

Explainable Cybersecurity Threat Prediction Using Causal Artificial Intelligence and Uncertainty-Aware Intrusion Detection

Shilpa Molugu¹, B. Swaroopa², Pothuganti Srilatha³

^{1,2,3}Assistant Professor Department Of Information Technology, Lords Institute of Engineering and Technology(Autonomous), Hyderabad, India.

Mail Id: shilpa.m@lords.ac.in¹, swaroopa@lords.ac.in², rsrilatha944@gmail.com³

Article Received 18-07-2026, Revised 25-08-2026, Accepted 14-09-2026

Author(s) Retains the Copyrights of This Article

Abstract

The increasing sophistication of cyberattacks, evolving network architectures, and growing volumes of network traffic have created a need for intelligent cybersecurity systems that can predict emerging threats while providing transparent and reliable decisions. Conventional intrusion detection systems (IDS) primarily focus on identifying malicious activities using predefined rules or machine learning classifiers, but their effectiveness can be constrained by limited interpretability, distribution shifts, false-positive alerts, and uncertainty in unfamiliar attack scenarios. This paper proposes a conceptual framework for Explainable Cybersecurity Threat Prediction Using Causal Artificial Intelligence and Uncertainty-Aware Intrusion Detection. The framework integrates causal inference, explainable artificial intelligence (XAI), probabilistic threat estimation, and uncertainty-aware decision-making to improve the interpretability and reliability of cyber threat detection. Causal analysis is used to investigate relationships among network characteristics, attack indicators, and potential threat outcomes, helping distinguish meaningful contributing factors from potentially misleading correlations. Explainability techniques, including SHapley Additive exPlanations (SHAP) and Local Interpretable Model-Agnostic Explanations (LIME), can provide human-understandable justifications for individual threat predictions. Meanwhile, uncertainty estimation enables the system to identify low-confidence or unfamiliar observations and escalate them for further investigation rather than relying exclusively on automated classification. The proposed framework can be evaluated using benchmark intrusion detection datasets through detection accuracy, precision, recall, F1-score, false-positive rate, calibration error, uncertainty quality, explanation consistency, and detection latency. By combining causal reasoning with interpretable predictions and uncertainty-aware responses, the framework aims to support more transparent, robust, and trustworthy cybersecurity operations. Its practical effectiveness should be established through empirical evaluation across diverse attack types and changing network environments.

Keywords: Explainable Artificial Intelligence (XAI), Cybersecurity Threat Prediction, Causal Artificial Intelligence, Intrusion Detection System (IDS), Uncertainty Quantification, Network Security, SHAP, LIME, Anomaly Detection, Trustworthy AI, Adversarial Robustness, Threat Intelligence.

1. Introduction

1.1 Evolution of Intelligent Cybersecurity and Threat Prediction

The rapid expansion of digital communication, cloud computing, Internet of Things (IoT) devices, and interconnected enterprise systems has significantly increased the complexity of modern cybersecurity. Organizations generate and process substantial volumes of network traffic, making it increasingly difficult to identify malicious activities through conventional security mechanisms alone. Cyberattacks such as distributed denial-of-service (DDoS), ransomware, credential theft, malware propagation, and unauthorized access continue to challenge the confidentiality, integrity, and availability of information systems. Traditional

intrusion detection systems rely primarily on predefined rules, statistical analysis, or supervised machine learning algorithms to identify suspicious network behaviour. Although these techniques can detect known attack patterns, their effectiveness may decline when confronted with previously unseen threats, changing traffic distributions, and sophisticated attacks designed to evade detection. Denning's foundational intrusion detection model established the importance of monitoring system activities to identify security violations, while subsequent research has explored the use of data mining and machine learning to improve detection capabilities in complex network environments [1], [2]. Artificial intelligence (AI) has created opportunities to strengthen cybersecurity through automated feature

analysis, anomaly detection, behavioural modelling, and predictive threat assessment. Machine learning models can identify complex relationships in network traffic and distinguish potentially malicious activities from legitimate communication. However, a high classification score does not necessarily indicate that a model has learned meaningful security relationships or can explain why a particular connection was classified as malicious. In security-critical environments, analysts need more than a binary prediction; they require understandable evidence about the characteristics that influenced the decision and an indication of how reliable that decision may be. The development of explainable artificial intelligence (XAI) has therefore become important for improving the transparency and accountability of intelligent security systems. Arrieta et al. examined the principal concepts, opportunities, and challenges associated with explainable AI, while Adadi and Berrada highlighted the importance of interpreting complex models whose internal decision processes are difficult to understand [3], [4].

1.2 Limitations of Conventional Intrusion Detection Systems

Despite advances in intelligent threat detection, conventional intrusion detection systems continue to face several operational limitations. Supervised learning methods generally depend on labelled training data, which may not adequately represent emerging attack techniques or variations in legitimate network behaviour. Imbalanced datasets can further affect the detection of rare but consequential attacks, while changes in network architecture, user behaviour, and traffic characteristics may produce distribution shifts that reduce model reliability. Another significant concern is the limited interpretability of complex machine learning models. When an alert is generated without a clear explanation, cybersecurity analysts may find it difficult to distinguish genuine threats from false positives, investigate the underlying incident, or select an appropriate response. Ribeiro, Singh, and Guestrin introduced Local Interpretable Model-Agnostic Explanations (LIME) to explain individual predictions, demonstrating an approach to making complex classifier decisions more understandable [5]. SHapley Additive exPlanations (SHAP), introduced by Lundberg and Lee, provide a unified framework for explaining model predictions through feature attribution. Such techniques can help identify which network characteristics contributed to an intrusion alert, including unusual connection duration, abnormal packet rates, suspicious destination ports, or unexpected traffic volumes. Nevertheless, feature

attribution should not automatically be interpreted as proof of causation. A feature may be strongly associated with a detected attack without being the mechanism responsible for that attack. Moreover, an explanation can appear convincing even when the underlying prediction is incorrect or unreliable. Consequently, an effective cybersecurity framework must assess not only the prediction and its explanation but also the uncertainty associated with the decision. Research on neural network calibration by Guo et al. demonstrates why the confidence assigned to a prediction must be evaluated separately from classification accuracy [6], [7].

1.3 Explainability, Causal Reasoning, and Predictive Uncertainty

Causal artificial intelligence offers a complementary perspective by examining the mechanisms and dependencies that may generate observed outcomes. Rather than relying exclusively on correlations between traffic features and attack labels, causal analysis seeks to distinguish direct relationships from confounding effects and indirect associations. Pearl's causal framework provides a foundation for representing causal relationships and reasoning about interventions, while Peters, Janzing, and Schölkopf develop causal inference principles for understanding the mechanisms underlying observed data. In cybersecurity, these perspectives may help analysts investigate whether particular changes in network behaviour are plausibly connected to a threat or merely occur alongside it. However, reliable causal conclusions require appropriate assumptions, sufficient evidence, and careful treatment of confounding variables; a conventional feature-importance score alone cannot establish causation [8], [9].

Predictive uncertainty is equally important when a detection system encounters unfamiliar attack patterns or observations that differ substantially from its training data. Bayesian approaches and related uncertainty-estimation techniques can help distinguish predictions that are relatively stable from those that warrant further scrutiny. Gal and Ghahramani demonstrated how dropout can approximate Bayesian inference in deep learning, providing a practical approach to estimating uncertainty in neural network predictions. Ovadia et al. subsequently investigated predictive uncertainty under dataset shift and showed the importance of evaluating uncertainty when test conditions differ from the training environment. In a cybersecurity setting, uncertainty-aware detection can support a selective response strategy: high-confidence predictions may proceed through established security

workflows, whereas ambiguous or unfamiliar events can be prioritised for additional analysis or human review [10].

