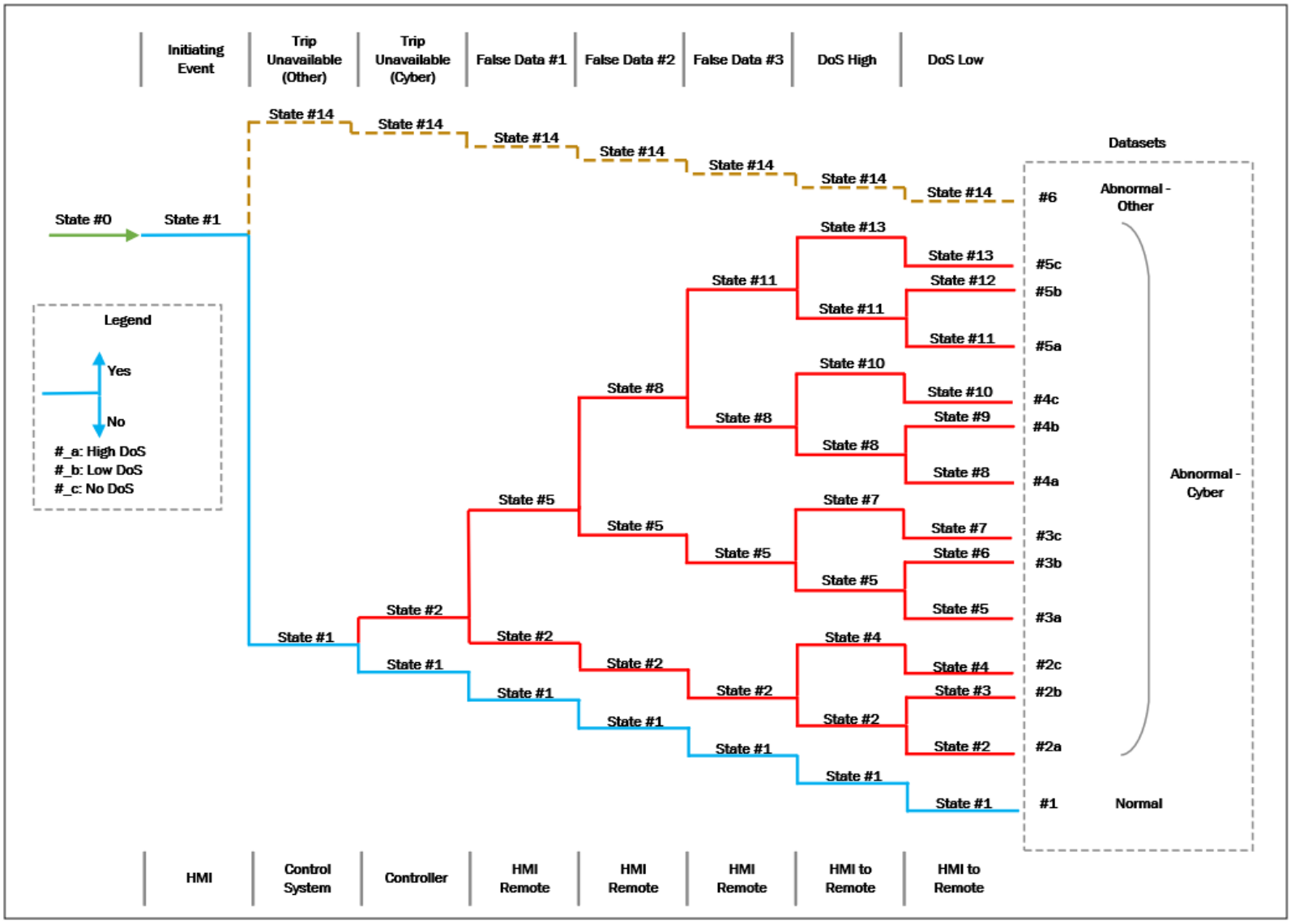

Figure 5. Use case progression, events and resulting states, and collected datasets.

4.4 Data Collection and Dataset Creation

During the use case implementation, 67 (out of 2,000) OT data signals and 11 (out of 1,000) IT data signals were collected. These signals, based on domain knowledge, were deemed to be most relevant to the use case and representative of important system behavior. Signals that did not change regardless of the event taking place or were not directly related to the use case such as room temperature or Heating Ventilation and Air Conditioning (HVAC) status were excluded.

OT data included:

- **Process data**: Data generated from physical measurements and derived quantities generated by system controllers and sensors.
- **Monitoring data**: Data generated from system monitoring such as alerts, alarms, commands, and status indications.

IT data included:

- **Communication data**: Data generated from network traffic such as packet data and other information generated as part of the network transmission process.
- **Host system data**: Data that describes the host system such as resource usage, services, and processes.

All the data collected was dynamic numerical time series data and originated from all system layers. Due to time and scope constraints, other types of data such as metadata or configuration data and other data formats such as categorical or text, were not collected.

Fourteen datasets were created, each corresponding to a targeted system state. Dataset #1 represents normal state (i.e., no cyber or other events) and consists of 13,400,000 datapoints (i.e., single observations of a signal value in the dataset) from the 67 OT data signals and 638,000 datapoints from the 11 IT data signals. The data was collected when system state was normal between August 2022 and June 2023. It was important to capture as many normal variations as possible. As a result, Dataset #1 includes reactor operation at different power levels, ranging from 0 to 100% power, and reactor trips (manual or not) where trip was available. The IT data comes from monitoring network traffic between Layers 3 and 4. Figure 6 shows the variation of power and pool temperature during normal operation. Figure 7 shows the variation of a sample of signals monitored by the operator during a reactor trip. All other datasets, described below, represent abnormal state.

Dataset #2a represents an abnormal plant state resulting from a cyberattack that renders the trip unavailable without other simultaneous cyber events, such as FDI or DoS. This dataset was created by taking normal reactor data (collected separately from the data in Dataset #1) and changing the manual trip signal from zero (no trip) to one (trip initiated). All other data in the dataset is normal for non-trip conditions. For example, the signals changed by a trip (e.g., magnets deactivated and sudden power drop) remain within their normal at power ranges. To generate adequate data for Dataset #2a, the manual trip indication value was changed from zero to one once every 20 seconds for a 1,560 second period. An example is shown on the left side of Figure 8.

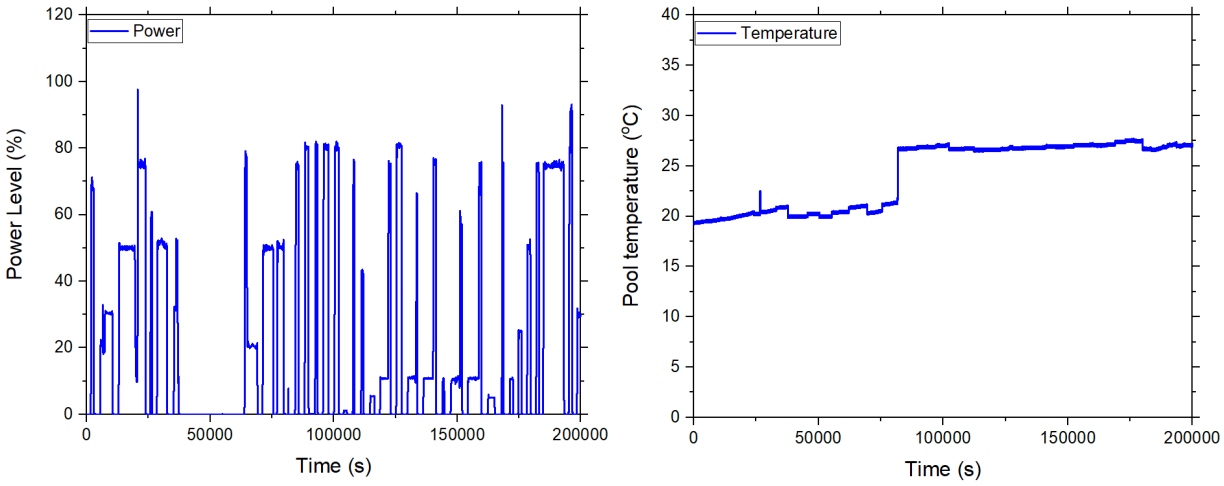

Figure 6. Variation of power levels and pool temperature during normal operation (normal state).

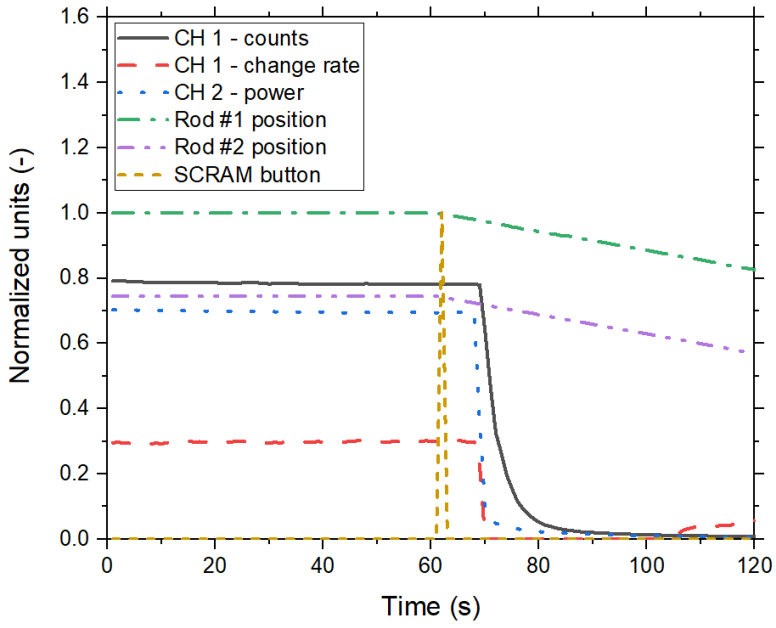

Figure 7. Variation of signals during reactor trip (normal state).

Starting from abnormal Dataset #2a, abnormal datasets #2b through #5c were created. These datasets combine signal values for the additional cyber events of DoS and FDI attacks with the base cyber event of trip unavailability, i.e., Dataset #2a.

Two types of DoS attacks were performed, a high intensity and a low intensity. The DoS events were performed by increasing the number of packets per second transmitted between Layers 3 and 4 using Kali Linux. The DoS attacks each lasted for 900 seconds. Average packet transmission during normal traffic (no DoS), low intensity DoS, and high intensity DoS were 30, 870, and 24,000 packets per second, respectively.

