

PNNL-39767

Evaluation of Artificial Intelligence (AI) and Machine Learning (ML) Approaches for Bill-of- Material (BOM) Analysis

September 2026

Corinne G. Roth
William J. Hutton
Fleurdeliza A. de Peralta
Adam Baji



U.S. DEPARTMENT
of **ENERGY**

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.** Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from
the Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062

www.osti.gov
ph: (865) 576-8401
fox: (865) 576-5728
email: reports@osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
or (703) 605-6000
email: info@ntis.gov
Online ordering: <http://www.ntis.gov>

Evaluation of Artificial Intelligence (AI) and Machine Learning (ML) Approaches for Bill-of-Material (BOM) Analysis

September 2026

Corinne G. Roth
William J. Hutton
Fleurdeliza A. de Peralta
Adam Baji

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Evaluation of Artificial Intelligence (AI) and Machine Learning (ML) Approaches for Bill-of-Material (BOM) Analysis

Corinne Roth, Dr. William Hutton, Fleur De Peralta
Pacific Northwest National Laboratory
Richland, WA

corinne.roth@pnnl.gov, will.hutton@pnnl.gov, fleurdeliza.deperalta@pnnl.gov

Adam Baji
University of Maryland, Baltimore County
Baltimore, MD
abaji1@umbc.edu

Abstract—Advanced reactors rely on complex programmable digital devices whose hardware and software supply chains can hide cybersecurity issues. This paper examines three areas of cybersecurity concern; common cause failure (CCF), diversity, and defense-in-depth (D^3). Ensuring D^3 requires deep visibility into hardware bills-of-materials (HBOMs) and software bills-of-materials (SBOMs). However, BOMs are often in a proprietary format, have inconsistent features, and can contain inexact matches. Furthermore, vendors do not consistently make BOMs available to customers.

This paper compares artificial intelligence (AI) and machine learning (ML) approaches to detect CCF and D^3 risks by analyzing HBOM and SBOM data at scale. AI/ML models help quantify risks that traditional cybersecurity review cannot easily detect by comparing component lineages, identifying shared dependencies, and uncovering hidden homogeneity. These methods support vendors, regulators, advanced reactor designers, and operators by facilitating cybersecurity analysis, improving component and supply chain transparency, and strengthening D^3 across diverse programmable digital devices across multiple systems throughout advanced reactors.

Index Terms—Artificial intelligence, Advanced reactors, cybersecurity, programmable digital devices, bills of materials

I. INTRODUCTION

The Advanced Reactor Safeguards and Security program supports research to help advanced nuclear reactor vendors address near-term challenges in meeting U.S. requirements for domestic Material Control and Accounting (MC&A), Physical Protection Systems (PPS), and cybersecurity for reactor construction [15]. Advanced reactors depend on programmable digital devices with intricate hardware and software supply chains, which can make it difficult to detect hidden common cause failures (CCF) [8]. Achieving Diversity and Defense-in-Depth (D^3) depends on having comprehensive insight into both Hardware Bills of Materials (HBOMs) and Software Bills of Materials (SBOMs) [12]. However, these documents are proprietary, which limits data availability [16].

HBOMs and SBOMs represent a system of components, sub-components, and dependencies. These nested frameworks generate large volumes of data. The amount of data is difficult to parse manually. Furthermore, it is difficult to determine if

there are any shared components or if there is true diversity amongst the devices when the dataset is large. Vendor diversity in components does not guarantee diversity across components and may hide CCF. The ability to distinguish differences between components and sub-components is also difficult; some may appear to be different but are in fact the same.

The ARSS “AI/ML Models for CCF & D^3 Analysis” research project investigates using artificial intelligence and machine learning (AI/ML) algorithms to analyze CCF and D^3 issues based on HBOMs and SBOMs [5]. Emerging AI/ML models handle large, complex datasets [3]. AI/ML models help quantify risks that traditional engineering review cannot easily detect [3]. This paper discusses data limitations and evaluates AI/ML approaches for CCF and D^3 analysis.

