Predictive Threat Intelligence Using Machine Learning

Thakor BhargaviBen Balvantsinh[1], Dr. Pratima Upadhyay[2]

[1]MCA Student, PP Savani University, Dhamdod, Surat, Gujarat, India

[2]Assistant Professor, PP Savani University, Surat, Gujarat, India, Email

DOI: https://doi.org/10.51584/IJRIAS.2026.11030095

Received: 04 April 2026; Accepted: 10 April 2026; Published: 16 April 2026

ABSTRACT

The number and sophistication of cyberattacks have dramatically expanded due to the quick development of digital technology, cloud computing, and Internet of Things (IoT) devices. Conventional intrusion detection systems are ineffective against zero-day and dynamic threats since they primarily rely on predetermined signatures and criteria. By examining both previous and current data, predictive threat intelligence (PTI) aims to foresee cyberattacks before they happen. Intelligent methods for extracting hidden patterns from massive network traffic are offered by machine learning (ML). Using the UNSW NB15 dataset, this study suggests a predictive threat intelligence system based on machine learning. A number of supervised learning methods are used and assessed, such as Random Forest, Artificial Neural Networks, Support Vector Machine, Logistic Regression, and EnXGBoost. According to experimental results, EnXGBoost performs the best, achieving an accuracy of 93.5% for multi-class classification and 98.7% for binary classification. In contemporary networks, the suggested approach facilitates real-time deployment, lowers false alarms, and improves proactive security.

Keywords: Predictive Threat Intelligence, Machine Learning, Intrusion Detection, UNSW NB15, EnXGBoost, Cybersecurity.

INTRODUCTION

The contemporary society utilizes a variety of Cloud solutions, digital communications and connected devices in order to maximise both access and effectiveness. However, these technologies can also pose serious threats to security. They are used more often than not, and do so with greater complexity than they have in the past. Malware, Ransomware, Phishing, DDoS (Distributed Denial of Service) and Data Loss are just a few examples of cyber attacks that continue to be more common and advanced with each passing day. Organizations are dealing with increasingly higher monetary losses and loss of image and reputation because of Cyber attacks. Some cybersecurity studies show that there are billions of dollars lost due to Security Breaches each year; therefore, there is an increasing urgent need to protect network structures.

Most of today's intrusion detection systems (IDS) rely predominantly on rule-based and signature- based methods for detecting and preventing attacks. An IDS examines the content of network traffic (Packets) to find any sequence that indicates that the packets were part of a known attack. While rule and signature based IDS have proven to be an effective means of defending against known attacks, they cannot detect new types of attacks, and they require a continuing and laborious effort to maintain an up-to-date signature database.

Predictive threat intelligence provides a way to anticipate prospective attacks and to go beyond the constraints of previous security methods by utilizing prior experiences, attack trends, or patterns of behavior to predict future attacks. Predictive threat intelligence (PTI) develops future attack predictions by employing machine learning to automatically examine large data sets to identify new, complex patterns in real-world data and to adapt quickly to changing attack strategies. Utilizing the UNSW-NB15 dataset as a source data set for this

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

research, we are proposing a PTI paradigm based on machine learning and the following primary objectives of this research include:

1. Designing an intelligent system for proactive threat detection.
2. Assessing different types of machine learning algorithm performance.
3. Evaluation of the precision of the author's methods of measuring intrusion detection into network based systems and networks.
4. Finding the best performing machine learning models for deployment in real-world settings.

LITERATURE REVIEW

Many previous studies have widely used the UNSW-NB15 dataset to build machine learning models for predicting cyber threats and to test how well intrusion detection systems perform in real -world network environments. Most previous studies comparing machine learning models used by traditional classification methods reported data- based results of their studies about comparison of LR, SVM, DT, RF, EnXGBoost and SVMs for IoT intrusion detection and classification (aka "intrusion"). Every ML/CNN (or combination thereof) based intrusion detection classification method that the authors tested had significantly higher success rates than traditional ML/CNN- based models when classifying malware based on the number of dimensions in the network traffic. The results of More et al. [2] demonstrated the value of using specific feature generators to analyze and select relevant features for building robust and repeatable machine learning models. The authors demonstrated a statistically significant increase in the performance of their feature extractor process when evaluating the Random Forest model compared to the other baseline ML-based intrusion detection models evaluated. Al-Obaidi et al. [3] evaluated nine attack types documented in the UNSW-NB15 dataset and found that ensemble and boosting- based approaches provided improved precision and recall rates than traditional classifiers for all nine attack types.