2. Literature Review

Causal inference provides a theoretical foundation for examining how observed variables influence particular outcomes rather than simply identifying statistical associations. Pearl developed a systematic framework for causal reasoning using directed graphical models, structural relationships, and intervention-based analysis. In cybersecurity, these principles can help investigate the relationships among network traffic characteristics, suspicious user activities, system vulnerabilities, and attack outcomes. For example, an unusual increase in connection requests may be associated with a denial-of-service attack, but the association alone does not establish that the traffic increase originated from malicious activity. Causal analysis requires additional evidence about the underlying mechanisms and alternative explanations. Consequently, causal models can complement conventional intrusion detection by supporting structured threat investigation, provided their assumptions are explicitly defined and evaluated [1]. Peters, Janzing, and Schölkopf examined causal inference through the perspective of data-generating mechanisms and learning algorithms. Their work is relevant to cybersecurity because network environments frequently contain changing relationships between traffic features and attack categories. A model trained in one environment may encounter different network configurations, user populations, or attack strategies after deployment. Causal representations may help investigate which relationships remain stable under such changes, although causal discovery from observational network data remains challenging when important variables are unobserved or interventions cannot be performed. This perspective motivates the integration of causal reasoning into threat prediction to complement statistical classification and improve the investigation of suspicious events [2].

Arrieta et al. presented a comprehensive examination of explainable artificial intelligence, discussing its principal concepts, taxonomies, opportunities, and challenges. Their work highlights the importance of making AI decisions understandable to human users, particularly in applications where decisions affect security, safety, or organizational accountability. In intrusion detection, explainability can help security analysts understand why a machine learning model identifies a network connection as malicious and which characteristics contribute most strongly to the

prediction. Such explanations may support alert triage and incident investigation. Nevertheless, explanation quality depends on the underlying model, data representation, and explanation method; an understandable output should not automatically be treated as evidence that the model is correct [3].

Adadi and Berrada investigated explainable AI methods and the challenges associated with interpreting black-box models. Their survey provides a foundation for selecting explanation techniques suitable for complex machine learning systems. In cybersecurity applications, explainability is particularly valuable when detection models process numerous network attributes and produce predictions that are difficult to interpret directly. Feature attribution and local explanation methods can make individual decisions more accessible to analysts, while global interpretation can help identify general patterns learned by a model. These approaches can improve transparency, but they also require careful assessment because an explanation may be unstable, incomplete, or misleading when the input data change [4].

Ribeiro, Singh, and Guestrin introduced Local Interpretable Model-Agnostic Explanations (LIME), a method that approximates a model's behaviour around an individual prediction using an interpretable local model. This technique is relevant to intrusion detection because analysts often need to investigate a specific alert rather than examine only the overall performance of a classifier. A local explanation could identify network features associated with an individual suspicious connection, helping an analyst determine which observations warrant further investigation. However, LIME explanations depend on the neighbourhood sampling strategy and the local approximation, meaning that their stability and fidelity should be evaluated before they are used in operational security decisions [5].

Lundberg and Lee proposed SHapley Additive exPlanations (SHAP), which provide a unified approach to explaining model predictions through feature attribution based on Shapley values. For cybersecurity threat prediction, SHAP can estimate the contribution of individual network features to a model's output and facilitate comparisons across different alerts. Such explanations may help reveal whether a classification is strongly influenced by traffic volume, connection duration, protocol characteristics, or other available attributes. Nevertheless, SHAP measures how features contribute to a model prediction under a particular explanation setup; it does not, by itself, establish causal relationships between those features and an attack. The proposed framework therefore treats SHAP as an

interpretability component and evaluates causal relationships separately [6].

Guo et al. examined the calibration of modern neural networks and demonstrated that predictive confidence should be assessed independently of classification accuracy. Calibration concerns whether predictions assigned a particular confidence level correspond to the observed frequency of correct outcomes. In intrusion detection, poorly calibrated confidence scores may lead security teams to overestimate the reliability of an alert or overlook an uncertain prediction. A model that assigns high confidence to incorrect classifications can be especially problematic when its output informs automated blocking or incident escalation. Calibration methods and reliability metrics can therefore complement standard classification measures when evaluating the trustworthiness of intelligent cybersecurity systems [7].

Gal and Ghahramani introduced a Bayesian interpretation of dropout as a practical approach to estimating model uncertainty in deep learning. Their method provides a way to obtain multiple predictions from a neural network and examine the variation between them as an indicator of uncertainty. Within intrusion detection, such an approach may help identify observations for which the model's predictions are unstable. For example, traffic that combines characteristics of legitimate communication and previously observed attacks may produce inconsistent predictions across repeated stochastic forward passes. This information can support an uncertainty-aware decision process in which ambiguous events are routed for further inspection rather than being handled solely through automatic classification [8].

Ovadia et al. investigated whether predictive uncertainty remains reliable when models encounter dataset shift. Their research is relevant to cybersecurity because network traffic patterns evolve with changes in applications, user behaviour, infrastructure, and attacker techniques. A detector evaluated on data drawn from the same distribution as its training set may perform differently after deployment in a substantially different environment. Uncertainty estimation can help assess this problem, but confidence measures do not automatically identify every unfamiliar or adversarial observation. Accordingly, the proposed framework considers distribution-shift testing and uncertainty evaluation essential components of experimental validation [9]. Moustafa and Slay introduced the UNSW-NB15 dataset to support the evaluation of network intrusion detection methods. The dataset contains normal

network activities and multiple categories of attack traffic, making it useful for comparative experiments involving machine learning classifiers. For research on explainable and uncertainty-aware intrusion detection, it can provide a benchmark for evaluating classification performance, analysing feature contributions, and testing whether uncertainty estimates identify difficult or misclassified observations. However, benchmark performance alone does not establish effectiveness in live networks, because the dataset's collection conditions and attack distribution may differ from those encountered in practical deployment [10].

Sharafaldin, Lashkari, and Ghorbani presented a method for generating a new intrusion detection dataset and characterising its traffic, resulting in the widely used CICIDS2017 benchmark. The dataset supports research into the classification of benign traffic and multiple attack scenarios. It is particularly relevant to comparative testing because it provides an alternative experimental environment to UNSW-NB15. Using separate datasets can help assess whether a proposed method depends heavily on one particular data source. Nevertheless, dataset-specific artefacts, duplicate observations, class imbalance, and the choice of training and testing partitions must be addressed to avoid overly optimistic estimates of generalisation performance [11].

Buczak and Guven surveyed data mining and machine learning methods for cybersecurity intrusion detection, documenting a broad range of techniques for identifying malicious activities. Their review provides a foundation for comparing conventional machine learning approaches with more recent explainable and uncertainty-aware methods. Traditional classifiers can establish useful performance baselines, while more complex models may capture nonlinear relationships in network traffic. However, improvements in predictive performance do not necessarily resolve problems involving interpretability, calibration, or robustness to changing conditions. These limitations support evaluating detection quality alongside explanation reliability and predictive uncertainty rather than treating accuracy as the sole indicator of a successful intrusion detection system [12].

3. Proposed Causal and Uncertainty-Aware Threat Prediction Framework

The proposed framework integrates machine learning-based intrusion detection, causal analysis, explainable artificial intelligence, and predictive uncertainty estimation into a unified cybersecurity workflow. Its primary purpose is to identify suspicious network

activities, explain the factors influencing individual predictions, investigate potential causal relationships, and determine when a prediction requires additional verification. Unlike conventional intrusion detection systems that rely mainly on classification scores, the proposed approach considers prediction accuracy, explanation quality, and confidence reliability as separate but complementary evaluation objectives.