The FDI events were simulated by taking real signals that occur during normal operation where trip is available and copied “as is” to the base trip unavailable Dataset #2a. The same data was used for the FDI curves in all datasets. The FDI attacks lasted 60 seconds before and after the trip button was pressed, leading

to 120 seconds per reactor trip, or 1,560 seconds per signal for a collection of 13 total reactor trips. Ideally, all 67 OT signals could be falsified. However, due to time constraints, only three signals that appear on the main console were chosen to be falsified. This was done with the assumption that an adversary would likely try to falsify data that the operator monitors directly and thus misdirect the operator. The signals chosen for the FDI attacks were:

- Channel 1 counts per second (FDI #1)
- Channel 1 counts per second and change rate (FDI #2)
- Channel 1 counts per second and change rate, and channel 2 counts per second (FDI #3)

Dataset #6 is identical to Dataset #2a. However, it was used with a different label for the purposes of AI/ML training. Dataset #2a represents a base cyber event that renders trip unavailable while Dataset #6 represents a malfunction that renders trip unavailable. Each of the 13 abnormal datasets has 104,520 OT and 9,900 IT datapoints.

The left of Figure 8 provides an example of the trip unavailability represented by Datasets #2a and 6. Figures 8 and 9 provide examples of the different FDI attacks along with the faked and real signals present in each one. The right of Figure 8 shows FDI #1, the left of Figure 9 shows FDI #2, and the right of Figure 9 shows FDI #3. Figure 10 provides an example of both the high intensity DoS (shown in blue) and the low intensity DoS (shown in red).

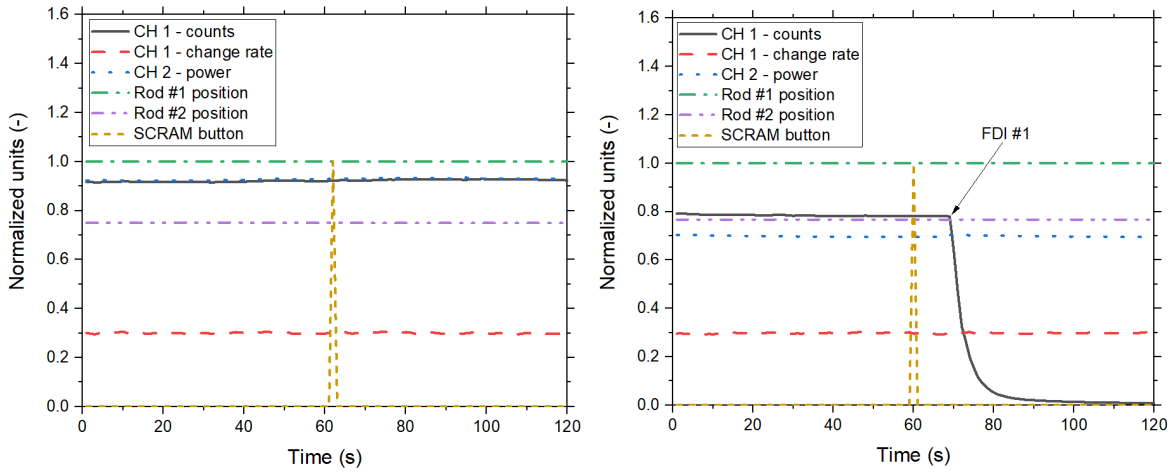

Figure 8. Snapshot of signals during abnormal states trip unavailable (left) and FDI #1 (right).

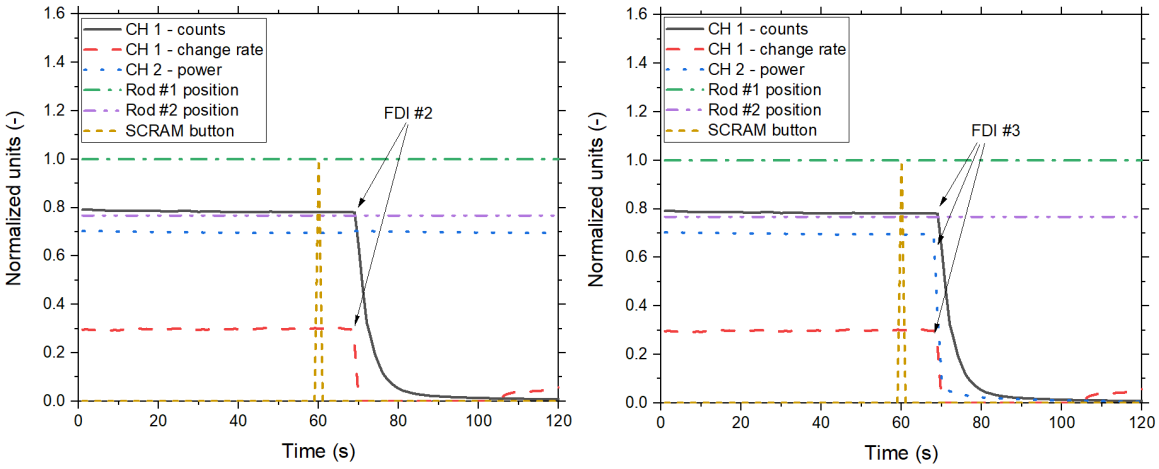

Figure 9. Snapshot of signals during abnormal states FDI #2 (left) and FDI #3 (right).

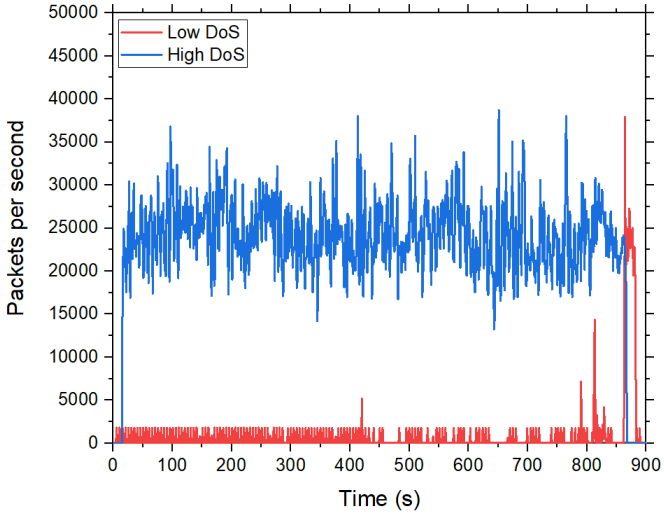

Figure 10. Network traffic during high intensity and low intensity DoS attacks.

4.5 Data Artifacts

It was found that the datasets included noise, outliers, null values, and other artifacts influenced by factors such as instrument response, connectivity delays, or operation variability. These artifacts are attributed to the nature of the collected data, i.e., real-time operational and information technology data. The artifacts were present in all datasets but with varying frequency depending on reactor mode (e.g., whether reactor was shut down or operating). Such artifacts may affect the performance of AI/ML algorithms and, depending on the artifact type and frequency, the data may require cleaning, preprocessing, and transformation before it is used to build AI/ML models. Data artifacts were handled by using similar portions of data without artifacts for dataset reconstruction. Data artifacts included:

Data fluctuations: Two types of data fluctuations were identified: (i) fluctuations due to electronic noise and (ii) fluctuations due to natural variability in the underlying process (e.g., neutron flux). In most cases, these fluctuations follow Gaussian distributions, characterized by a mean value and a standard deviation. Figure 11 below shows examples of signals with different types of data fluctuations. Signals #1 and #2, shown in the top two graphs of Figure 11, are control rod drive position indicators which have electronic

noise but with different characteristics. The bottom left of Figure 11 shows fluctuations in Signal #3 (neutron flux) which is due to natural variability in the data. The bottom right of Figure 11 shows Signal #4, which is from a temperature sensor, where both electronic noise and fluctuations due to natural convention can be observed.

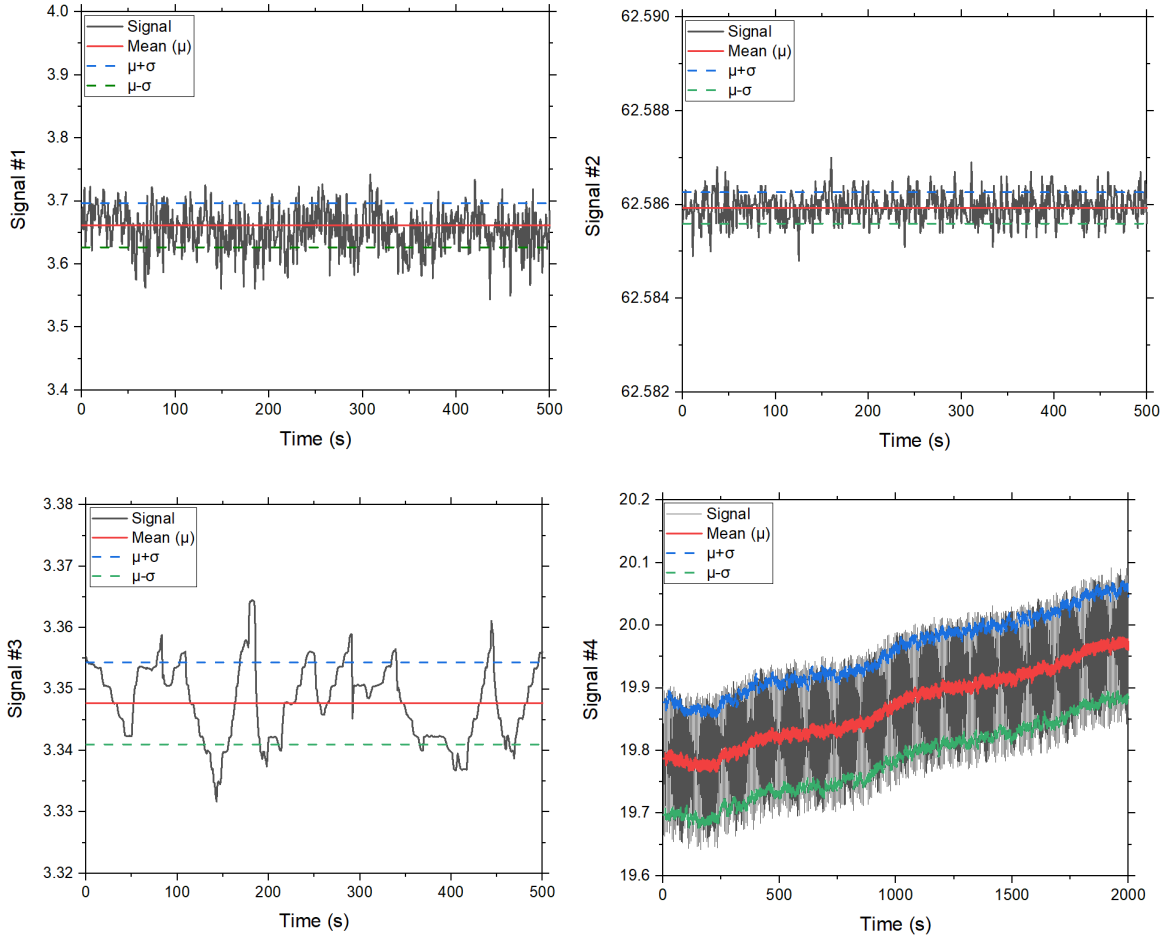

Figure 11. Example of signals with different types of fluctuations.