II. BACKGROUND

Advanced reactors consist of programmable digital devices across multiple non-safety related systems [5]. These devices are made up of many different software and hardware components [5]. The supply chains of these devices are complex; they can be entirely unique for individual devices, nearly identical for many devices, or have similarities across multiple components [5].

The complex and intricate nature of these supply chains does not ensure diversity; diversity in vendors does not automatically equate to diversity across components in a programmable digital device [5]. Therefore, it is important to compare HBOMs and SBOMs when performing CCF and D^3 analysis [5]. In contrast, the depth of HBOMs and SBOMs content can be exhaustive, making it difficult for humans to effectively and efficiently review [5]. So, this project is exploring AI/ML approaches for performing CCF and D^3 analysis of HBOMs and SBOMs [5].

A. CCF & D^3

A common cause failure is “the simultaneous failure of two or more redundant components, systems, or safety functions due to a single shared cause or coupling mechanism” [5], [8]. Common elements that can lead to failures include deficiencies in structure or components, environmental hazards, or mishaps

made by humans [5]. In contrast, cybersecurity related failures include software packages used by different vendors or human error due to misconfiguration [5], [7].

Defense-in-depth is a safety and security strategy consisting of multiple overlapping layers to delay and deter failures [5], [12]. Diversity lowers the probability that one common cause will disable several systems at the same time [5], [11]. Intertwining these two concepts strengthens the resiliency of advanced reactors [5].

B. HBOMs & SBOMs

HBOMs are a comprehensive inventory that encompasses both physical and logical elements constituting a hardware product or system [1]. It accounts for not only the completed device, but also its internal components and associated dependencies [1]. SBOMs, by contrast, provide a comprehensive inventory of all components, libraries, and modules within a software product in a machine-readable format [6]. This record enables organizations to efficiently track software elements, identify potential vulnerabilities, and manage security risks throughout the supply chain [6].

Typical elements found in HBOMs are processors, storage modules, manufacturer and supplier details, and version numbers [1]. In contrast, SBOMs generally include component names, versions, hashes, unique IDs, dependencies, and license information [6].

HBOMs and SBOMs give vendors, subject matter experts (SMEs), cybersecurity analysts, and other reviewers deep visibility into the operating systems [5]. This strengthens the vulnerability and risk management capabilities by revealing the components in the system [5]. Hence, HBOM and SBOM analysis is one of the leading practices for cybersecurity, infrastructure security, and technology security [10].

C. Data Limitations

Securing the nation's supply chain involves generating detailed HBOMs and SBOMs across critical manufacturing and infrastructure [13]. However, advanced nuclear reactor HBOMs and SBOMs are unavailable because their components are considered proprietary [5], [16].

Open-source HBOMs are virtually non-existent due to the proprietary nature of the components, which can be considered a competitive advantage. In contrast, open-source SBOMs are available [14]. However, these SBOMs are not representative of potential software found in advanced nuclear reactor systems.

The lack of HBOMs and SBOMs that represent programmable digital devices for advanced nuclear reactors does not limit this project's ability to evaluate AI/ML cybersecurity analysis. A BOM is a BOM no matter which components the BOM represents. Using AI/ML for CCF and D^3 analysis is domain-agnostic. If an AI algorithm is found to be well suited to BOM data and analysis in one domain (e.g. network switches), given the content of a BOM, it is expected that the approach will directly map to BOMs used in other application domains (e.g., nuclear reactors). This is especially true