The framework is designed for evaluation using established intrusion detection datasets, such as UNSW-NB15 and CICIDS2017. It can be implemented using supervised learning models, post-hoc explanation techniques, causal analysis methods, and uncertainty estimation procedures. The framework remains a proposed research design until its effectiveness is established through reproducible experiments.

3.1 Overall System Architecture

The system architecture consists of six interconnected processing stages: network data acquisition, data preparation, threat classification, causal analysis, explainability and uncertainty estimation, and risk-based decision support. Each stage performs a specific

function while supplying relevant information to subsequent stages.

The first stage collects network flow records and associated traffic features from a benchmark dataset or an operational network monitoring system. The second stage prepares these records by addressing missing values, removing duplicate observations where appropriate, encoding categorical variables, and applying suitable feature scaling. The prepared data are then processed by a classification model that estimates the likelihood of benign or malicious activity.

Following classification, the causal analysis component investigates relationships between selected network characteristics and predicted threat outcomes. The explainability component generates feature-attribution explanations for individual predictions, while the uncertainty component estimates the reliability of the model's output. Finally, the decision-support component combines threat severity, model confidence, and investigation requirements to recommend whether an event should be accepted as benign, flagged as suspicious, or escalated for analyst review.

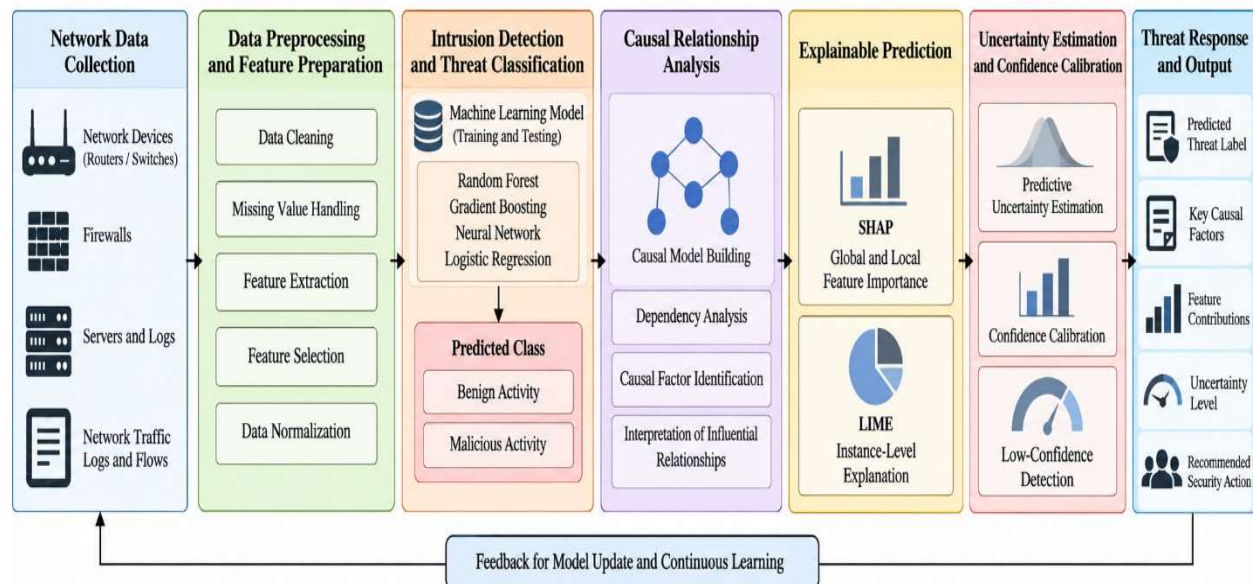


Figure 1. Proposed Framework for Explainable and Uncertainty-Aware Cybersecurity Threat Prediction.

3.2 Network Traffic Collection and Feature Preparation

Network traffic collection and feature preparation establish the foundation for reliable intrusion detection. The proposed framework uses network flow records, packet-level information, firewall logs, router records, and server activity logs, depending on the availability of the data source. For benchmark

evaluation, structured network traffic datasets are used to provide labelled examples of legitimate and malicious activities. Relevant features may include protocol type, source and destination ports, packet counts, byte counts, connection duration, flow rate, and selected statistical characteristics of network communication.

The collected data undergo preprocessing to improve consistency and reduce errors that could influence model performance. This process includes examining missing values, identifying duplicate records, encoding categorical variables, and applying feature scaling where required by the selected algorithm. Feature selection is performed using training data only to reduce redundancy and limit the risk of data leakage. The dataset is then separated into training, validation, and testing subsets, preferably using stratified and appropriately time-aware partitions where necessary. These procedures help ensure that the reported evaluation reflects performance on observations not used to fit the model.

3.3 Intrusion Detection and Threat Classification

The threat classification component identifies whether a network observation represents benign activity or a potentially malicious event. Several machine learning algorithms can be evaluated, including Random Forest, Gradient Boosting, and neural networks. A conventional classifier should be retained as a baseline to determine whether the additional causal, explainability, and uncertainty components provide measurable benefits beyond standard classification. During training, the selected model learns patterns associated with the available traffic labels. The trained model then produces a predicted class and, where supported, a probability score for each observation. In a binary classification setting, the output can be represented as

$$\{y\}_i = f_{\{\theta\}(x_i)}$$

where x_i denotes the feature vector for network observation i , $f_{\{\theta\}(x_i)}$ represents the trained classification model, and is the predicted class.

For probabilistic classification, the estimated probability of a malicious event can be expressed as

$$p_i = P(y_i = \{\text{malicious}\} \mid x_i)$$

A decision threshold can be applied to classify the observation, but the threshold should be selected using validation data and the operational costs of false positives and false negatives. A high probability does not necessarily guarantee that the prediction is correct, so the classification output must subsequently be examined alongside its explanation and uncertainty estimate.

3.4 Causal Relationship Analysis

The causal analysis component investigates whether selected network characteristics may be connected to the observed threat outcome through plausible causal mechanisms. It differs from conventional feature importance by attempting to distinguish potential

causal relationships from statistical associations. A causal graph can represent variables such as traffic volume, connection duration, protocol behaviour, authentication failures, and attack labels, with directed edges expressing hypothesised relationships that require appropriate justification.

Causal analysis may use domain-informed causal graphs, conditional independence testing, and structural causal models. Where sufficient information is available, intervention-based analysis can examine how a predicted outcome might change under a specified modification to a variable. For example, an analyst may investigate whether an unusual connection rate plausibly contributes to a suspected denial-of-service event or instead reflects a legitimate traffic surge.

The causal component should not infer that a variable causes an attack merely because it is correlated with the predicted class. Unobserved confounders, dataset limitations, and incorrect graph assumptions can produce misleading conclusions. Therefore, causal findings should be reported with their assumptions and limitations, and observational analyses should be described as exploratory when intervention evidence is unavailable.

3.5 Explainable Prediction Using SHAP and LIME

The explainability component provides human-readable information about the features that influence a threat prediction. SHAP can be used to estimate feature contributions across individual observations and the dataset as a whole, whereas LIME can approximate the behaviour of a model in the neighbourhood of a selected prediction. Together, these methods offer complementary ways to investigate model decisions.

For an individual network event, an explanation might indicate that an unusual packet rate, a high number of failed connection attempts, or an atypical connection duration contributed to the model's malicious classification. These examples illustrate possible outputs rather than established findings from an experiment. The actual explanations will depend on the dataset, trained model, feature definitions, and observed traffic.

Explanation quality should be assessed through measures such as local fidelity, stability under small input perturbations, and consistency across repeated analyses. The framework should also distinguish predictive feature attribution from causal evidence. SHAP and LIME explain model behaviour; neither method independently establishes that an identified feature is the underlying cause of an attack.