Outliers: An outlier is a datapoint that significantly deviates from the rest of the signal data. Outliers can create issues with data normalization and potentially skew model performance if not properly identified and handled. The left of Figure 12 shows an example of a signal with outliers of various amplitudes. The outliers identified are attributed to physical processes affecting sensor behavior (e.g., a bubble hitting the thermocouple or electromagnetic interference).

In total, we found 109,660 outliers in the OT data, representing 0.0482% of our total collected data. This dataset had around 227,000,000 total datapoints (including data collected while the plant was in shutdown), of which around 13,000,000 datapoints were operational data. The ratio of operating to total data was 5.6%. Of these outliers, the majority (98.02%) occurred during operation (107,555 outliers), and 1.98% occurred during shutdown (2,173 outliers), which seems to indicate a high prevalence of outliers during operational modes.

Null values: A null value represents a datapoint in which no value was recorded from a particular sensor at a specific time. Random isolated null values could be part of expected system behavior and do not have

significant operational impact. However, clusters of continuous null values and a strong correlation with outliers or noise could indicate sensor malfunction or constraints in transmission during data extraction.

We found 1,811,427 total null values out of 227,637,525 total datapoints. This translates to 0.78% of the initial data collected. Null values were only observed during shutdown mode. There was not a strong correlation between the null value occurrence and other artifacts such as outliers or noise. Null values tended to appear in clusters which could indicate issues with network transmission protocols (OT data relies on Modbus and User Datagram Protocol (UDP) protocols) during data extraction, as there were no obvious sensor malfunctions.

Instrument artifacts: In addition to data fluctuations, outliers, and null values, several other instrument related artifacts were observed. These artifacts can be described as a systematic error or distortion in the measurement process caused by the limitations or imperfections of the instrument itself. Unlike random noise, which is unpredictable and varies from one measurement to another, or outliers, which are extreme values that deviate significantly from the rest of the data, these artifacts are consistent and reproducible under similar conditions. The right of Figure 12 shows an observed signal spike due to instrument range change.

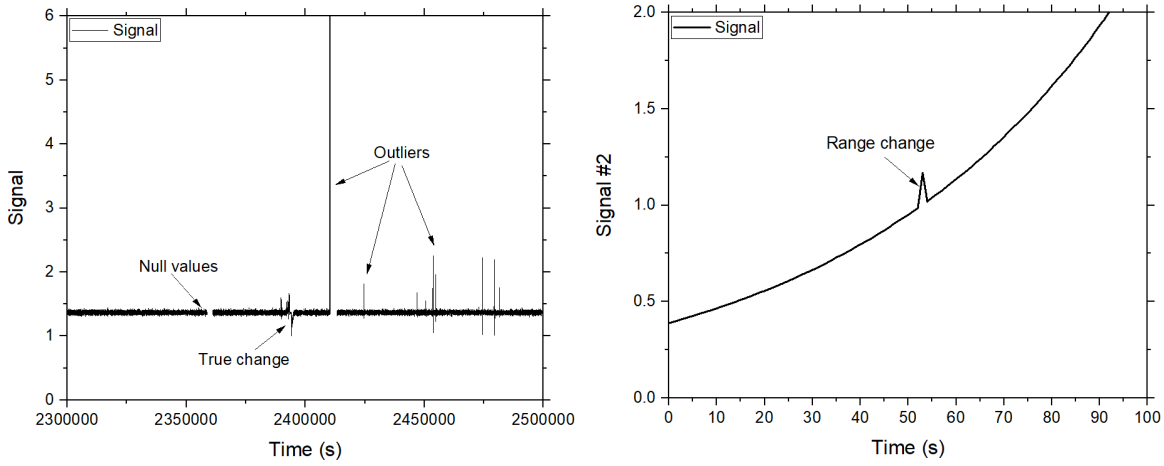

Figure 12. Example of a signals with null values, outliers, and an instrument related artifact.

Operator artifacts: Systematic variations in signals can be generated from variabilities in operation due to human behavior and decision-making processes. Each operator's technique impacts the system differently, which leads to variations in how the operational objective is achieved.

5. AI/ML TECHNOLOGIES AND IMPLEMENTATION

This section presents the implementation of AI/ML models and their performance characterizing cyber events. It includes the design and development of a dedicated classifier architecture, AI/ML model training, validation, and AI/ML model selection and testing. Data collected as part of the use case included a combination of cyber events that result in modification of objects (e.g., false data injection) and modification of relationships between objects (e.g., denial of service). The use case comprised eight events resulting in fourteen different system states (one normal and thirteen abnormal) with 67 OT signals and 11 IT signals. Results are presented using both separate and combined OT and IT data. The performance of the classifier is evaluated and discussed.

5.1 Basis for Identification and Selection of Potential AI/ML Technologies

Project-specific potential AI/ML technologies were identified and selected based on the following criteria:

- AI/ML technologies should allow for characterization of nuclear states (type of problem).
- AI/ML technologies should be able to handle the type, amount, and complexity of the data generated during use case implementation (size, amount, and complexity of data).
- AI/ML technologies should satisfy project-specific performance metrics to the extent possible (performance metrics).
- Implementation of AI/ML technologies should be feasible within the project time and resource constraints (time and resource constraints).
- AI/ML technologies should be currently available and open-source (time and resource constraints).
- AI/ML technologies should be explainable to the extent possible (explainability).
- AI/ML technologies with documented performance in similar domains should be preferred to the extent possible (expertise and experience).

More details on these criteria are available in Section 2.1 of the Task 3 report (Ref. 3).

5.2 Implementation Methodology

Composite classifier

A three-level classification architecture, referred to as the composite classifier and shown in Figure 13, was selected to provide insights on potential AI/ML capabilities to characterize cyber events. This architecture was designed to identify abnormal events, differentiate cyber events from other abnormal events, and differentiate among cyber events. It is noted that this approach is not unique among potentially useful classifiers and that other architectures could be explored and implemented to achieve similar state characterizations.

The composite classifier architecture includes three levels that each perform a binary classification (i.e., output is zero or one, with one representing a positive classification). Each level acts independently and has an AI/ML model that is trained, validated, and tested on a potentially different dataset to classify different states. When the output of all three levels is combined, the composite classifier output provides more information than a standalone binary classifier including the system state (normal or abnormal), the event (cyber or other) and the type of event (FDI, DoS, or combination).

Such an architecture receives and processes OT and IT data separately as a more efficient way of extracting information from the data. Tests with combined OT and IT data were also performed for comparison.

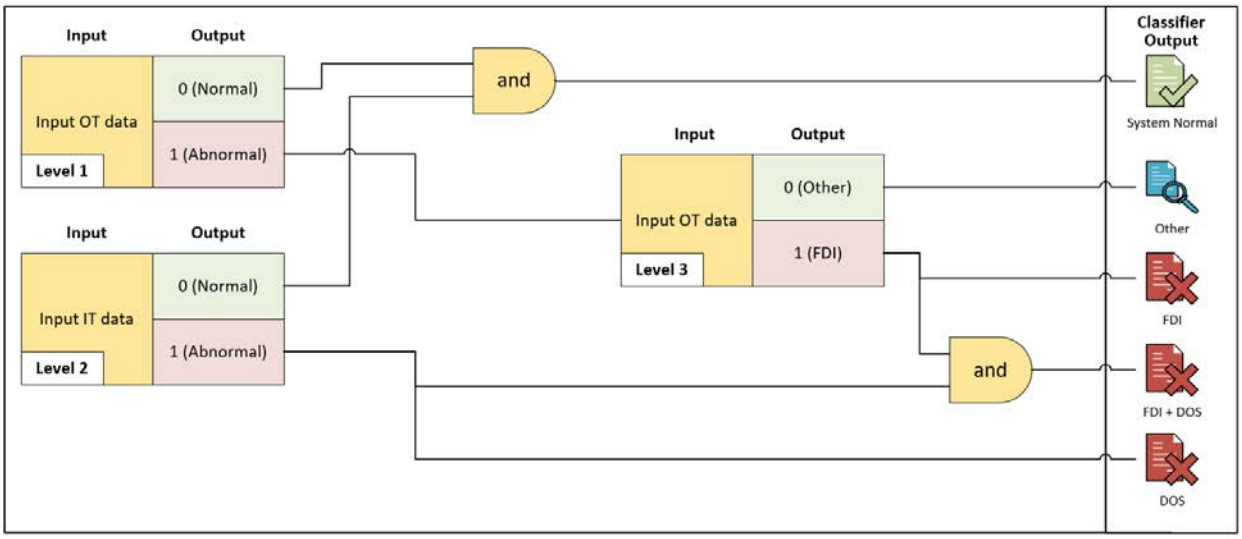

Figure 13. Implemented composite classifier.

In this architecture, Level 1 receives as input OT data and performs binary classification to identify whether system state is normal (output zero) or abnormal (output one). Similarly, Level 2 receives as input the IT data and performs binary classification to confirm the presence of abnormal behavior in the IT data that would indicate a DoS attack. If the output from both Level 1 and Level 2 is normal, then the composite classifier's output is normal, and Level 3 is bypassed. However, if Level 1 identifies an abnormality in the OT data, Level 3 is subsequently used to scan the OT data again to further characterize the identified abnormality as FDI or Other. If FDI is detected at this stage, it signifies a confirmed cyber event. On the other hand, if Level 3 determines the output to be Other that would suggest that the abnormality observed in Level 1 was due to a different kind of event potentially unrelated to cybersecurity. If, in addition, Level 2 is abnormal then this would signify the presence of a DoS attack. A truth table for the composite classifier covering all possible classes is shown in Table I.

With this architecture, the classifier can provide complete characterization of an event including identification of system state (normal or abnormal) and type of event (cyber FDI, cyber DoS, cyber FDI and DoS, cyber DoS and Other, or Other). For example, an output [1 0 1] would be interpreted as "the system state is abnormal, resulting from cyber event which includes FDI." An output [1 1 0] would mean that "the system state is abnormal, resulting from cyber event which includes DoS." An output [0 0 0] would mean that "the system state is normal."

Table I. Composite classifier output truth table.

Level 1	Level 2	Level 3	Output	System Description	
				State	Event Type
0	0	0	[0 0 0]	Normal state–no cyber events	
1	0	0	[1 0 0]	Abnormal	Other
1	0	1	[1 0 1]	Abnormal	Cyber - FDI
0	1	0	[0 1 0]	Abnormal	Cyber - DoS
1	1	0	[1 1 0]	Abnormal	Cyber - Other + DoS
1	1	1	[1 1 1]	Abnormal	Cyber - FDI + DoS
0	0	1	[0 0 1]	Not applicable	
0	1	1	[0 1 1]	Not applicable	

Model training and testing

The methodology used for creating, training, validating, and testing various AI/ML models is shown in Figure 14 and described in more detail below.