Previous studies have used several techniques for preprocessing data to minimize the redundancy associated with characteristic variables and rectify imbalanced datasets. For example, in [4], the researchers demonstrated that resampling methods can be helpful in enhancing the detectability of small-sized classes/instances; furthermore, the researchers compared many different types of machine learning algorithms on imbalanced datasets. Through utilising a multi -layer perceptron (MLP) in addition to using a hybrid feature selection technique called information gain ratio (IGR) with recursive feature eliminations (RFE), Yin and colleagues [5] were able to achieve significant levels of classification accuracy, as well as much lower levels of computational complexity than would otherwise have been the case had they only utilized their approach without the use of IGR.

Talukder and colleagues [6] discovered that improvements in algorithm performance on both very large-scale and unbalanced datasets can be attained by increasing generalization capability when employing oversampling methods and stack embedding methods for feature embedding.

Ensemble learning techniques such as stacking have been used in supplementary studies as a reference for improving prediction accuracy. For example, Karthi et al. [7] used a stack of classifiers that combined random forests (RF), support vector machine (SVM), and logistic regression as an ensemble classifier that performed well against multiple types of intrusion. Dasari et al. [8] created another stacked ensemble learning technique called NIDSSELT (Network Intrusion Detection System Stacked Ensemble Learning Technique) model that used extra trees (ET) and extreme gradient boosting (EnXGBoost) classifiers to provide improved detection stability while overcoming the problems associated with class imbalances. All techniques had proven success rates for general izing to complicated heterogeneous traffic conditions.

Hybrid architectures and deep learning techniques have become very popular for the purpose of generating predictive threat intelligence. As an example of this, Dangol [9] demonstrated that CNN- BLSTM models linked with resampling techniques such as ADASYN could provide improved results when classifying attacks. Additionally, Luqman et al. [10] produced a parameter -based Intelligent Intrusion Detection System within the context of the Internet of Things (IoT), incorporating both Random Forestand Support Vector Machine classifiers in conjunction with sophisticated pre- processing approaches. Furthermore, in the context of hybrid

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

architectures, Biradar et al. [11] acknowledged that Attention-AUGMENTED Graph Neural Networks (GNN) integrated with Recurrent Neural Networks (RNN) had superior capabilities to identify spatial and temporal dependencies within the flow of data through networks.

Also worthy of mentioning is the increasing focus in the field of meta-learning and adaptive intrusion detection systems. For example, Alrayes et al. [12] established a model-agnostic meta-learning (MAML) framework for improving the adaptability of intrusion detection systems to the changing behaviour of malicious software within the IoT. Evidence supporting the legitimacy of ensemble-based methodologies in predictive threat modelling was established by Putra [13] and Putro [14] by evaluating multiple classical classifiers on the UNSW-NB15 dataset. Finally, Swaroop et al. [15] researched the application of case-based reasoning through the use of support vector machines and random forests in order to enhance the interpretability as well as the performance of anomaly detection.[16] The dataset employed for this project is called UNSW-NB15 and represents a large-scale, state- of-the-art benchmark dataset developed specifically for researching intrusion detection in networks. This dataset was designed to provide researchers with both real examples of 'normal' network behaviour as well as examples of many different types of cyber attacks that can reasonably simulate situations that might be observed in a real-world network. The IXIA PerfectStorm tool was used to create this dataset, and it has been developed to provide a modern -day representation of today's network environments as well as today's attack methods, making it ideal for assessing machine learning-based models used to detect intrusion attempts.

DATASET DESCRIPTION

The UNSW-NB15 dataset also provides support for both binary and multi -class classification and contains over 2.5 million records that contain 49 different extracted feature attributes. The attack taxonomy consists of nine primary categories: Fuzzers, Analysis, Backdoor, DoS (Denial-of- Service), Exploits, Generic, Reconnaissance, Shellcode and Worms, and provides a complete overview of the methods employed by modern intruders.