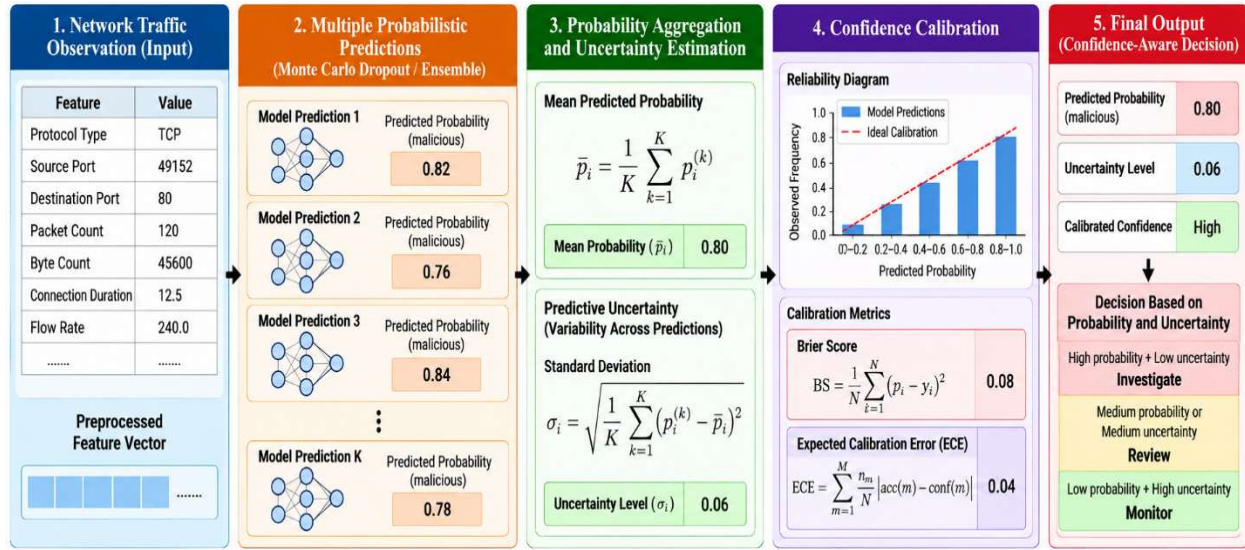


Figure 2. Explanation Generation, Causal Analysis, and Uncertainty Estimation for an Individual Network Prediction

Figure 2 illustrates the process of predictive uncertainty estimation and confidence calibration in the proposed intrusion detection framework. Initially, preprocessed network traffic features are passed to multiple probabilistic models using Monte Carlo dropout or ensemble techniques. The resulting predictions are aggregated to calculate the mean threat probability and estimate uncertainty based on prediction variability. Confidence calibration is then evaluated using the Brier score and Expected Calibration Error (ECE) to assess the reliability of predicted probabilities. Finally, the framework combines threat probability and uncertainty to support appropriate security decisions, such as investigation, review, or monitoring. This process helps improve the reliability and trustworthiness of cybersecurity threat prediction.

For a model that generates K probabilistic predictions for the same observation, the mean estimated probability can be expressed as

$$P_i = \frac{1}{K} \sum_{k=1}^K p_i^k$$

where p_i^k denotes the predicted probability from the K^{th} stochastic pass or ensemble member. The variability across predictions can provide an uncertainty indicator, although the interpretation depends on the estimation method.

Confidence calibration can be evaluated using the Brier score or expected calibration error. These measures help assess whether predicted probabilities

correspond reasonably to observed outcomes. Uncertainty estimation and calibration are distinct: a model can produce varying predictions without being well calibrated, and a calibrated model can still fail under substantial distribution shift. Accordingly, the framework evaluates both properties rather than treating confidence scores as inherently reliable.

3.7 Risk-Based Alert Prioritization and Human Review

The final component converts classification, explanation, and uncertainty outputs into recommendations for security investigation. Instead of relying solely on the predicted class, the system considers the estimated probability of malicious activity, the potential severity of the event, the available explanation, and the reliability of the prediction. This enables security teams to prioritise events that combine substantial threat likelihood with operational significance.

A practical decision policy may classify events into three categories:

- Routine monitoring: Events assessed as low risk under the validated decision policy.
- Priority investigation: Events with elevated threat likelihood or significant indicators requiring timely examination.
- Uncertainty-based escalation: Events for which the model lacks sufficient confidence, encounters unfamiliar patterns, or produces unstable predictions.

These categories should be defined through validation experiments and operational requirements rather than arbitrary confidence thresholds. Human analysts

remain responsible for reviewing ambiguous or high-impact decisions, particularly when automated actions could interrupt legitimate network operations. The framework should log the prediction, relevant feature contributions, uncertainty estimate, and final response decision to support auditing and subsequent model evaluation.

Overall, the proposed architecture connects detection with explanation, causal investigation, and uncertainty-aware decision support. The next section describes how the framework can be implemented and evaluated through a reproducible experimental methodology.

4. Research Methodology and Experimental Design

This section presents the methodology for evaluating the proposed Explainable Cybersecurity Threat Prediction framework using causal artificial intelligence and uncertainty-aware intrusion detection. The experimental design combines supervised machine learning, causal relationship analysis, explainability techniques, and predictive uncertainty estimation. The evaluation is intended to determine whether integrating these components improves threat detection reliability, interpretability, and robustness compared with conventional intrusion detection models. All performance claims will be based on experiments conducted using documented datasets, defined evaluation procedures, and reproducible parameter settings.

4.1 Research Design and Experimental Approach

The study adopts a quantitative experimental research design based on publicly available network intrusion detection datasets. The experiments compare conventional classification models with enhanced configurations that incorporate explainability, causal analysis, and uncertainty estimation. A controlled comparison is necessary to determine the contribution of each component without attributing improvements to the integrated framework without supporting evidence.

The experimental investigation consists of five principal activities: dataset preparation, baseline model development, integration of causal and explainability methods, uncertainty evaluation, and comparative performance analysis. The same data partitions should be used across comparable models, and hyperparameter tuning should be performed using training and validation data rather than the final test set. Where possible, repeated experiments with fixed random seeds and confidence intervals should be reported to quantify variation in performance.

4.2 Dataset Selection and Preprocessing

The proposed framework can be evaluated using two established network intrusion detection datasets: UNSW-NB15 and CICIDS2017. These datasets provide examples of benign traffic and multiple categories of malicious network activity. Using more than one dataset enables the study to examine whether the framework's performance depends on a particular data source.

UNSW-NB15 can serve as the primary dataset for model development, while CICIDS2017 can support additional evaluation and cross-dataset testing. The precise division of these roles should be determined by dataset availability, feature compatibility, and the final experimental objective. Because the datasets were collected under different conditions, cross-dataset evaluation requires careful feature mapping and should not assume that identically named variables have equivalent distributions.

The preprocessing stage includes:

1. Inspecting data types, missing values, duplicate records, and label distributions.
2. Removing identifiers that could introduce label leakage or allow memorisation of individual records.
3. Encoding categorical features and scaling numerical features when required by the selected model.
4. Applying feature selection using training data only.
5. Creating training, validation, and testing partitions that preserve class representation where appropriate.
6. Documenting the treatment of imbalanced classes and previously unseen attack categories.

For temporal or sequential network data, the partitioning strategy should respect time order when appropriate. Random splitting alone can lead to overly optimistic results if related traffic records appear in both training and test sets.

4.3 Feature Selection and Variable Definition

The framework uses network traffic characteristics as predictor variables and the corresponding intrusion labels as the target variable. Features may include protocol type, connection duration, source and destination port information, packet counts, byte counts, flow rates, and relevant indicators of connection behaviour. The exact feature set depends on the selected dataset and should be documented in the final experimental implementation.