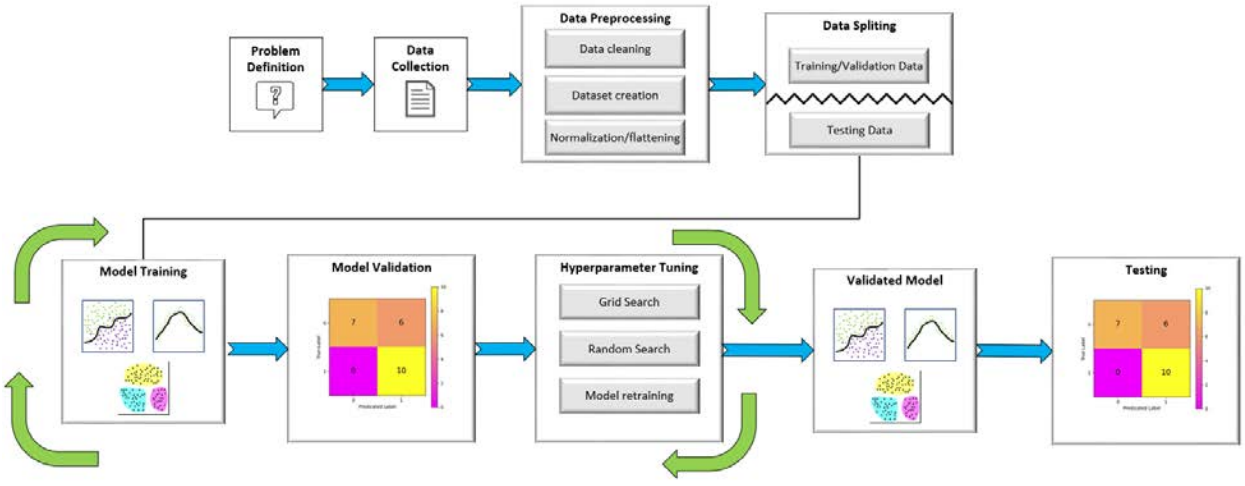

Figure 14. AI/ML implementation methodology.

1. Problem Definition: The selected use case involves multi-step binary classification of time series multi-variate signals for various normal and abnormal events. The signals, either OT or IT, are processed and the output is a binary class (e.g., normal vs. abnormal). The use case is described in more detail in the Task 4 report (Ref. 4).

2. Data Collection and Construction: This step included raw data collection and dataset construction. Around 227,000,000 total datapoints across 67 OT and 11 IT signals were collected between August 2022 and June 2023. This data included the reactor in operation and shutdown. The data was analyzed to identify potential artifacts that could influence AI/ML performance. The analysis of the various data artifacts is presented in the Task 5 report (Ref. 5). Datapoints with reactor shutdown or significant data artifacts were removed and 14 datasets were constructed with a total of (i) 13,400,000 OT and 638,000 IT normal datapoints and (ii) 418,080 OT and 19,800 IT abnormal datapoints. The process for collecting data and then constructing the datasets for the use case is described in more detail in Task 4 report (Ref. 4). The process for collecting data is described in more detail in Task 4 report (Ref. 4). The collected datasets are shown in Task 5 report (Ref. 5).

3. Data Preprocessing: This step included (i) data cleaning in order to identify and handle data artifacts such as missing values, outliers, and other irregularities, (ii) dataset creation suitable for training AI/ML algorithms, (iii) data normalization, and (iv) data flattening.

Data was analyzed to identify potential artifacts that could have an effect in AI/ML performance. Data artifacts are described in more detail with examples in Task 5 report (Ref. 5).

Dataset creation was done for two cases: (i) creating OT and IT datasets that suitable for training, validation, and testing of AI/ML models in Levels 1, 2, and 3 and (ii) combining the OT and IT data and creating datasets suitable to for training, validation, and testing of AI/ML models in Level 1 (note: Level 1 in the combined OT and IT case was modified to accept both OT and IT data). The datasets for both cases are shown in Task 5 report. (Ref. 5)

Data normalization was performed to ensure that all the signals are on the same scale. Without scaling, signals that output very large numbers (such as Channel 1 counts, which is measured in neutron counts per second and often has a magnitude in the millions or billions) can overpower other signals that may be just as important but involve smaller numbers. Normalization occurs separately over each signal in the dataset, putting every signal into the same range of values. Both min-max and standard normalization methods were examined. It was found that, for this use case, the standardization method used had negligible effect on performance.

Data flattening was used for changing the time series data into single portions that can be read by AI/ML models. This involved choosing a window length and including all of the datapoints in this window length as one input to a model. Note: there are other AI/ML algorithms, e.g., artificial neural networks, that can accept as input a matrix and do not require data flattening.

4. Data Splitting: This step includes splitting the datasets into training, validation, and testing sets. The percentage of the dataset reserved for training, validation and testing determines how much data is available for training, validation, and testing, respectively. For example, a higher test split allows for a more stringent performance evaluation but might lead to less training data, which may then deteriorate performance. The following data splitting ratios (train/validation/testing) were used to identify any effects on performance: (i) 50/30/20, (ii) 60/20/20, and (iii) 70/10/20. A complete list of all datasets with various data splits is in the Task 5 report (Ref. 5).

5. Model Training: In this step the actual process of creating and training the AI/ML models using the training datasets was performed. Initially, the training was performed with varying input parameters to identify the best performing model. The input parameters examined window length, window step, scaling, and balance ratios. Once the best model for each level of the composite classifier was identified that model was used in the composite classifier. Model selection is described in later in this section.

6. Model Validation: In this step, validation of trained AI/ML models was performed using the validation dataset. If performance was not found satisfactory, then model is updated using hyperparameter tuning in Step 7 and trained again. Once performance is satisfactory, the validated model was tested in Step 8.

7. Hyperparameter Tuning: This step included fine-tuning the AI/ML models using the validation set by adjusting hyperparameters (e.g., learning rate or tree depth) using techniques such as grid search or random search. The AI/ML models are then re-trained with the optimized hyperparameters. In this work, hyperparameter tuning was performed, however for the selected algorithms the effect on performance was minimal.

8. Model Testing: In this final step, performance evaluation of a trained AI/ML model on the testing dataset was performed. Performance metrics included accuracy, F1 score, precision, recall, confusion matrices, and ROC curves. Accuracy is better suited for balanced classes as it is biased toward the largest class when classes are imbalanced. F1 score considers class imbalance and thus minimizes any bias. Precision and recall provide information on the level of false positives vs. false negatives. Confusion matrices were used to visually represent false positives and false negatives and provide a detailed picture of algorithm performance.

Model selection

Several input parameters that could potentially affect performance, not related to the AI/ML model hyperparameters, were identified. A list of different, independent (as opposed to model hyperparameters) performance-affecting variables was created. These variables are: (i) window length, which represents the number of timesteps fed as input to the model; (ii) window step, which indicates the number of consecutive points between time windows; (iii) balance ratios (BR) for training, validation, and testing sets, which signify the ratio of normal to abnormal data in each set (note: with increasing balance ratio the abnormal data are held constant and normal data size increases); (iv) scaling, which describes the type of normalization applied to data, with the choices being min-max or standard; (v) implemented algorithm (Decision Tree, Random Forest, Logistic Regression, Linear SVM, Naïve Bayes); and (vi) training/validation/testing data split. Table II summarizes the input parameters and their values.

Table II. Parameters used for identifying the best performing AI/ML model.

Independent Variable	**Description**	**Values**
Window Length	Number of timesteps fed as input to the model	All levels: 1, 5, 10, 20, 30
Window Step	Number of points between consecutive time windows	All levels: 1, 3, 5, 7, 10
Training Balance Ratio	Ratio of normal to abnormal data during training	Level 1: 1, 3, 5, 10, 20, 30 Level 2: 1, 3, 5, 10, 20, 30 Level 3: 0.33
Validation Balance Ratio	Ratio of normal to abnormal data during validation	Level 1: 30 Level 2: 30 Level 3: 0.33
Testing Balance Ratio	Ratio of normal to abnormal data during testing	Level 1: 30 Level 2: 30 Level 3: 0.33
Scaling	Type of normalization applied to data	All levels: Min-max, standard
Algorithm	Type of AI/ML algorithm used	All levels: Decision Tree, Random Forest, Logistic Regression, Linear SVM, Naïve Bayes
Train/Validation/Test Split	Proportion of dataset saved for testing	All levels: 50/30/20, 60/20/20, 70/10/20

An exhaustive grid search through all these parameters was performed and the performance of each AI/ML model was evaluated. It was found that not all AI/ML models performed equally well. Among all parameters, the algorithm type, balance ratio, and window length had the biggest effect on performance. Window step and data split appeared to have a noticeable but minimal effect.

The effect of train/validation/test split for two different training balance ratios is shown in Figure 15. The different training data splits shown are 50%, 60%, and 70% and the two balance ratios examined are 1 (seen on the left) and 30 (seen on the right). The results shown in Figure 15 were produced in Level one with the following variables held constant: window length = 20, window step = 1, scaling = standard scaling, validation balance ratio = 30, and testing balance ratio = 30. The train/validation/test split does not significantly impact the F1 score of each model, with the exception of SVMs at low training balance ratio. This is an indication that the training data was adequate to allow sufficient training and reduction in training split did not affect the performance. Because the amount of abnormal data was fixed in this experiment, a higher training balance ratio required the addition of more normal training data. A higher training balance ratio was beneficial for SVMs, Logistic Regression, and Naïve Bayes as the performance of models developed with these algorithms seemed to improve with increased training data. There is also significant difference between models. Independent of training balance ratio, Random Forest and Decision Tree models appear to outperform SVM, Logistic Regression and Naïve Bayes.

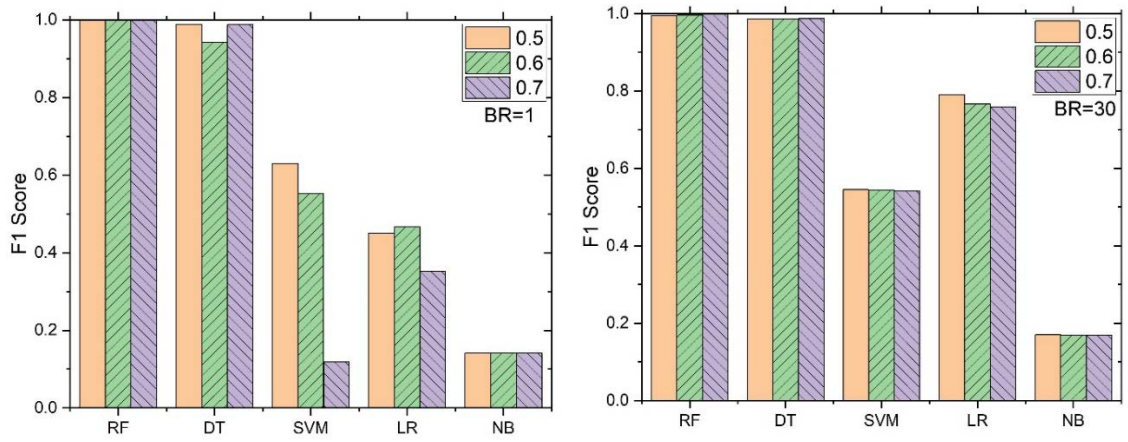

Figure 15. Effect of train/validation/test split for two different training balance ratios.