In order to capture the unique characteristics of networks, the features extracted from the UNSW- NB15 dataset have been organized in accordance to numerous feature groups. Basic feature groups are representative of the fundamental connection characteristics used to describe a connection between two network devices (e.g., protocol type, connection duration, and source/destination port numbers). Content feature groups describe the characteristics of the data contained during a connection (e.g., payload-related and error-related characteristics extracted from packet inspection). Time feature groups describe the temporal statistics of the connection traffic over specific time intervals, providing for analysis of the temporal behavior (i.e., patterns) of connections. Flow feature groups describe the packet count, byte count and flow rates used to create connection flows between two devices and, thus are important for detecting abnormal and/or excessive network traffic.

The dataset's robustness provides a wealth of information, but there are challenges that must be met before developing a model. One major challenge is the imbalance between the number of observations in each of the different types of attack categories; thus, learning algorithms may have a tendency to be biased toward the majority. Furthermore, the dataset includes redundant and correlated attributes and are hindered by many noisy observations found within the dataset as well as uneven distributions of attacks throughout. All of these factors can impede the likelihood of performance when done correctly Classification performance when done correctly.

We have created an organized process for degree processing which has decreased issues with determining which of the features in raw network traffic were actually useful for ML to work. By converting raw network traffic into structured numerical attributes (i.e., packet size, connection time, and flow rate) at the time of feature extraction we were able to make ML function correctly. Unnecessary attributes were removed to allow the model to have a simpler and more understandable output through methods that included correlation analysis and mutual information as feature selection methods. A dimensionality reduction method, such as PCA, was used to help decrease the dimensionality of the features while still accounting for most of the variance in the data. We also generated synthetic samples for the minority classes using SMOTE to help perform a better classification by having a balanced class representation.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

System Architecture

System Architecture Suggestion

The design of the proposed predictive threat intelligence system employs a structured layered architecture which is showing in figure 1- provides modularity, scale, and the ability to effectively detect threats. Each layer is responsible for specific phases of the data processing and predictive pipeline. By organizing the architecture into a number of interconnected layers (or data processing and predictive pipeline phases), the layered architecture aids in improving visibility, maintainability, and real-time responsiveness of systems in a constantly changing network environment.

The Data Collection Layer (also called the "Data Collection Layer" in the architecture) collects unprocessed network traffic data from various sources: intrusion detection systems, routers, firewalls, and network monitoring facilities. The Data Collection Layer (DCL) is continuously collecting both packet level data (or object layer) and flow-level data (or communication layer) to facilitate total visibility of network activity. The collected network traffic data are the primary input for the data analysis layer.

To improve the quality of data and to prepare it for use with machine learning algorithms, the Preprocessing Layer will carry out some key data preparation procedures. The procedures carried out in this stage will involve cleaning the data to remove any records that are either missing or inconsistent, normalising the numerical attributes to place them all within a common range of values, and encoding (or converting) the categorical characteristics into a machine readable format.

Proper preprocessing of the data will allow for a reduction in noise from these datasets, which, in turn, will increase the likelihood of the model converging during the training phase. Once preprocessing is complete, the Feature Engineering Layer will extract, select, and transform the relevant attributes from the original raw traffic data. The aim of this stage is to derive features that make traffic behaviour patterns more easily identifiable. Redundant and irrelevant attributes will be eliminated, and transformed features will be produced, thus enhancing the ability of these features to provide accurate forecasts. Effective feature engineering will improve the representation of the data and dramatically increase the performance and generalisation of the model.

The dataset will be sent to the Model Training Layer, where it will be used to train the supervised machine learning algorithms using labelled data. This layer will involve parameterisation, selection of the model and validation of the selected model. The model that will perform best in providing the ability to differentiate between legitimate/benign and malicious/threatening network activity will need to be adjusted and validated. In addition, the model will learn to identify the dispersed pattern characteristics of the various types of threat, by training the model on the use of previous attack patterns.

The system enters the final Prediction Layer after successfully being trained and/or having previous knowledge of inputs being made on it. It analyzes all incoming real-time traffic on the network and determines if it is legitimate or malicious (to continue to build a model). Once this analysis is complete, it then utilizes its previously trained data models to evaluate the incoming traffic streams and access the features extracted from live network streams and perform predictions with minimal delays (so that proactive threat detection can happen).

The Alert Generation Layer is the third level of this architecture and is responsible for detecting malicious traffic in incoming streaming traffic. After an alert has been generated by the Alert Generation Layer, the administrator (or end- user) will receive a copy of the alert. The information that is passed along with the alert gives the administrator useful information such as what type of attack has occurred, how severe the attack is, where the attack originated, and when the attack occurred. This allows the administrator to act quickly to address the security issue.