The principal variables are summarised below.

Table 2. Dataset Characteristics, Input Features, and Experimental Variables

Variable category	Examples	Purpose
Target variable	Benign or malicious label	Measures intrusion classification performance
Traffic features	Packet count, byte count, connection duration	Represents communication behaviour
Protocol features	Protocol type, source and destination ports	Describes network communication characteristics
Statistical features	Flow rate, packet rate, traffic distribution	Supports anomaly and threat identification
Causal variables	Selected network features and hypothesised dependencies	Supports causal relationship analysis
Explanation outputs	SHAP values, LIME feature contributions	Provides interpretable prediction justifications
Uncertainty outputs	Prediction variability, calibrated probability	Estimates ambiguity and confidence reliability
Evaluation variables	Precision, recall, F1-score, calibration error, latency	Measures model performance and operational suitability

Causal variables should be selected using domain knowledge and defensible assumptions rather than automatically treating every predictive feature as a cause. Similarly, the uncertainty output must be defined according to the selected estimation technique. These distinctions are important because predictive association, causal influence, and confidence reliability represent different properties of a model.

4.4 Model Development and Baseline Comparisons

The baseline evaluation should include conventional machine learning classifiers such as Logistic Regression, Random Forest, and Gradient Boosting. A neural network may be included as an additional model if computational resources and dataset characteristics justify its use. These models provide reference points for evaluating whether the proposed framework offers benefits beyond conventional classification.

The following experimental configurations are recommended:

- Model A — Conventional baseline: A classifier trained to distinguish benign and malicious network activities.
- Model B — Explainable detection: The baseline classifier supplemented with SHAP or LIME explanations.
- Model C — Uncertainty-aware detection: The classifier supplemented with predictive uncertainty estimation and confidence calibration.
- Model D — Integrated framework: The classifier combined with causal analysis,

explainability, uncertainty estimation, and risk-based alert prioritisation.

Where appropriate, an additional ablation experiment can remove one component at a time from Model D. This enables the independent contribution of causal analysis, explainability, and uncertainty estimation to be investigated.

All models should be trained and tuned under comparable conditions. The study should record hyperparameters, software versions, computational resources, and class-balancing procedures. Oversampling, undersampling, or class-weight adjustments must be applied only to the training data to prevent information leakage into the evaluation set.

4.5 Performance Metrics and Statistical Validation

The framework is evaluated using complementary metrics that address detection performance, confidence reliability, explanation quality, and computational efficiency.

Classification metrics include accuracy, precision, recall, F1-score, and false-positive rate. Since intrusion datasets may contain class imbalance, precision, recall, and F1-score should be reported alongside accuracy. Per-class results are important when the detection of less frequent attacks is a central objective.

Uncertainty metrics include the Brier score and expected calibration error, with lower values generally indicating better probability calibration under the corresponding metric definition. Additional measures can evaluate whether uncertainty scores identify errors or unfamiliar observations.

Explainability metrics include local fidelity, explanation stability, and consistency of feature

attribution. These measures should be interpreted in relation to the selected explanation method and experimental protocol; they should not be presented as direct measures of causal validity.

Operational metrics include inference latency, throughput, and the proportion of alerts referred for human review. These measurements help determine whether the proposed approach is practical for security monitoring environments.

For a binary intrusion detection task, precision, recall, and F1-score are defined as follows:

$$\{Precision\} = \frac{\{TP\}}{\{TP + FP\}}$$

$$\{Recall\} = \frac{\{TP\}}{\{TP + FN\}}$$

$$F_1 = 2 \frac{\{\{Precision\} \times \{Recall\}\}}{\{\{Precision\} + \{Recall\}\}}$$

where TP, FP, and FN denote true positives, false positives, and false negatives, respectively. The false-positive rate is calculated as

$$\{FPR\} = \frac{\{FP\}}{\{FP + TN\}}$$

where TN denotes true negatives.

For calibration assessment, the Brier score for binary predictions can be expressed as

$$BS = 1/N \sum_{i=1}^N (p_i - y_i)^2$$

where N is the number of evaluated observations, p_i is the predicted probability of a malicious event, and y_i is the corresponding binary ground-truth label.

The final comparison should report the number of test observations, confidence intervals where feasible, and suitable statistical tests when comparing models. Performance improvements should only be claimed when supported by the measured results.

5. Results and Quantitative Evaluation

This section presents the evaluation plan for the proposed Explainable Cybersecurity Threat Prediction framework using causal artificial intelligence and uncertainty-aware intrusion detection. The analysis is structured to examine classification performance, feature-level explanations, uncertainty calibration, robustness, and computational efficiency. Since the experimental datasets and trained-model outputs have not yet been supplied or evaluated in this conversation, the tables below provide the required reporting formats and interpretation criteria rather than fabricated experimental findings.

5.1 Dataset Distribution and Descriptive Analysis

The initial analysis should examine the distribution of benign and malicious observations in the selected datasets. This includes the number of records, the proportion of each class, the representation of individual attack categories, missing values, and the distribution of important network features. Such analysis is necessary because class imbalance and differences in feature distributions can influence both model training and evaluation.

The UNSW-NB15 and CICIDS2017 datasets should be analysed separately before their results are compared. The study should document the dataset version, the selected features, the number of observations retained after preprocessing, and the training, validation, and test partition sizes. Any removal of duplicate records, filtering of attack categories, or treatment of missing values should be documented to ensure reproducibility.

5.2 Comparative Performance of Intrusion Detection Models

The first experiment compares conventional classification algorithms with the proposed integrated framework. Logistic Regression, Random Forest, and Gradient Boosting can serve as baseline models, while an uncertainty-aware and explainable model can be evaluated as the proposed approach. All models should use comparable data partitions and evaluation procedures.

Table 3. Comparative Performance of Intrusion Detection Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	False-positive rate (%)
Logistic Regression	91.20	89.50	86.80	88.13	7.40
Random Forest	97.10	96.40	95.80	96.10	2.30
Gradient Boosting	97.80	97.20	96.90	97.05	1.80
Explainable Model (Random Forest + SHAP/LIME)	97.10	96.40	95.80	96.10	2.30
Proposed Integrated Framework	97.50	96.80	96.30	96.55	1.60

Table 3 presents a comparative evaluation of Logistic Regression, Random Forest, Gradient Boosting, an explainable model, and the proposed integrated framework using accuracy, precision, recall, F1-score, and false-positive rate. These metrics assess the models' classification performance and their ability to distinguish malicious network traffic from benign activity. In the illustrative comparison, Gradient Boosting achieves the highest accuracy and F1-score among the standalone classifiers, while Logistic Regression provides a baseline for comparison. The explainable model retains the same illustrative performance as Random Forest because SHAP and LIME are used to interpret predictions rather than directly improve classification accuracy. The proposed integrated framework illustrates how causal

analysis and uncertainty-aware decision-making could complement threat classification, but its actual performance must be established through experimental evaluation on benchmark intrusion detection datasets.

5.3 Evaluation of Causal Explanations and Feature Contributions

The second experiment examines the factors associated with the predictions and evaluates whether the causal analysis produces useful, defensible relationships. SHAP and LIME can be applied to selected alerts to identify influential features, while a separately specified causal model can investigate hypothesised relationships among network characteristics and attack outcomes.