The effect of training balance ratio is illustrated in more detail in Figure 16a. The results shown in Figure 16 were produced in Level 1 (specifically using the Logistic Regression model) with the following variables held constant: window length = 20, window step = 1, training data split = 60%, scaling = standard scaling, validation balance ratio = 30, and testing balance ratio = 30. It is seen that as the training balance ratio increases and approaches a balance ratio of ten, the overall performance of Logistic Regression improves. However, beyond a training balance ratio of ten, the performance starts to degrade, F1 score decreases, and false negatives tend to increase while false positives drop to zero. Similar, although less pronounced behavior, was observed in the other models. This is expected, as the size of the normal dataset is significantly larger than the abnormal dataset and the gain in error minimization is in favor of false positives rather than false negatives. In other words, a much larger normal dataset, even though it provides more data for training and testing, must be used with caution as it may increase false negatives which can carry a larger risk factor in a nuclear system.

On the other hand, Figure 16b shows a case where the training balance ratio is held constant at 30 but the size of abnormal data varies from 0 to 100% with respect to Figure 16a. The results shown in Figure 16 were produced in Level 1 (specifically using the Random Forest model) with the following variables held constant: window length = 20, window step = 1, training data split = 60%, scaling = standard scaling, validation balance ratio = 30, and testing balance ratio = 30. In this case, it is observed that the performance improves as more data is available for training with increasing abnormal and normal data (but the ratio between normal and abnormal remains constant). This is expected, as more training data provides the models with more information on the class positives and negatives. In other words, there is a minimum amount of data needed to obtain satisfactory performance. If the amount of data is not adequate, then the number of false positives and false negatives tends to increase significantly, despite seemingly high levels of accuracy and F1 score. Additional work is needed to identify the parameters that determine the minimum amount of data needed to reach a certain level of performance.

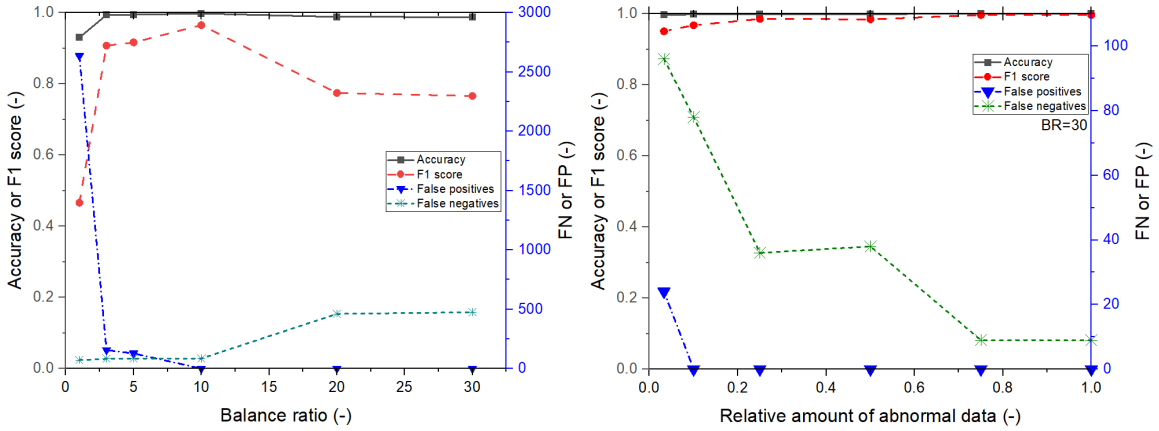

Figure 16. Example of effect of balance in Logistic Regression model performance (Figure 16a, left). Example of effect of changing the amount of abnormal data with constant balance ratio for Random Forest (Figure 16b, right).

The effect of window length on model performance was also explored. The effect of window length for two different training balance ratios (BR=1 and BR=30) is shown in Figure 17. The results shown in Figure 17 were produced in Level 1 with the following variables held constant: window step = 1, training data split 60%, scaling = standard scaling, validation balance ratio =30, and testing balance ratio = 30. It can be seen that the window length does not significantly impact the F1 score of each model, with the exception of SVM. Random Forest and Decision Tree appear quite robust (less variation) against training balance ratio and window length with only very small variations. Similarly to previous results, higher balance ratio tends to improve F1 score.

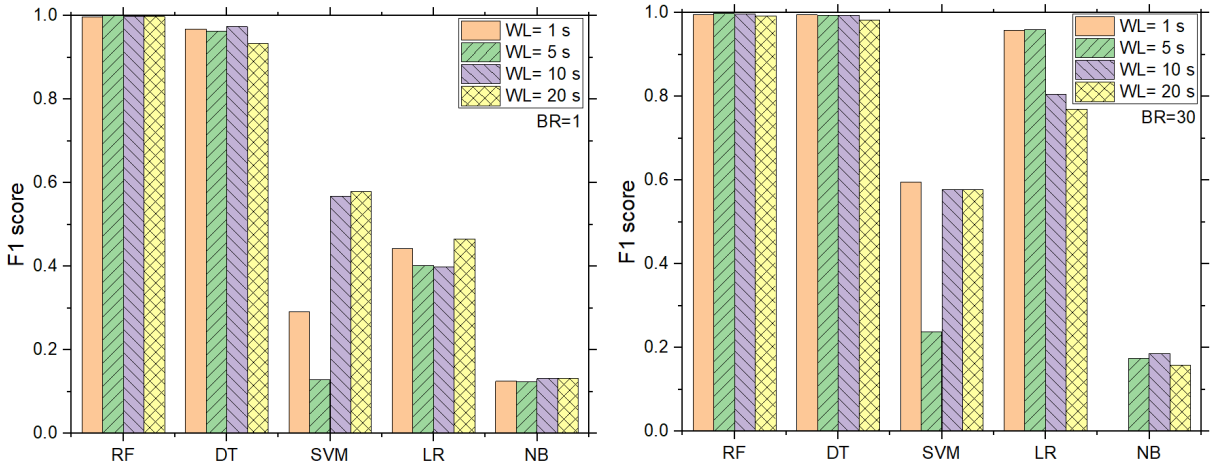

Figure 17. Effect of window length in AI/ML model performance for two different balance ratios.

The computational time required to train each AI/ML model for different balance ratios and window length of 20 seconds is shown in Figure 18 on the left. The right of Figure 18 shows the relative increase in computational time for Random Forest as a function of training balance ratio and window length. The results shown in Figure 18 were produced in Level 1 with the following variables held constant: window step = 1, training data split 60%, scaling = standard scaling, validation balance ratio = 30, and testing balance ratio = 30. Random Forest requires significantly more time for training while Decision Tree is much faster. The relative increase in computational time of Random Forest as a function of the balance

ratio and window length increases at a near-linear rate. In other words, 30 times more training data (due to normal data increasing 30 times while abnormal remains constant) will increase the training time approximately 30 times. Even though a computational time of approximately 800 seconds for Random Forest was manageable for this work, this could be problematic in a large nuclear system where significantly more signals and datapoints are anticipated. An alternative would be to use a Decision Tree instead where performance is similar to a Random Forest.

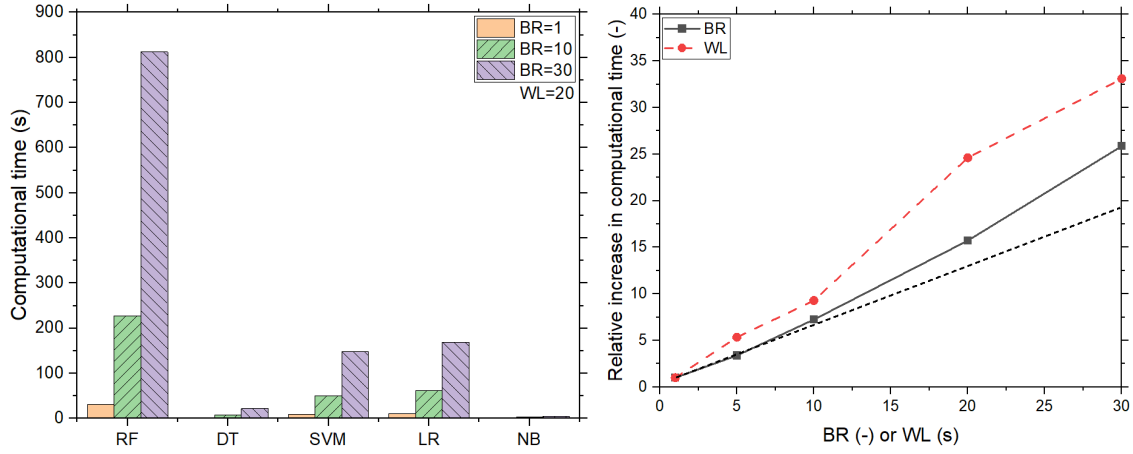

Figure 18. Computational time as a function of AI/ML model and training balance ratio (left). Relative increase in computational time for Random Forest as a function of training balance ratio and window length (right).

In Level 1, Random Forest outperformed all other models consistently across the range of all independent performance-affecting variables. It also appears that Random Forest is very robust against various balance ratios and window lengths with only minimal changes in performance. For Level 2 and 3, all models performed equally well. This can be attributed to large separation in characteristics between classes in Levels 2 and 3, which is not the case for Level 1. Another advantage of Random Forest that proved to be critical in this study is its explainability. Using Random Forest, we were able to identify erroneous models and the signals that introduced this behavior, something that is not feasible with SVM, Logistic Regression or Naïve Bayes. A limitation of Random Forest is that it has a larger computation time for training, which was manageable in the present use case (only a few minutes of training time) but can be problematic in a scenario with thousands of signals and much larger datasets, as scaling to higher dimensions does not appear to be linear. An alternative to Random Forest with reduced computational requirements but similar performance would be Decision Tree, which also provides explainable results.

For these reasons, Random Forest was selected as the best model to use in all levels both using the classifier with separate OT and IT datasets (in Levels 1, 2, and 3) and with combined OT and IT datasets (with a modified Level 1 that can accept OT and IT simultaneously). The set of best performing variables is: (i) window length 20 seconds, (ii) window step 1 second, (iii) standard scaling, (iv) data split 60/20/20, and (v) training balance ratio 20. A detailed comparison between models for this set of parameters is shown in Task 5 report.