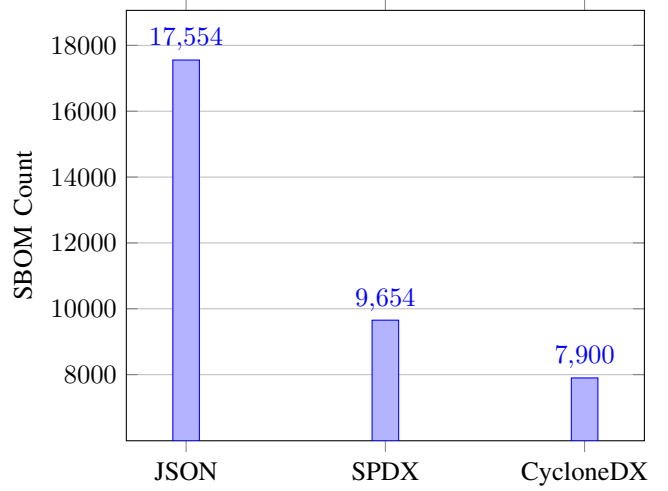


Fig. 1. SBOM Formats

for general purpose programmable digital devices such as a programmable logic controller (PLC).

This project obtained a collection of open-source SBOMs as a proxy dataset for software used in advanced reactors for AI/ML evaluation. The final BOM collection dataset is a large-scale corpus of software bills-of-material files gathered from multiple sources and formats [14]. It contains 17,554 JSON SBOMs, 9,654 SPDX documents and 7,900 CycloneDX documents, representing a wide range of software systems such as container images, Linux distributions, language runtimes, and application stacks.

III. AI/ML APPROACHES

AI refers to computational methods that perform tasks associated with human reasoning and pattern recognition. ML refers to a subset of AI, which uses data to identify patterns without relying solely on predefined models. Unsupervised ML is relevant to BOM analysis because it can identify structures in data without labeled examples [2].

HBOMs and SBOMs can contain thousands of components and nested dependencies. AI/ML can help organize these data, identify shared components, and prioritize potential CCF pathways.

This project's dataset contains over 35,000 SBOMs [14] in various formats (e.g., JSON, SPDX, and CycloneDX—see Fig. 1). The dataset is large and heterogeneous but does not contain authoritative labels identifying duplicates, diverse systems or CCF relationships. Therefore, supervised ML is not an immediate fit for cybersecurity analysis.

BOMs also contain relational information because components can depend upon other components. Methods should therefore consider both overall system similarity and dependency structure [6]. Graph-based analysis, clustering, and large language models (LLMs) offer useful but different capabilities for this problem, each of which is explored in detail below.

TABLE I
AI/ML SUITABILITY TO BOM DATA

Approach	Primary Use	Suitability	Key limitation
Graph-based analysis	Identify shared direct and transitive dependencies	High	Depends on complete and accurate dependency data
Hierarchical clustering	Group similar BOMs and identify near-duplicates	High	Results depend on feature selection and similarity measures
Supervised learning	Classify relationships or predict CCF relevance	Low	Requires authoritatively labeled training data
Large-language models	Extract information, suggest aliases, and summarize results	Moderate	May produce unsupported conclusions and requires verification
Hybrid	Use clustering to find related systems and graph analysis to explain commonalities	High	More complex to implement and validate

A. Graph-Based Analysis

BOMs can be represented as graphs with components shown as nodes and dependencies shown as edges [4]. This representation preserves the relationships among components instead of treating each BOM as a flat list.

Graph analysis can identify direct and transitive dependencies shared by multiple systems. For example, two systems may have different top-level applications but rely on the same library, runtime, or operating-system component [12]. These shared dependencies may not be obvious from component names alone but could be relevant to a D³ analysis.

Graph measures can also help identify widely used or highly connected components. Common-subgraph analysis can find groups of components and dependency paths that appear in more than one system [4]. The main limitation is that the results depend on the completeness of all BOMs used in the analysis. Missing dependency information can hide commonalities or make systems appear more diverse than they are.

B. Hierarchical Clustering

Hierarchical clustering, which is a type of unsupervised learning, groups BOMs based on similarity without requiring predefined labels. Features could include component names, package names and dependency counts, vendors, software ecosystems, versions, and graph characteristics.