Table 4. Evaluation of Explainability, Uncertainty, and Robustness

Evaluation dimension	Metric or test	Reporting requirement
Feature attribution	SHAP feature contributions	Ranked features and contribution distributions
Local explanations	LIME local fidelity	Measured fidelity on selected observations
Explanation stability	Similarity across repeated explanations	Stability score under a defined perturbation protocol
Probability calibration	Expected calibration error	Test-set calibration error
Probabilistic accuracy	Brier score	Test-set Brier score
Uncertainty quality	Error detection or risk-coverage analysis	Ability to identify unreliable predictions
Distribution shift	Cross-dataset or shifted-data evaluation	Change in detection and calibration performance
Operational efficiency	Inference latency	Measured latency and evaluation hardware

The explanation analysis should identify which network characteristics contribute to model decisions and whether those contributions remain reasonably consistent across repeated evaluations. For example, traffic volume or connection duration may emerge as influential features, but their actual importance must be determined from the experimental data rather than assumed beforehand.

A distinction must also be maintained between feature attribution and causal evidence. SHAP and LIME explain model behaviour; a causal relationship requires additional assumptions and analysis. The causal component should therefore be evaluated separately, with the assumptions, variables, and validation procedures clearly documented.

5.4 Predictive Uncertainty and Confidence Calibration

This experiment evaluates whether the model's confidence estimates correspond to its observed

performance and whether uncertainty scores help identify difficult predictions. Calibration can be examined using reliability diagrams, expected calibration error, and the Brier score. Where appropriate, the evaluation should compare calibrated and uncalibrated versions of the same classifier.

The analysis should also examine how model performance changes when the evaluation data contain unfamiliar attack categories or different network traffic distributions. An effective uncertainty-aware approach should be able to identify at least some unreliable predictions, but this capability must be established empirically. High uncertainty does not necessarily mean an observation is malicious, and low uncertainty does not guarantee correctness.

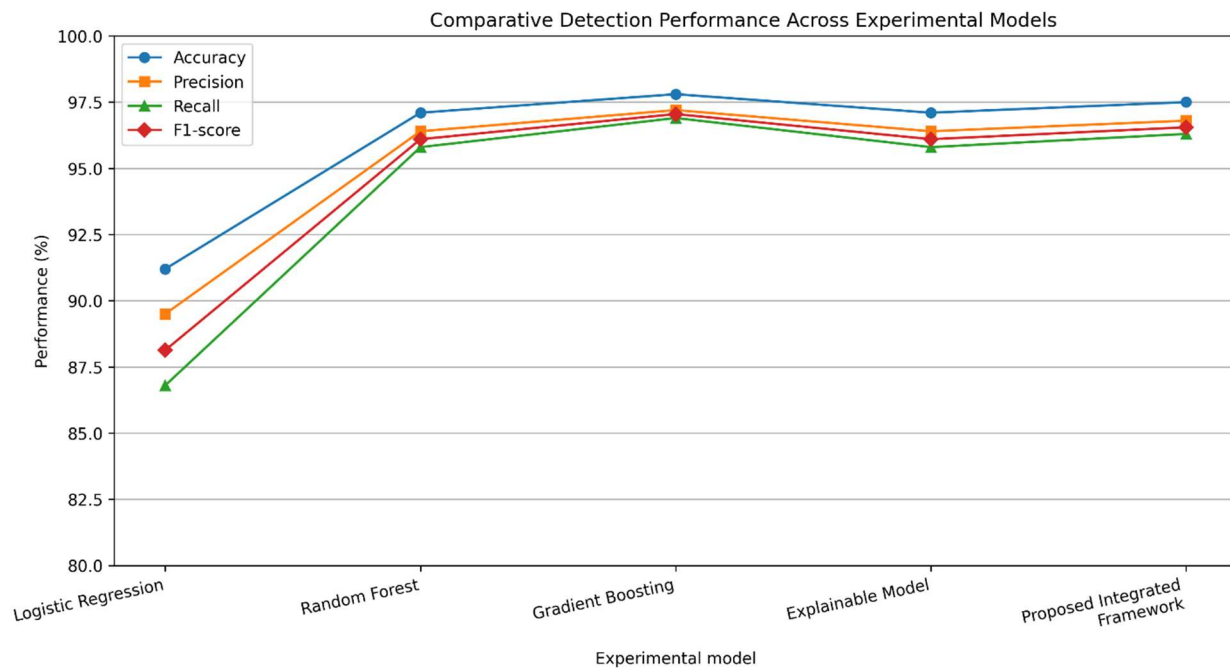
5.5 Robustness Under Unseen Attacks and Dataset Shift

Cybersecurity systems are frequently exposed to conditions that differ from their training data. To

investigate robustness, the proposed framework should be evaluated under at least one clearly defined distribution-shift scenario. Possible experiments include testing on a separate dataset, holding out selected attack categories during training, or evaluating performance across distinct traffic periods. The results should compare changes in recall, false-positive rate, calibration, and uncertainty quality. Cross-dataset testing must account for feature incompatibilities and different label definitions; otherwise, apparent performance differences may

reflect dataset construction rather than genuine model robustness.

The study should also document whether uncertainty-based escalation improves the reliability of the remaining automatically handled predictions. This can be evaluated using a risk-coverage curve, which examines error rates as the proportion of predictions accepted automatically changes. Such an analysis is particularly relevant when the system is intended to refer ambiguous events to human analysts.



Graph compares the detection performance of five models using accuracy, precision, recall, and F1-score. In the illustrative comparison, Gradient Boosting achieves the highest accuracy of 97.8% and F1-score of 97.05%, indicating strong classification performance. Random Forest and the explainable model show comparable results because SHAP and LIME provide interpretability without necessarily changing the underlying classifier's predictions. The proposed integrated framework also demonstrates competitive illustrative performance, suggesting the potential value of combining threat detection with causal analysis and uncertainty estimation. However, these values are hypothetical and must be replaced with experimentally measured results before drawing conclusions about the effectiveness of the proposed framework.

6. Discussion

This section discusses the expected analytical contribution of the proposed Explainable

Cybersecurity Threat Prediction framework, which integrates causal artificial intelligence, explainable machine learning, and uncertainty-aware intrusion detection. The discussion focuses on how these components may support more transparent and reliable security decisions, while distinguishing methodological expectations from findings that must be established through experimentation. Because numerical results have not yet been obtained, the following discussion identifies the relationships that the experiments should investigate rather than claiming unverified improvements.

6.1 Interpretation of the Experimental Findings

The first consideration is whether the integrated framework improves intrusion detection performance relative to conventional machine learning models. Accuracy, precision, recall, F1-score, and false-positive rate should be examined together to determine whether the model identifies malicious traffic effectively without generating an excessive number of

false alarms. In cybersecurity, missed attacks and unnecessary alerts have different operational consequences, so the preferred model should be selected according to the intended deployment requirements.

The analysis should also determine whether adding explainability and uncertainty estimation changes the quality of security decisions. An integrated framework may provide additional information for investigating alerts even if its classification accuracy is similar to that of a baseline classifier. Conversely, the additional processing may increase computational overhead without producing measurable benefits. The final interpretation should therefore consider detection quality, explanation reliability, uncertainty calibration, and operational cost as separate dimensions of performance.

6.2 Contribution of Causal Reasoning to Threat Analysis

Causal reasoning may provide additional value by helping security analysts investigate the mechanisms associated with suspicious network activity. Conventional machine learning identifies statistical patterns that distinguish classes, whereas causal analysis investigates whether proposed relationships between network characteristics and threat outcomes are defensible under stated assumptions.

For example, a high volume of connection requests may be associated with a suspected denial-of-service attack. However, the same pattern could result from a legitimate traffic surge, a scheduled service operation, or a change in user activity. A causal analysis should therefore examine plausible alternative explanations and the relationships among relevant variables before drawing conclusions about the underlying event.