5.3. Performance Evaluation

Performance evaluation using separate OT and IT

Using Random Forest for the classifier in all three levels, each Random Forest model was trained again with the optimal set of parameters: (i) window length 20 seconds, (ii) window step 1 second, (iii) standard

scaling, (iv) data split 60/20/20, and (v) training balance ratio 20. Once trained, the Random Forest model in Levels 1 and 2 was tested with a balance ratio of 30, and the Level 3 model with a balance ratio of 0.33. The performance metrics for each level are shown in in Table III and the confusion matrices in Figure 19. It can be seen that the performance in Levels 2 and 3 achieves F1 score of 100%, while the performance at Level 1 is 99.7% F1 score with six false negatives and zero false positives.

Table III. Performance metrics for separate OT and IT.

	Accuracy	F1 score	Precision	Recall
Level 1	0.999845	0.997598	1.0	0.995208
Level 2	1.0	1.0	1.0	1.0
Level 3	1.0	1.0	1.0	1.0

		L1-Predicted	
		Positive	Negative
Actual	Positive	1246	6
	Negative	0	37,560

		L2-Predicted	
		Positive	Negative
Actual	Positive	355	0
	Negative	0	10,668

		L3-Predicted	
		Positive	Negative
Actual	Positive	930	0
	Negative	0	310

Figure 19. Confusion matrix for Level 1 (L1), Level 2 (L2) and Level 3 (L3) for separate OT and IT.

Additional tests were performed with out-of-training data (i.e., data not part of the previous training datasets) to test the robustness of the Random Forest among the different levels of the composite classifier. In these tests, each Random Forest model attempts to identify abnormal events for which it was not specifically trained on. A new training dataset for each classification level was created for this purpose and included false data injections in Channels 3 and 4 but not in Channels 1 and 2 combined with a denial of service attack. The performance metrics for each level are shown in Table IV and the confusion matrices in Figure 20. The Random Forest model was able to correctly identify that the event was abnormal in Level 1 with an F1 score of 86% and 361 false negatives. Level 2 identified the DoS with an F1 score of 100% and zero false negatives or positives. It is noted, initially the Level 2 Random Forest model showed poor performance which, following further analysis into the Random Forest structure, was then attributed to one of the Central Processing Unit (CPU) temperature monitoring signals (one of the collected 11 IT signals). It appears that during data collection of normal data, the CPU temperature was higher than the CPU temperature during data collection of abnormal data. When out-of-training abnormal data was collected, the CPU temperature was again higher and when the data was fed into the model, the model erroneously classified everything as normal simply because the CPU temperature signature matched that of normal data. Once this signal was corrected, Level 2 performed correctly. Finally, Level 3 showed poor performance with an F1 score of 34% and 5,981 false positives.

Table IV. Performance metrics for separate OT and IT and out-of-training data.

	Accuracy	**F1 score**	**Precision**	**Recall**
Level 1	0.988537	0.867231	1.0	0.765584
Level 2	1	1	1	1
Level 3	0.2	0.34	0.2	1

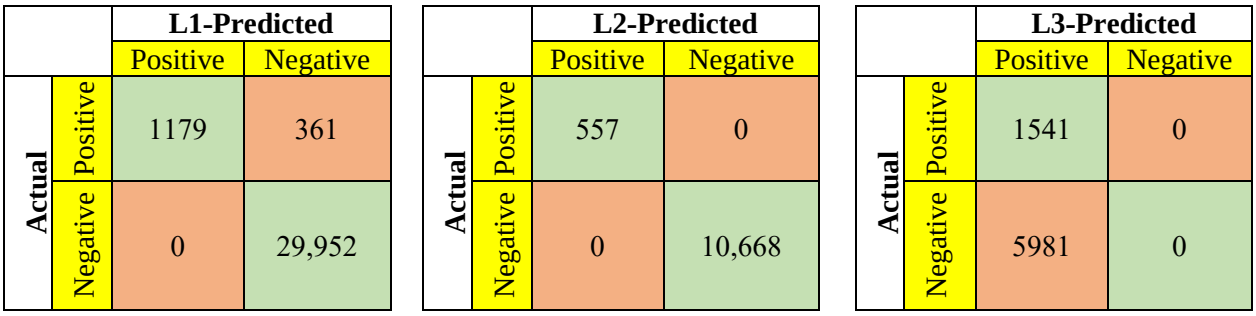

Figure 20. Confusion matrix for Level 1 (L1), Level 2 (L2) and Level 3 (L3) with out-of-training data.

Performance evaluation using combined OT and IT

OT and IT signals were combined to construct a dataset with 78 OT and IT signals. All algorithms were trained and tested using this dataset with two classes, normal and abnormal, with the following parameters: (i) window length 20 seconds, (ii) window step 1 second, (iii) data split 60/20/20, and (iv) training balance ratio 1. Once trained, the AI/ML models were tested with a balance ratio of one. The performance metrics are shown in Table V. Random Forest and Decision Tree have 100% F1 score with zero false negatives and zero false positives. The F1 score of SVM, Logistic Regression and Naïve Bayes ranges from 60% to 80%. The confusion matrix for Random Forest is shown in Figure 20.

Although an approach using combined OT and IT can classify normal vs abnormal events, it might be much harder to differentiate between various events unless using a multi-class architecture. Combining the OT and IT signals is an additional step in the dataset construction process, as typically OT and IT signals are collected using very different monitoring tools/software. Another difficulty in combining OT and IT data are differences in sampling frequency. Finally, for certain events, it may not be even possible to collect OT and IT simultaneously, such as during a denial of service attack when the capability to collect OT data could be hampered by the attack.

Table V. Performance metrics for combined OT and IT.

	Accuracy	**F1 score**	**Precision**	**Recall**
RF	1	1	1	1
DT	1	1	1	1
SVM	0.58952	0.693953488	0.553207267	0.930754835
LR	0.828135	0.807005254	0.920128	0.718652527
NB	0.718653	0.608507	1	0.437305053

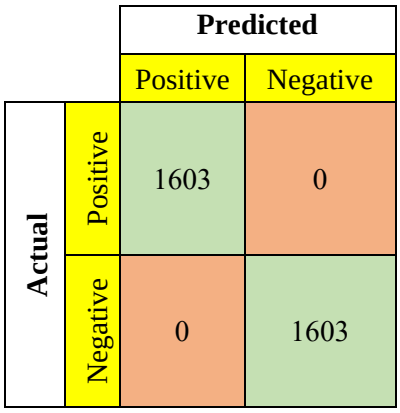

Figure 21. Confusion matrix for Random Forest using combined OT and IT.

Performance Evaluation of Composite Classifier

The performance of the composite classifier using separate OT and IT was evaluated. The composite classifier operates with three levels of binary classification (Level 1, Level 2, and Level 3) that, when combined, result in a multi-classification with six classes (Normal - 0, Other (malfunction) - 1, FDI- 2, DoS - 3, Other + DoS - 4, and FDI + DoS - 5). Table I shows the possible combinations and resulting classes together with a description of each class.

To test the performance, datasets belonging to each class were created and then processed through each of the levels of the composite classifier. The combined output of the composite classifier was recorded, and it was compared with the actual class. The confusion matrix for the composite classifier is shown in Figure 22. The composite classifier correctly identifies the majority of instances for each class, as indicated by the high values along the diagonal. There are no instances where Class 0, Class 1, Class 2, or Class 5 are incorrectly predicted as another class. There is only one instance where Class 4 is misclassified as Class 0, and one instance where Class 3 is misclassified as Class 5. Overall, the composite classifier demonstrated improved performance with minimal misclassifications when compared with the individual binary classifiers (Levels 1, 2, and 3). This shows that a classifier architecture that combines multiple binary classifiers can provide multiple layers of detection of abnormalities and potentially outperform the individual classifiers.

		Predicted					
		Class 0	Class 1	Class 2	Class 3	Class 4	Class 5
Actual	Class 0	10,324	0	0	0	0	0
	Class 1	0	87	0	0	0	0
	Class 2	0	0	3	0	0	0
	Class 3	0	0	0	5	0	1
	Class 4	1	0	0	0	256	0
	Class 5	0	0	0	0	0	346

Figure 22. Confusion matrix of the composite classifier.

6. OBSERVATIONS–INSIGHTS

The overarching goal of this project was to test the following hypothesis and provide evidence to support answering the following questions:

Hypothesis: *AI/ML can be feasibly and usefully applied to characterize system states resulting from cyber events.*

Outcome: Based on the results of the implemented use case, AI/ML can be usefully applied to characterize system states and differentiate between cyber events.

Challenges: It is important to have a good understanding of the problem space early in the process and to dedicate enough time and resources for setting and exploring the problem space in enough detail. In this project, setting and exploring the problem space was found to be both time-consuming and challenging even for a simple use case as the one in this project. The number of system states and events resulted in a relatively large number of potential end state combinations and dedicated datasets which made it challenging to collect data and construct the datasets. The increased number of datasets required substantially more AI/ML training and a more sophisticated classification architecture than just a simple binary one. Another challenge was artifacts, hidden patterns, and correlations in OT or IT data that resulted in time-consuming data cleaning and an increased risk of potentially erroneous models that could go undetected.

Gap: This work covers a single use case with limited number of system states and cyber events. A nuclear system would typically have more real-time signals and exponentially more system states. The range of cyber events could also be more extensive than those studied herein. Additional work is required to provide further evidence that AI/ML can be applied more generally and cover a broader range of system states and events with satisfactory results.

Research Questions

Q1: *Can abnormal events, such as equipment malfunctions and cyberattacks, result in distinguishable system states?*

Outcome: Based on the results of this study, cyber events can result in distinguishable states. Using OT and IT data (either separately or combined) and a well-trained AI/ML system, these events can be differentiated with reasonable performance.

Challenges: Obtaining data and signals for each state under identical initial conditions was found to be rather challenging, especially when normal data exhibits small patterns that make distinguishing between normal and abnormal events difficult. This complexity necessitated the inclusion of more abnormal state data to improve model accuracy. For example, generating data for equipment malfunctions is particularly difficult due to the various ways they can occur, complicating AI/ML model training and testing for all possible scenarios.

Gap: The use case in the study included a limited number of cyber events (FDI and DoS attacks) compared to the possible space of cyber events that could occur on a cyber-physical system. Additional work is required to better understand the degree to which certain cyber events may or may not be distinguishable from certain system states such as those resulting from equipment malfunction.

Q2: *Can system states be represented with IT and OT time series data?*

Outcome: Based on the results of this study, system states can be reasonably represented with OT and IT time series data.

Challenges: IT and OT data were generated from separate systems that are not connected making synchronization and sampling a challenge.

Gap: There could be system states for which OT and IT representation would not be adequate to fully describe the system state. For example, an equipment malfunction due to corrosion might not be captured in real-time by OT data unless this condition is specifically monitored. Additional work will be required to better understand challenges with representing system states, including states or events not monitored in real-time.