Hierarchical clustering is useful because it shows relationships at several levels. Analysts can examine broad families and then narrow the analysis to closely related systems. Density-based methods such as hierarchical density-based spatial clustering of applications with noise (HDBSCAN) may also work well because they can identify groups of near-duplicate BOMs without forcing every system into a cluster [9].

Clustering can help find products that have different names or vendors but contain similar software. However, the results depend heavily on the features and similarity measure used. Systems placed in the same cluster are similar according to those selected features, but they do not necessarily share a failure mode.

C. Large-Language Models

Large-language models (LLMs) are computer programs trained to read, understand, and generate human-like text. LLMs could be used to support portions of BOM analysis that involve unstructured or inconsistent text. Possible uses include extracting component information from technical documents, suggesting aliases for differently named components, and summarizing the differences between two BOMs. An LLM could also provide a natural-language interface for querying analysis results [2].

These models are less suitable for calculating component overlap or tracing dependency paths, which can be handled more reliably with deterministic and graph-based methods. LLMs may also produce unsupported relationships or overlook important differences between component versions. Any LLM-generated match should therefore be checked against identifiers, hashes, versions, supplier information, and source records.

For this problem, LLMs would be most useful as an analyst-support tool rather than the primary analysis method [2]. Their role should be limited to extracting information, proposing candidate matches, and explaining results produced by more traceable methods.

IV. FUTURE WORK

The next steps for this project are to evaluate the effectiveness of the AI/ML approaches identified in this paper for CCF and D³ analysis. Prior to experimentation, development code to parse and preprocess the SBOM dataset is needed to ensure that any empty or incomplete SBOMs are removed to maintain the integrity of subsequent AI/ML analyses.

Once the dataset has been cleaned, systematic application and comparison of the discussed AI/ML methods, including graph-based analysis e.g., graph clustering, community detection, and graph embedding for similarity search to identify shared components, hierarchical clustering approaches like HDBSCAN to detect near-duplicate systems, and hybrid strategies that combine hierarchical clustering to group related systems with graph analysis to explain commonalities.

The results of these evaluations will inform the suitability of each approach for large-scale SBOM analysis, supporting supply chain transparency and cybersecurity risk management in advanced reactor systems.

V. CONCLUSION

Common cause failure, diversity, and defense-in-depth analysis of programmable digital devices found in advanced reactors is a challenging cybersecurity problem. Current artificial intelligence and machine learning approaches offer solutions for supply chain transparency and efficient, effective cybersecurity analysis. Next steps for this project are to evaluate the effectiveness of suitable AI/ML approaches identified in Table I for CCF and D³ analysis. Demonstrating the effectiveness of AI/ML using a large SBOM dataset will validate this type of cybersecurity analysis when the advanced reactor industry develops their own BOMs. AI/ML analysis will strengthen situational awareness, transparency, and help mitigate supply chain risk management in advanced reactor non-safety systems. It is highly likely this method of AI/ML and BOMs cybersecurity analysis directly maps to other industries and other cybersecurity questions as well.

VI. ACKNOWLEDGEMENTS

Microsoft Copilot and OpenAI ChatGPT were used to assist with grammar, clarity, conciseness, organization and formatting. All technical content, interpretations, and final revisions were approved and reviewed by the authors. AI was not used as a substitute for judgement or source verification.