Using the two layers of development creates a more robust system due to the structured integration of data acquisition, pre-processing, feature optimization, model training and prediction/alert generating processes/methodologies. The overall predictive threat intelligence model also utilizes a feature engineering component within the model that assists to increase the overall accuracy of predictions made by models by

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

reducing the dimensionality of data and improving the overall quality of the data used for prediction purposes.

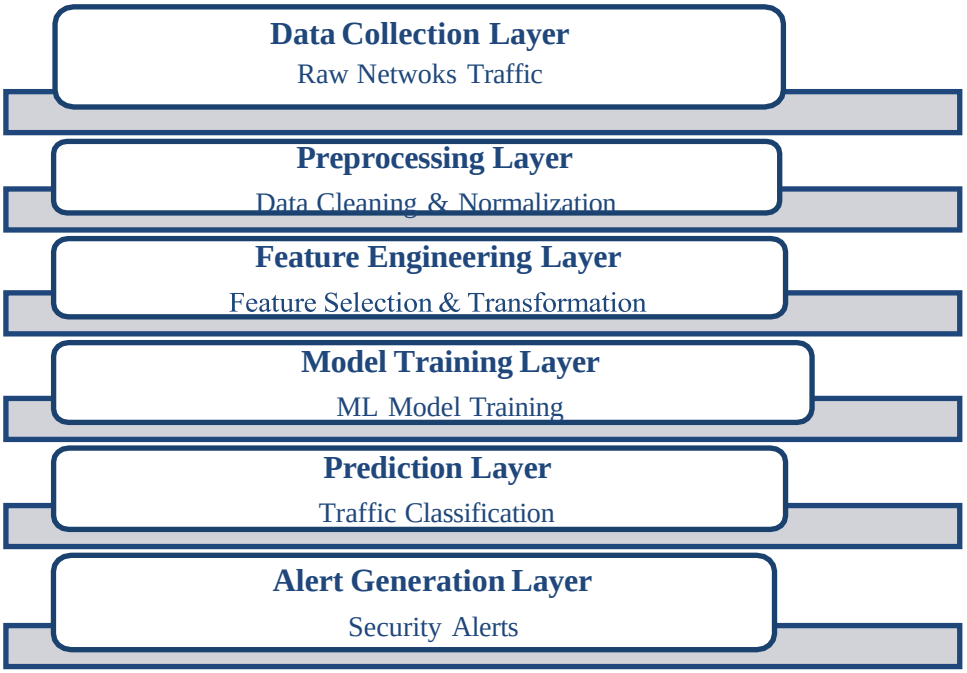

Figure 1: Predictive Threat Intelligence System Architecture

EXPERIMENTAL METHODOLOGY

Before the development of the model, an extensive preprocessing procedure is employed to ensure that the dataset is consistent, reliable and appropriate for supervised learning. Systematic identification and exclusion of records containing invalid, inconsistent or ambiguous values will prevent learning from being distorted. Categorical (protocol category, service type, etc.) features will be converted to numeric form using appropriate encoding techniques; this will allow them to be effectively integrated with the machine learning algorithms.

Standardizing the continuous variables to a common scale will help maintain numerical stability and improve the efficiency of the optimization process through the use of statistical normalisation techniques that allow every feature to have a mean of zero and a variance of one. This will allow for faster convergence in model training and reduce the effect of high magnitude variables. In addition, statistical analyses will be performed to identify and remove the effect of extreme or outlying observations that may negatively affect classifier accuracy. The combination of these preprocessing procedures will improve the quality of the feature representations and increase the generalisation ability of the predictive models. A thorough analysis will help to compare different supervised learning techniques. When choosing which algorithms to analyze, we selected 5 algorithms for this project. We chose Logistic Regression based on its efficiency and its ability to produce interpretable models.