Q3: *How should the data be collected and processed for input suitable to an AI/ML system?*

Outcome: This study provided an implementation methodology for collecting and processing datasets suitable for input to AI/ML. The methodology appeared adequate, and performance was within reasonable expectations.

Challenges: Data collection was time-consuming and dependent on monitoring and communication system settings. This had a noticeable influence on the number of null values and data artifacts, which then required a rigorous pre-processing effort to remove data artifacts. Data management required a significant amount of time as well to perform shuffling, data split, and finding optimal way of loading several millions of datapoints to memory. It is noted that the many combinations of independent parameters required every pre-processing step to be redone to ensure correct AI/ML training and testing.

Gap: The implementation methodology was specific to the use case, the amount of the data collected, and the selected AI/ML algorithms. The proposed methodology might not be adequate for a different use case with a larger dimension space, or different AI/ML algorithms. Additional work is required to evaluate the implemented methodology under conditions substantially different that the use case herein.

Q4: *Can an AI/ML system trained with data extracted from normal and abnormal system states characterize those states?*

Outcome: Based on the results of this study, AI/ML can be trained with data extracted from normal and abnormal system states. The trained AI/ML models were able to characterize those states and their performance was evaluated.

Challenges: There were certain parameters that are not known beforehand such as the balance ratio between normal and abnormal data or the time series data window length. Both balance ratio and window length need to be carefully selected to avoid undertraining or having certain events go undetected (e.g., if event duration is shorter/longer than selected window length). In addition, AI/ML training can be dependent on the classification architecture, but the optimal architecture may not be evident from the start. Finding an optimal classification architecture that is simple enough without compromising accuracy required several trial-and-error iterations.

Gap: The data collected included correlated and uncorrelated signals from a variety of sensors as expected from a real-world system. It was observed several times during preliminary tests that small changes in certain signals could go unnoticed and negatively affect the performance of AI/ML models. For example, data collected on different days had small changes in certain signals, such as room temperature, not critical to the system states. However, these changes in signals played a key role in confusing the AI/ML models, leading to erroneous models that would classify the system states based on those signals instead. To avoid

this issue, data should be carefully curated, and experimental conditions must be well controlled to avoid erroneous AI/ML models. Additional work is required to better understand what constitutes reasonable performance and how to establish criteria (e.g., risk assessment) for determining acceptable levels of performance under different conditions, how such erroneous models are created, how frequent they are and how to develop methods for early detection, e.g., use of explainable models.

Q5: *Can an AI/ML system differentiate between cyber events and other non-cyber events?*

Outcome: Based on the results of this study, AI/ML was able to differentiate between cyber events and other non-cyber events with reasonable performance.

Challenges: Abnormal events, cyber or other, had small datasets. In addition, generating data was difficult and time-consuming. Differentiating between two abnormal classes with very small dataset in each class was challenging.

Gap: There could be cyber events that are very similar to other non-cyber events and for which differentiation might be more challenging. Additional work is required to better understand limitations of AI/ML to differentiate cyber events.

Q6: *Can AI/ML models be combined to perform multi-state classification?*

Outcome: Based on the results of this study, AI/ML models can be combined in a composite classifier having multiple binary classifiers using Boolean logic to perform multi-state classification.

Challenges: Defining and quantifying overall performance was not straightforward. Several performance metrics were used, and a particular metric could carry different weight depending on the application. For example, using a generic metric such as accuracy or F1 score was useful for quickly evaluating the performance of a model but at the end minimizing false negatives at the expense of accuracy could be more important for certain events.

Gap: Constructing an efficient composite classifier becomes increasingly difficult as the number of classified states increases. For example, classifying falsified signals requires an exponentially increasing number of combinations that must be identified. This can be challenging when the number of signals is large. Additional work is needed to better understand the efficiency and performance tradeoffs between one stand alone or composite classifier architectures for multi-state classification and to identify approaches that can generalize well based on a small number of signals.

Q7: *What are the limitations of such an AI/ML system, its performance, and what variables can affect its performance?*

Outcome: Based on the results of this study, limitations of AI/ML included the ability to handle real-time dynamic high dimensionality data, the complexity associated with dataset management due to large number of datasets, the exponentially increasing number of potential events that could be performed and collected, and the time required to perform a grid search and identify a best model due to large number of independent variables. These limitations were somewhat constrained in this use case due to the small number of signals and events, but it is expected that these limitations can grow exponentially in a larger nuclear system with thousands of signals, states, and events. Another major limitation of AI/ML algorithms was a tendency to produce erroneous models because of data artifacts. This was addressed herein by using algorithms that provide a degree of explainability (e.g., Decision Tree and Random Forest). Finally, variables that affected AI/ML performance included balance ratios and window lengths.

Challenges: Because of the novel problem space of this project, developing an experimental methodology and use case that provided both meaningful results but remained tractable required significant effort. Both experimentation, iteration, and extensive nuclear domain knowledge were required. A significant portion of the project effort was also needed to develop and understand the experimental framework, as well as to make changes to the framework as the problem space was better understood.

Gap: Although performance of AI/ML models in this work was deemed acceptable, there is no generally agreed upon guidance or technical basis to support the fact that this performance would be acceptable at a nuclear system. Additional work is needed to develop a technical basis to support identification of the level of acceptable performance for each AI/ML model. Also, additional work is needed to establish a technical basis for the selection of representative and realistic balance ratios, and a methodology extensible to larger and more complex problem spaces.

Insights

This study offered insights in the development and application of AI/ML for characterization of cyber events. These are discussed below:

Performance: It was found that AI/ML system performance is highly dependent on several factors including data quality, experimental conditions, balance ratio between different classes, and selected algorithms. Maximum achievable performance also depends strongly on the data available which may not be easily obtained in a nuclear system. For example, an acceptable performance of 99.999% would require a lot more data compared to 99.9%. This would be the case for any general performance metric, such as accuracy or F1 score. Additional work is needed to better understand what affects the accuracy and reliability of an AI/ML system and how can such effects be determined or measured.

Data artifacts: It was found that data artifacts are present in the datasets. Depending on the system state, data artifacts can appear in significant quantities. Data artifacts, such as outliers or missing values, may provide information on potential issues in system operation, e.g., in this work clusters of missing values indicated network irregularities. Some artifacts must be handled during data preprocessing to minimize issues with AI/ML training and performance.

Unique aspects: Implementing a use case in a nuclear system may present unique challenges including temporal inconsistencies between different types of data, e.g., OT vs IT, difficulty in obtaining data under well-defined and controlled experimental conditions, a large number of system states and signals which results in a high dimensionality space, data artifacts, and a bias toward false positives to minimize unintended consequences due to false negatives.

Data management: Even for a relatively simple use case as the one in this study, it was found that several steps are needed to transform the collected datasets to datasets suitable for input to AI/ML algorithms. The number of states and events and their combinations can increase exponentially rendering dataset management very time-consuming and challenging. This issue coupled with the large number of signals creates a dimensionality explosion that can create issues with certain algorithms. For example, there were cases where AI/ML algorithms would not converge due a mismatch between the selected algorithm and input dataset. On the other hand, data management is critical to ensure that the right data are fed to the right algorithm.

Robustness: Several parameters may not be known beforehand, e.g., true balance ratios. Evaluating AI/ML algorithms under a broad parameter range and identifying an AI/ML model that is robust against unforeseen

changes in these parameters would be beneficial. Random Forest and Decision Tree models were shown to be more robust than the rest of the models.

Explainability: Hidden patterns in data and/or data collection under different or not well controlled initial conditions may create erroneous models (i.e., models that appear to perform well but do not capture the true behavior of the event) that can go undetected. In this study, the presence of such erroneous models was identified during analysis performed in the structure of Random Forest and Decision Tree models while it was missed for the other AI/ML algorithms. The use of simple and explainable algorithms, either stand alone or as a benchmark, is critical in ensuring that the correct event behavior captured during training and for identifying hidden patterns in the data.

Alternate approaches: In this study, two approaches were implemented, one with separate OT and IT and one with combined OT and IT. Alternate approaches not used in this study include the use of unsupervised AI/ML algorithms to identify system states and deep learning AI. Approaches based on pre-defined thresholds to detect divergence from normal behavior without the use of AI/ML would probably not be suited given the dynamic nature of the data. However, additional work will be needed to better understand the limitations of these techniques.

Cybersecurity risks: Introducing a new tool, in this case an AI/ML system, can potentially introduce new cybersecurity vulnerabilities and increase the attack surface. There could be several potential vulnerabilities in any of the several stages needed to develop and deploy an AI/ML tool for cyber event characterization.

During the development phase, vulnerabilities can be related to the data extraction and collection process (during collection or while the data is in storage), the algorithm, the final model, any libraries used, training and testing procedure, classification architecture, etc. For example, intentional data or model manipulation (e.g., introducing data artifacts or data and model poisoning) can create erroneous models that appear to perform well but can generate misleading results under certain conditions. Moreover, the reliance on open-source libraries and algorithms can introduce additional vulnerabilities. Open-source code, while beneficial for its cost-effectiveness, could allow an avenue for intentionally hidden flaws or malicious code that attackers can exploit. For example, a seemingly benign update to an open-source library used by the AI/ML system could contain vulnerabilities that an adversary could exploit to manipulate the AI/ML models. Even with vetted code, supply chain vulnerabilities could introduce bugs or opportunities for undetected modifications.

During the deployment phase, intentional modification of OT or IT data could cause the AI/ML system to misclassify the state, which could lead to inappropriate responses and potentially catastrophic outcomes. For example, if an attacker modifies sensor data inputs, the AI/ML could misinterpret the operational state, leading to inappropriate responses. Insecure communications, weak authentication mechanisms, insufficient privilege escalation control, and man-in-the-middle attacks between the instrumentation and control system and the AI/ML tool could create pathways for data modifications. In addition, it is expected that frequent maintenance and retraining will be required to account for changes in operational conditions over time. This could present another opportunity for introducing modifications to data and models. Finally, model inference attacks where an attacker might deduce information from the outputs of the AI/ML model and use reverse engineering the model to discover its vulnerabilities.

Addressing these vulnerabilities requires a comprehensive approach to security, including rigorous testing, robust security policies, employee training, and continuous monitoring and updating of systems.