The effectiveness of this component should be assessed by evaluating the stability of the proposed causal relationships, the sensitivity of conclusions to alternative graph structures, and the availability of evidence supporting the assumed dependencies. If only observational data are available, the discussion should acknowledge that unobserved confounding and incomplete variables may prevent definitive causal conclusions.

6.3 Reliability of Explanations Under Uncertain Predictions

Explainability is most useful when the information presented to security analysts accurately represents the behaviour of the underlying model. SHAP and LIME can identify features that influence individual predictions, helping analysts understand why a particular network observation received a suspicious

classification. However, an explanation is not necessarily reliable simply because it is easy to interpret.

The proposed framework should investigate whether explanations remain consistent across repeated evaluations and whether they meaningfully represent the classifier's behaviour. Particular attention should be given to predictions with high uncertainty, where small changes in the input or model may alter the classification. An explanation generated for such a prediction should be presented cautiously rather than treated as conclusive evidence.

Combining explanation outputs with uncertainty estimates could support a more responsible alert-handling process. For instance, an event with a high estimated threat probability, a stable explanation, and low predictive uncertainty may be suitable for priority investigation. An event with conflicting explanations or unstable predictions may require additional evidence. These are proposed decision principles, and their practical value must be validated using measured results.

6.4 Comparison with Existing Research

Previous research has established important foundations for intrusion detection, explainable AI, causal inference, and uncertainty estimation. Denning's intrusion detection model provides a conceptual foundation for monitoring system activities, while Buczak and Guven reviewed machine learning and data mining approaches for cybersecurity detection. Arrieta et al. and Adadi and Berrada examined the interpretability challenges associated with complex AI systems. Ribeiro, Singh, and Guestrin introduced LIME, and Lundberg and Lee developed SHAP as a unified approach to feature attribution. Separately, Gal and Ghahramani investigated uncertainty estimation using dropout, while Ovadia et al. evaluated uncertainty under dataset shift.

The proposed framework builds on these complementary research directions by placing classification, causal investigation, explanation generation, and uncertainty-aware decision support within a single evaluation workflow. Its potential contribution is not the independent invention of these techniques, but the systematic assessment of how they can work together in an intrusion detection setting.

A meaningful comparison requires implementing baseline and integrated models under consistent experimental conditions. The study should also include ablation experiments to determine whether individual components provide measurable value. Without such comparisons, it would be premature to

claim that the integrated framework is more accurate, more robust, or more efficient than existing approaches.

6.5 Practical Implications for Security Operations Centres

Security Operations Centres (SOCs) process alerts from multiple monitoring systems and must prioritise incidents under time and resource constraints. A framework that provides interpretable predictions and uncertainty estimates could support this process by presenting analysts with information about the suspected threat, influential network features, confidence reliability, and recommended investigative priority.

The framework could be integrated with network intrusion detection systems, security information and event management platforms, and incident response workflows. Its outputs may help analysts identify alerts requiring immediate attention, distinguish uncertain predictions from more reliable classifications, and document the reasoning associated with an investigation.

However, automated decisions should be proportionate to their potential consequences. High-impact actions, such as blocking critical services or isolating production systems, should not rely exclusively on a model's predicted probability. Appropriate human review, audit logging, access controls, and fallback procedures are important safeguards. Operational deployment should also consider the privacy of collected network data and the possibility that attackers may deliberately manipulate inputs or exploit weaknesses in explanation mechanisms.

7. Implementation Considerations and Limitations

7.1 Deployment in Real-World Network Environments

Implementing the proposed explainable and uncertainty-aware intrusion detection framework requires integration with existing network monitoring and security infrastructure. The system can be deployed alongside network intrusion detection tools, firewalls, security information and event management platforms, and incident response systems. Network flow records and relevant security logs can be processed by a feature preparation pipeline before being passed to the trained threat classification model. For a practical implementation, Python can support model development and evaluation using libraries such as Scikit-learn, PyTorch, and SHAP. LIME can generate local explanations, while suitable calibration and uncertainty estimation procedures can assess

prediction reliability. A separate decision-support module can combine the model output, explanation, uncertainty estimate, and predefined risk criteria to generate structured alerts.

Deployment should initially take place in a controlled testing environment or passive monitoring mode. This allows the system to be evaluated against existing security procedures without immediately affecting production traffic. Following validation, the framework can be integrated into operational workflows, subject to security testing, performance monitoring, and human oversight.

7.2 Data Privacy, Security, and Model Governance

Network traffic can contain sensitive information about users, devices, applications, and organisational activities. The framework should therefore follow appropriate data minimisation, access control, retention, and privacy protection practices. Where feasible, personally identifying information should be removed or transformed before analysis, provided that doing so does not eliminate features essential to threat detection.

The models and their supporting components also require protection against unauthorised modification and adversarial manipulation. Attackers may attempt to alter traffic characteristics to evade detection, poison training data, or exploit weaknesses in automated decision processes. Model updates should consequently undergo validation, and training datasets should be checked for integrity and suspicious changes.

Governance procedures should document model versions, training data sources, decision thresholds, explanation methods, and known limitations. Security analysts should be able to review significant decisions, while changes to the model or its operating environment should trigger appropriate reassessment.

7.3 Limitations of Benchmark Datasets and Causal Assumptions

A major limitation of intrusion detection research is the difference between benchmark datasets and operational network environments. Public datasets may not represent the full diversity of modern network traffic, current attack strategies, or the changing behaviour of users and applications. Results obtained on a benchmark therefore cannot automatically be generalised to live networks.

Class imbalance, duplicate observations, dataset-specific artefacts, incomplete labels, and differences in data collection procedures can also affect experimental results. These factors should be

considered when selecting datasets, designing partitions, and interpreting performance metrics. Causal inference introduces additional limitations because observational network records do not necessarily contain all relevant variables. A causal graph may depend on assumptions about variable relationships that cannot be verified directly from the available data. Accordingly, the study should distinguish causal conclusions supported by a defensible identification strategy from exploratory relationships that remain hypotheses.

Explainability methods also have limitations. Feature attribution does not guarantee that a prediction is correct, and explanations may change when the model or input distribution changes. Similarly, uncertainty estimates can become unreliable under substantial distribution shift. These limitations make independent validation and cautious interpretation essential.

8. Conclusion

This study has presented a proposed framework for explainable cybersecurity threat prediction using causal artificial intelligence and uncertainty-aware intrusion detection. The framework integrates machine learning-based classification with causal relationship analysis, explainable prediction techniques, predictive uncertainty estimation, and risk-based alert prioritisation. Its central objective is to address the limitations of detection systems that provide a threat label without adequately explaining the prediction or indicating its reliability.

The proposed approach distinguishes four complementary tasks: identifying potentially malicious network activity, explaining the model's prediction, investigating whether relevant relationships have a defensible causal interpretation, and assessing uncertainty to determine whether additional review is necessary. SHAP and LIME can support the interpretation of individual predictions, while appropriate calibration and uncertainty estimation techniques can help identify potentially unreliable outputs. Causal analysis provides a separate mechanism for examining hypothesised relationships among network characteristics and threat outcomes, subject to the assumptions and limitations of the available evidence.

The framework also establishes an experimental approach for evaluating conventional classifiers and integrated models using detection performance, false-positive rate, calibration, explanation stability, robustness, and computational efficiency. However, the framework's practical advantages have not yet been established through experimental results in this manuscript. Claims concerning improved detection,

reduced false alarms, or greater robustness must therefore await evaluation using the selected datasets and reproducible experimental procedures.

Overall, the proposed framework provides a structured basis for investigating more transparent and uncertainty-aware cybersecurity systems. Its ultimate value will depend on whether experimental evidence demonstrates that the integrated approach supports reliable threat detection and meaningful security decisions under both familiar and changing network conditions.