Support Vector Machine is included in our analysis because of its capability to handle high- dimensional feature spaces and produce high- quality decision boundaries. Use of Random Forest will give us an ensemble based model that can produce robust predictions while reducing the likelihood of overfitting. An Artificial Neural Network will allow us to model complex and non- linear data relationships as they relate to network traffic data. Finally, we will use EnXGBoost as a scalable, regularized, and highly effective multi-class classification algorithm. Including these 5 algorithms in our analysis provides us with the vantage point to evaluate statistical, ensemble, and deep learning algorithms when making predictions related to threat intelligence. Using a 70% - 30% split of the data into training and testing data we will use the testing data only to validate how well a trained model performs on unseen data, while the training data is used to train models and adjust their parameters. To help reduce the amount of variance in the evaluation process we will utilize k-Fold Cross-Validation during the training process. Each data fold will contribute to both the training and validation processes. This iterative process allows for more robust model training and leads to reducing the chance of variance in the outcome of the evaluation from the data used for training both the model.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

Categorization has its standard performance measures, which can be used as an objective way of measuring the effectiveness of a model. Accuracy is defined Precision is defined as how many of the total attacks are identified correctly. Combining precision and recall into a single value (the harmonic mean) produces the F1 score, which is a balanced measure of precision and recall. This gives a useful measure of the model for moments when there is an imbalance in the categories being evaluated. Another evaluation tool is ROC-AUC, which measures how well the model is able to distinguish between multiple threshold values. by its ability to make predictions correctly. Recall is used to determine how many of the actual harmful events were detected. FAR is another important measure of performance that can be calculated as well; this measure is used to quantify how frequently false alerts are produced. This is crucial in an operational cybersecurity environment, where high numbers of/proportion of false alerts may cause alert fatigue. The overall design of the proposed experimental framework incorporates extensive preprocessing, thorough evaluation of the model, and a complete assessment of the performance of the threat intelligence system so that the resulting predictive threat intelligence system is as reliable and effective as possible.

Hyperparameter Optimization

Hyperparameter optimization is performed in order to enhance the model's accuracy, robustness , and ability to generalize as well as develop the "best" performance of the model. Hyperparameter tuning also reduces potential overfitting, regulates the overall complexity of the model and ensures that all important patterns within the data were successfully captured . Systematic search techniques are implemented to obtain optimal parameter settings , this can be done using "Grid Search" which entails evaluating all possible major hyperparameter values that you have defined via extensive processes , while "Random Search" randomly samples major hyperparameter combinations from the defined range of possible combinations thus providing a more computationally efficient way of finding optimal parameter settings although still being competitive in terms of model performance . Cross-validation processes will be embedded with both search techniques to achieve reliability and minimize bias in evaluating candidate parameter combinations via means of evaluating each candidate parameter configuration across multiple datasets (folds) for stable and unbiased estimations of the model's overall performance.

Out of all of the models attempted, Extreme Gradient Boosting (EnXGBoost) poses the greatest challenge in requiring hyperparameter tuning because EnXGBoost has great sensitivity to the configuration settings. There are several parameters that were optimized through this study which included: max_depth - the maximum depth of each of the individual decision trees (max_depth) has a significant impact on overall model complexity. Also , the learning_rate - which dictates the amount of weight the model assigns to each boosting step which has an impact on how quickly the model converges. Finally n_estimators - is the total number of boosting trees in the ensemble of decision trees.

The proposed predictive threat intelligence system is implemented within a structured software and hardware environment to ensure efficient development and evaluation. The system is developed using the Python programming language because of its robust support for data science applications, wide ecosystem, and adaptability. Several open-source libraries are employed to facilitate different stages of the implementation process. Scikit- learn is used for implementing conventional machine learning algorithms and evaluation metric ,while TensorFlow facilitates the creation and training of models for artificial neural networks. In order to classify data using the boost of gradient boosting, the EnXGBoost library is utilized. Numerical calculation, data preprocessing, and manipulation will utilize both NumPy and Pandas.

The experiments will be done using a standard machine learning experimentation environment where there are enough resources in terms hardware configuration to support training and evaluating models via machine learning. The system has an Intel Core i7 CPU that provides sufficient computing power to train and test a model, and has been fitted with 16 GB of random access memory so that it can efficiently handle large-scale data, like UNSW-NB15, with ease.

To provide quick access to data and minimize input/output times while processing data, a 512GB Solid- State Drive (SSD) is included in this system. The system has the capability of utilizinga Graphics Processing Unit (GPU) at the user's request to provide additional computational support for accelerating the training of Neural

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

Networks, particularly models. Therefore, deep learning-type the combination of systematic hyperparameter optimization and a properly configured implementation environment result in stable and repeatable performance for high-performance predictive modeling of Cybersecurity Threat Detection.