7. FUTURE WORK

Potential future work includes:

- Train AI/ML algorithms to learn normal plant states (outlier detection instead of attack classification).
- Expand the number of events and system states to include explicit equipment malfunctions.
- Expand the types and scopes of cyber events performed or simulated.
- Perform additional tests with combined IT and OT data.
- Identify minimum number of IT and OT signals below which performance is significantly affected.
- Evaluate additional AI/ML algorithms.
- Deeper investigation into how balance ratio impacts performance.
- Develop methods for identifying erroneous models.
- Evaluate the potential benefits of dimensionality reduction methods.
- Evaluate advantages and limitations of the composite classifier and its capability to generalize to states not represented in the training data.
- Explore other classification architectures and how they compare to the composite classifier.
- Explore the performance of classification architectures that combine multiple binary classifiers vs. a single multi-class classifier.
- Develop performance-based assessment or evaluation methods for cybersecurity AI/ML models.
- Evaluate the effects of data artifacts and associated correction methods on model performance.
- Explore potential use of data artifacts as signatures (using data artifacts to distinguish between real plant data and false data, and to provide additional state data useful for classification).

REFERENCES

1. U.S. Nuclear Regulatory Commission (NRC). Technical Letter Report TLR-RES/DE-2024-03a, “Research Plan Development.” 2024. (ML23040A169).
2. NRC. Technical Letter Report TLR-RES/DE-2024-03b, “Identification of a Representative Use Case.” 2024. (ML23062A349).
3. NRC. Technical Letter Report TLR-RES/DE-2024-03c, “Identification of AI/ML Technologies.” 2024. (ML23102A182).
4. NRC. Technical Letter Report TLR-RES/DE-2024-03d, “Use Case Implementation.” 2024. (ML24052A002).
5. NRC. Technical Letter Report TLR-RES/DE-2024-03e, “Performance Evaluation and Gap Analysis.” 2024. (ML24193A007).
6. National Institute of Standards and Technology (NIST) Computer Security Resource Center. “Glossary.” 2024. https://csrc.nist.gov/glossary.
7. NRC. NUREG-2261, “Artificial Intelligence Strategic Plan Fiscal Years 2023-2027.” 2023.
8. NIST. SP 800-16, “Information Technology Security Training Requirements: a Role- and Performance-Based Model.” 1998.
9. NIST. SP 800-172, “Enhanced Security Requirements for Protecting Controlled Unclassified Information: A Supplement to NIST Special Publication 800-171.” 2021.
10. NIST. SP 800-37 Revision 2, “Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy.” 2018.
11. NIST. SP 800-12 Revision 1, “An Introduction to Information Security.” 2017.

APPENDIX A - GLOSSARY

The following definitions are used throughout the report. These definitions are specific to the project scope and have been selected to facilitate a more precise formalism and clearer representation and analysis of the terms used throughout this project's problem space. Effort has been made to align these definitions with the more generic NIST cybersecurity definitions (Ref. 6) and U.S. NRC AI Strategic Plan (Ref. 7) definitions. Where appropriate, the NIST related definition is also provided.

A.1 General definitions

Cyber Event: If an event is unauthorized and executed using digital software, hardware, and/or information with malicious intent then it is classified as cyber event. A cyber event is unauthorized action taken by an adversary aiming to intentionally modify system state by targeting objects or the relationship between objects using digital software, hardware, and/or information with malicious intent. Potential consequences from an event may include loss of confidentiality, integrity or availability and may result in compromised plant safety, disruption of normal operation, and/or leaked information.

NIST definition of cyber event: A cybersecurity change that may have an impact on organizational operations (including mission, capabilities, or reputation).

NIST definition of cyber incident: Actions taken through the use of an information system or network that result in an actual or potentially adverse effect on an information system, network, and/or the information residing therein.

Event: An event is an action that may result in state change either by changing an object or the relationship between objects.

NIST definition of event: Any observable occurrence in a system.

Mode: Mode describes the condition (operational or not) of a system. A nuclear system can have multiple modes, (e.g., startup, power operation, standby, refueling, or hot shutdown). There are four modes in PUR-1: startup, operating, shutdown, and secured.

Object: An object can be any system component, equipment, instrument, data or data connection. Objects can be interconnected and communicate with other objects. Any two or more objects can be interconnected and a change in one object may affect another object depending on the relationship between objects. Relationships between objects can be one-way or two-way. For example, a plant controller (object) may communicate over a two-way Ethernet connection (relationship) to send or receive information from or to an actuator (object).

State: A state describes the condition of the mode. A state can change as a result of an event that modifies an object or the relationship between objects. For simplicity, states are categorized as normal or abnormal although other categories can be used if necessary. For example, a system state can change from normal to abnormal as a result of a cyber event or equipment degradation. Note: a mode may be inclusive of multiple possible nuclear system states.

System: A system is a set of objects including the relationships among the objects. For each instant in time, the information needed to describe the condition of the system, (i.e., the condition of its objects and relationships) represents the system's state. A system may change from one discrete state to another in response to an event, (i.e., it is a discrete event-driven state space). Some system states are normal (reflect

the allowable system conditions), and some systems states are abnormal and may reflect unacceptable or unauthorized (in terms of cybersecurity aspects) conditions of objects and relationships. The formalism adopted by this project represents a nuclear powerplant as a system.

A.2 Definitions specific to AI/ML

AI/ML Algorithm: An AI/ML algorithm is a specific mathematical formula or procedure that is used to learn patterns or make predictions from data.

Artificial Intelligence (AI): The term AI refers to a machine-based system that can go beyond defined results and scenarios and has the ability to emulate human-like perception, cognition, planning, learning, communication, or physical action. For a given set of human-defined objectives, AI can make predictions, recommendations, or decisions influencing real or virtual environments (adapted from NUREG-2261).

AI/ML Model: AI/ML algorithm and AI/ML model are related concepts, but are not interchangeable. An AI/ML model is the end result of running an AI/ML algorithm on data. It is the mathematical representation of the learned patterns in the data, which can be used to make predictions or classify new data points. The model can later be used to make predictions on new data without being explicitly programmed for the task. A single algorithm can create many models based on the input data and the specific parameters used.

Machine Learning (ML): ML is a subset of AI that is characterized by providing machine-based systems with the ability to automatically learn and improve on the basis of data or experience, without being explicitly programmed (adapted from NUREG-2261).

Overfitting: A situation at which a ML model learns the underlying patterns in only the training data but fails to generalize to new, unseen data. As a result, an overfit model performs well on the training data but has poor performance on new data.

Testing Dataset: A dataset, different from the training or validation dataset, used to evaluate the model's performance and assess the generalization of the model to data not previously seen by the model.

Training Dataset: A dataset used to train the ML model. It consists of examples of both normal and abnormal classes and is used to teach the model the patterns associated with each behavior.

Validation Dataset: A dataset, different from the training dataset, used to optimize the model's hyperparameters and avoid overfitting. The performance on the validation dataset helps in making adjustments to improve the model's generalization.

A.3 Definitions specific to data

Datapoint: Refers to a single observation (signal and timestep) in the dataset.

Dataset: Refers to a collection of data in tabular format of rows and columns, where each row represents a timestep and each column a different signal value.

Information Technology (IT): Computing and/or communications hardware and/or software components and related resources that can collect, store, process, maintain, share, transmit, or dispose of data. IT components include computers and associated peripheral devices, computer operating systems, utility/support software, and communications hardware and software (NIST SP 800-16).

IT Data: Data extracted from IT components, computing and/or communications hardware and/or software components and related resources that can collect, store, process, maintain, share, transmit, or dispose of data.

Operational Technology (OT): The hardware, software, and firmware components of a system used to detect or cause changes in physical processes through the direct control and monitoring of physical devices (NIST SP 800-172, NIST SP 800-37).

OT Data: Data extracted from OT components, hardware, software, and firmware components of a system used to detect or cause changes in physical processes through the direct control and monitoring of physical devices.

Signal: Refers to a time-varying quantity that represents a measurable phenomenon. In the context of this project, a signal represents each of the 67 measured or derived OT or 11 IT parameters from the sensors, network, and setpoints. Examples of the different signals used are neutron counts, control rod position, channel counts, water temperature, data packets, etc.

Timestep: Represents a specific duration in time at which signals are sampled. In the context of this project, the timestep is one second, which means that every second a new value is recorded for each signal.

Time Window or Window: Refers to a series of consecutive datapoints, captured over a continuous time period. Each datapoint within the sequence corresponds to a specific second in time, and the ordering of these datapoints is crucial for understanding the temporal patterns and dynamics of the data. For example, a continuous recording of the values of all 67 signals over a specific time, such as 10 seconds, is referred to as a 10 second window.

A.4 Definitions specific to experimental framework

Characterization: The capability to describe the features or properties of a state or event relative to established measures. For example, a state may be characterized as "an abnormal state resulting from a high-intensity DoS cyberattack." A complete characterization includes state or event identification and differentiation from other states or events.

Classification: Refers to the process of assigning a data or signals to a specific class label.

Classifier: A classifier is a type of AI/ML algorithm typically used to assign a class label to a data input.

Composite Classifier: The implemented classification architecture which consists of a Boolean combination of three binary classifiers (Levels 1, 2, and 3) each with a different classification objective.

Denial of Service (DoS): The unauthorized increase of network traffic, e.g., number of data packets or packet size, aiming to disrupt transmission of critical OT data to the operator.

NIST definition: The prevention of authorized access to resources or the delaying of time-critical operations (NIST SP 800-12).

Differentiation: The capability to distinguish (differentiate) among different states or events. For example, "state 1 is different than state 2." Differentiation is a subset of characterization.

Domain Knowledge: Domain knowledge refers to expertise and understanding of the specific field or industry from which the data is sourced. It involves a deep understanding of the domain-specific concepts, terminologies, processes, and challenges. This knowledge is essential for several aspects of a machine learning project. Leveraging domain expertise can guide the selection of features, data preprocessing, and selection of metrics. Subject-matter experts can provide valuable insights into which features are likely to be relevant for the problem at hand.

False Data Injection (FDI): The unauthorized falsification of OT data and signals critical to the operation of a cyber-physical system with the intension to distract or mislead the operators.

Identification: The capability to establish or label as a distinct state or class of states. For example, "this is a (an abnormal) state." Identification is a subset of characterization.

Level 1: A binary classifier that is a component of the composite classifier. It takes in OT data as input and classifies it as either normal (output zero) or abnormal (output one). If the OT data is found to be abnormal then the composite classifier logic further classifies the abnormal state at Level 3.

Level 2: A binary classifier that is a component of the composite classifier. It takes in IT data as input and classifies it as either normal (output zero) or DoS (output one). If DoS is detected, the output of Level 2 can be combined with the output of Level 3 to detect DoS and FDI simultaneously.

Level 3: A binary classifier that is a component of the composite classifier. It is only triggered if Level 1 identifies an abnormality. It uses OT data as input and classifies it as either Other (output zero) or FDI (output one). If FDI is detected, the output of Level 3 can be combined with the output of Level 2 to detect FDI and DoS simultaneously.

Trip Unavailable: Refers to an event that causes the unacceptable condition of the trip system to act as expected and shutdown the reactor and which results in abnormal state.

Performance Metrics: metrics for assessment of performance of an AI/ML algorithm. For classification, performance metrics typically use are accuracy, precision, recall, F1 score, and ROC curve.