REFERENCES

- [1] CyberNX. From Chips to Firmware: Understanding HBOM in Modern IT Environments. [Online]. Available: <https://www.cybernx.com/en-us/hbom-guide/>
- [2] Databricks. LLM Vs AI: A Practical Guide to Differences, Use Cases, and Tools. [Online]. Available: <https://www.databricks.com/blog/llm-vs-ai>
- [3] S. Fazzino. (2023, October) Cybersecurity Defense-in-Depth Using AI and ML. [Online]. Available: <https://www.jackhenry.com/fintalk/cybersecurity-defense-in-depth-using-ai-and-ml>
- [4] W. Hamilton, *Graph Representation Learning*. Springer, 2020. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-031-01588-5>
- [5] W. Hutton, J. Smith, C. Roth, and F. D. Peralta, "Evaluation of Hardware and Software Bill of Materials (HBOMs/SBOMs) Extraction Methods," Pacific Northwest National Laboratory, Tech. Rep., August 2026.
- [6] IBM. What is a software bill of materials (SBOM)? [Online]. Available: <https://www.ibm.com/think/topics/sbom>
- [7] Infosec. (2023, August) Understanding Cybersecurity Breaches: Types, Common Causes and Potential Risks. [Online]. Available: <https://www.infosecinstitute.com/resources/general-security/understanding-cybersecurity-breaches-types-common-causes-and-potential-risks/>
- [8] S. Kumar, "Chapter 8 - common cause failures and dependency modeling in single and multiunit npp sites," in *Reliability and Probabilistic Safety Assessment in Multi-Unit Nuclear Power Plants*. Academic Press, 2023, pp. 201–220. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780128193921000126>
- [9] C. Malzer and M. Baum, "A hybrid approach to hierarchical density-based cluster selection," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2020, pp. 223–228.
- [10] NIST. Software Security in Supply Chains: Software Bill of Materials (SBOM). [Online]. Available: <https://www.nist.gov/itl/executive-order-14028-improving-nations-cybersecurity/software-supply-chain-security-guidance-20>
- [11] R. W. (ORNL), R. (ORNL), M. C. (ORNL), D. H. (ORNL), K. K. (ORNL), A. L. (ORNL), G. T. M. (ORNL), M. M. (ORNL), J. M. (ORNL), I. O. W.P. Poore, A. Q. (ORNL), T. W. J. (ORNL), and M. W. rman (NRC), "Diversity Strategies for Nuclear Power Plant Instrumentation and Control Systems (NUREG/CR-7007, ORNL/TM-2009/302)," Oak Ridge National Laboratory, Tech. Rep., February 2010.
- [12] G. G. Preckshot, "Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems (NUREG/CR-6303, UCRL-ID-119239)," Lawrence Livermore National Laboratory, Tech. Rep., December 1994.
- [13] President of the United States. (2026, July) Securing America's Defense Supply Chains and Ensuring Domestic Acquisition of Critical Materials (Executive Order 14415). [Online]. Available: <https://www.whitehouse.gov/presidential-actions/2026/07/securing-americas-defense-supply-chains-and-ensuring-domestic-acquisition-of-critical-materials/>
- [14] sbomex. S bom explorer - discover and pull public sboms. [Online]. Available: <https://github.com/interlynk-io/sbomex>
- [15] SNL. Advanced Reactor Safeguards and Security. [Online]. Available: <https://energy.sandia.gov/programs/nuclear-energy/safety-security-and-safeguards-for-advanced-nuclear-power/advanced-reactor-safeguards-and-security/>
- [16] B. C. (SNL), D. W. (DOE), K. L. B. (INL), A. E. (SNL), J. R. (SNL), N. S. (SNL), A. T. (SNL), M. R. (SNL), C. L. (SNL), L. M. (SNL), B. K. (SNL), A. H. (SNL), P. M. (SNL), S. E. (INL), R. Y. (INL), C. G. (ANL), N. H. (ANL), B. B. (ANL), R. V. (ANL), L. C. (BNL), Y. C. (BNL), M. C. (LANL), A. L. (LANL), P. G. (ORNL), M. A. (ORNL), S. C. (ORNL), K. H. (ORNL), S. B. (PNNL), and G. F. (SRNL), "Advanced Reactor Safeguards & Security 2024 Program Roadmap," Department of Energy, Tech. Rep., July 2024. [Online]. Available: https://www.sandia.gov/app/uploads/sites/273/2024/08/ARSS-Roadmap-SAND2024_FINAL.pdf

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354

1-888-375-PNNL (7665)

www.pnnl.gov