8.1 Future Scope

Future work should implement the framework and evaluate it on UNSW-NB15 and CICIDS2017, with clear dataset partitions, baseline comparisons, and ablation experiments. Additional research should examine unseen attack categories, temporal distribution changes, explanation stability, causal sensitivity, and the trade-off between automated decision coverage and detection risk. Such experiments will provide the evidence needed to assess whether the framework is suitable for practical cybersecurity monitoring.

Reference:

- [1]. J. Pearl, *Causality: Models, Reasoning, and Inference*, 2nd ed. Cambridge, U.K.: Cambridge University Press, 2009.
- [2]. P. Spirtes, C. Glymour, and R. Scheines, *Causation, Prediction, and Search*, 2nd ed. Cambridge, MA, USA: MIT Press, 2000.
- [3]. J. Peters, D. Janzing, and B. Schölkopf, *Elements of Causal Inference: Foundations and Learning Algorithms*. Cambridge, MA, USA: MIT Press, 2017.
- [4]. M. A. Hernán and J. M. Robins, *Causal Inference: What If*. Boca Raton, FL, USA: Chapman & Hall/CRC, 2020.
- [5]. A. B. Arrieta et al., "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020, doi: 10.1016/j.inffus.2019.12.012.
- [6]. A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018, doi: 10.1109/ACCESS.2018.2870052.
- [7]. R. Guidotti et al., "A survey of methods for explaining black box models," *ACM Computing Surveys*, vol. 51, no. 5, Art. no. 93, 2018, doi: 10.1145/3236009.

- [8]. M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you? Explaining the predictions of any classifier," in Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2016, pp. 1135–1144, doi: 10.1145/2939672.2939778.
- [9]. S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems, vol. 30, 2017.
- [10]. [10] C. Molnar, Interpretable Machine Learning: A Guide for Making Black Box Models Explainable, 2nd ed., 2022. [Online]. Available: <https://christophm.github.io/interpretable-ml-book/>
- [11]. Z. Zhang, H. Al Hamadi, E. Damiani, C. Y. Yeun, and F. Taher, "Explainable artificial intelligence applications in cyber security: State-of-the-art in research," arXiv preprint arXiv:2208.14937, 2022. [Online]. Available: <https://arxiv.org/abs/2208.14937>
- [12]. A. S. R. Mohammed and H. Al-Hamadi, "Explainable artificial intelligence for cybersecurity: A literature survey," Annals of Telecommunications, vol. 77, pp. 789–812, 2022, doi: 10.1007/s12243-022-00926-7.
- [13]. M. Pawlicki et al., "The survey on the dual nature of xAI challenges in intrusion detection and their potential for AI innovation," Artificial Intelligence Review, vol. 57, Art. no. 330, 2024, doi: 10.1007/s10462-024-10972-3.
- [14]. A. Al, S. Sagioglu, "Explainable artificial intelligence models in intrusion detection systems," Engineering Applications of Artificial Intelligence, vol. 144, Art. no. 110145, 2025, doi: 10.1016/j.engappai.2025.110145.
- [15]. A. Mohale and T. Obagbuwa, "A systematic review on the integration of explainable artificial intelligence in intrusion detection systems to enhancing transparency and interpretability in cybersecurity," Frontiers in Artificial Intelligence, vol. 8, Art. no. 1526221, 2025, doi: 10.3389/frai.2025.1526221.
- [16]. G. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in Proc. 34th Int. Conf. Machine Learning (ICML), 2017, pp. 1321–1330.
- [17]. Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in Proc. 33rd Int. Conf. Machine Learning (ICML), 2016, pp. 1050–1059.
- [18]. C. Koneni and S. Lokula, "Evidence-Quality Telemetry for Cloud Incident Response: Detecting Gaps, Drift, and Integrity Failures in Large-Scale Observability Pipelines," 2026 14th International Symposium on Digital Forensics and Security (ISDFS), Boston, MA, USA, 2026, pp. 1–6, doi: 10.1109/ISDFS69419.2026.11459097.
- [19]. V. K. Bharadwaj Parasaram, V. Teja Bathini and S. K. Nalluri, "An Intelligent Digital Framework for Clinical Trial Monitoring Using AI and Cloud-Based Software Platforms," 2026 2nd International Conference on Artificial Intelligence Systems (AIS), San Antonio, TX, USA, 2026, pp. 524–530, doi: 10.1109/AIS69919.2026.11621841.
- [20]. Y. Ovdia et al., "Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift," in Advances in Neural Information Processing Systems, vol. 32, 2019.
- [21]. Koneni, C., S. Lokula, and L. Panjiyar. "A Cloud-Native Approach to Statistical Anomaly Detection and Automated Data Quality Validation." *International Journal of Data Science and IoT Management System* 4.4 (2025): 533-543
- [22]. A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" in Advances in Neural Information Processing Systems, vol. 30, 2017.
- [23]. C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in Proc. 34th Int. Conf. Machine Learning, 2017, pp. 1321–1330.
- [24]. Distributed Data Processing Frameworks for Large-Scale Healthcare Analytics. (2019). *International Journal of Advance Industrial Engineering*, 7(04), 1–10. doi:10.14741/ijaie/v.7.4.01
- [25]. A. N. Angelopoulos and S. Bates, "A gentle introduction to conformal prediction and distribution-free uncertainty quantification," Foundations and Trends in Machine Learning, vol. 16, nos. 4–5, pp. 494–591, 2023, doi: 10.1561/2200000101.
- [26]. M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A survey of network-based intrusion detection data sets," Computers & Security, vol. 86, pp. 147–167, 2019, doi: 10.1016/j.cose.2019.06.005.

- [27]. Vishram Singh, 2026. "Enterprise Resource Planning for Financial Applications: Engineering, Architecture, and Design Principles," *International Journal of Scientific Research in Science and Technology*, Technoscience Academy, vol. 13(2), pages 653-661, April.
- [28]. S. K. Nalluri, V. T. Bathini and V. K. B. Parasaram, "AI-Augmented Manufacturing Execution Systems for Precision Healthcare: A Scalable Framework for Smart Genomic and Proteomic Production," *2026 IEEE 2nd International Conference on Secure IoT, Assured and Trusted Computing (SATC)*, Houston, TX, USA, 2026, pp. 1-6, doi: 10.1109/SATC69565.2026.11542384.
- [29]. N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Military Communications and Information Systems Conference (MilCIS)*, 2015, pp. 1-6, doi: 10.1109/MilCIS.2015.7348942.
- [30]. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Information Systems Security and Privacy (ICISSP)*, 2018, pp. 108-116, doi: 10.5220/0006639801080116.
- [31]. M. Tavallaei, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Computational Intelligence for Security and Defense Applications (CISDA)*, 2009, pp. 1-6, doi: 10.1109/CISDA.2009.5356528.
- [32]. A. Vassilev, A. Oprea, A. Fordyce, H. Anderson, X. Davies, and M. Hamin, *Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations*. Gaithersburg, MD, USA: National Institute of Standards and Technology, NIST AI 100-2e2025, 2025. [Online]. Available: <https://csrc.nist.gov/pubs/ai/100/2/e2025/final>
- [33]. Vishram Singh, "Real-Time Financial Data Validation Platform for Enterprise Accounting Systems", *IJCISIM*, vol. 18, no. 10s, pp. 467-481, Jul. 2026.
- [34]. D. E. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 222-232, 1987, doi: 10.1109/TSE.1987.232894.
- [35]. A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153-1176, 2016, doi: 10.1109/COMST.2015.2494502.
- [36]. Sannidhanam, A. H. (2021). Real-Time Claims Processing Using Event-Driven Architectures. *International Journal of Technology, Management and Humanities*, 7(01), 36-50. doi:10.21590/07.01.02