RESULT

The following are results of the binary classification based upon four metrics: accuracy and precision, recall, and F1 score, which includes both precision and recall. The figure 2 displays the performance comparison of each model against these metric comparisons in a visual manner.

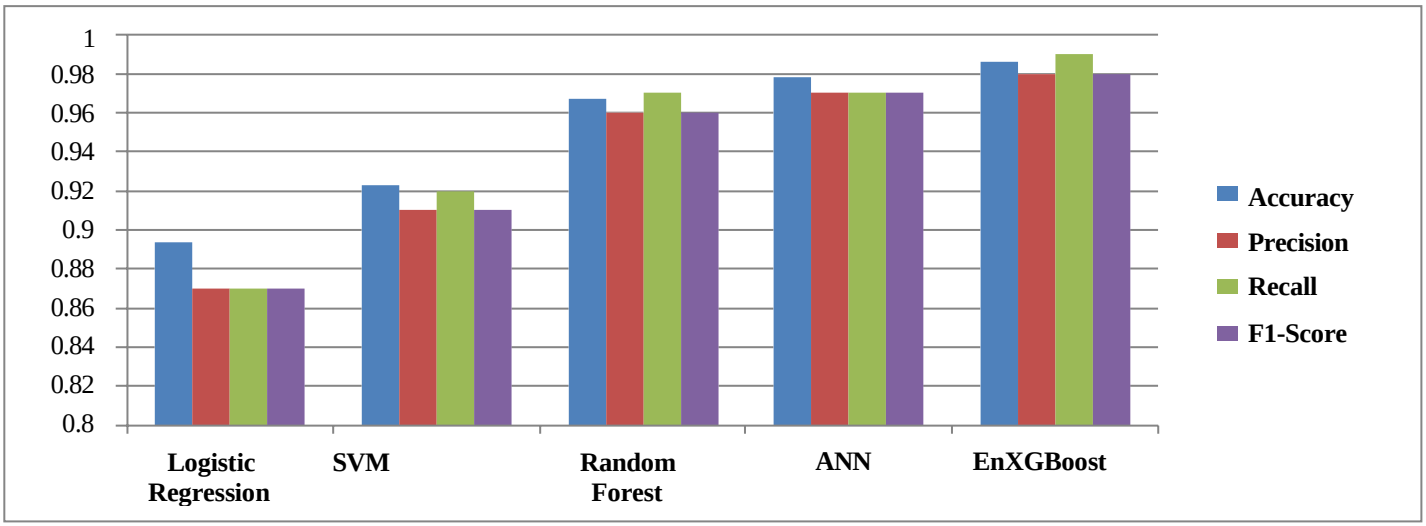

Figure 2: Performance comparison of the Proposed Model

From examination of the models' results and their visualizations, several conclusions can be made. The following are highlights of those conclusions:

Performance Leader (EnXGBoost): The EnXGBoost classifier provides the highest level of performance across all metrics with an accuracy of 98.7% and 0.99 recall. This indicates that EnXGBoost is very capable of accurately classifying all instances that are classified as positive.

Deep Learning Performance (ANN): The ANN provided very similar performance to EnXGBoost with an accuracy of 97.5%; however, the results were balanced among all metrics with precision, recall, and F1 score all being equal to 97%.

Ensemble Strength: Random Forest and EnXGBoost (both ensemble methods) produce results far superior to those produced by logistic regression and SVM (both linear methods). In addition, as model complexity increases, there is a significant increase in performance.

Performance Consistency: For each model, precision, recall, and F1 scores are very similar. This would indicate that the dataset is probably fairly balanced and that none of the models developed a bias toward one of the classes.

The multi-class classification algorithm comparisons can be found in the figure 3 below. The range of the y-axis has been adjusted so that we can see the gap between the performances of the three algorithms, which is where the height of the bars is based strictly on the relative accuracy of each algorithm. In comparison to the two algorithms for binary classification, EnXGBoost also outperformed the other two algorithms in the multi-class classification problem, with an impressive 93.5% accuracy rate at the end of the data set. The EnXGBoost classification algorithm appears to perform very well with multi class classification problems that have more complicated decision boundaries (multi -class classification). The ANN classification algorithm performed very well with 91.2% accuracy, indicating that deep learning architectures (ANN) are very effective classification methods for both two- dimensional and three-dimensional data. Solid Benchmark: Random

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

Forest also demonstrated to perform very well with an 88.4% accuracy rating at the end of multi-class classification data set; while Random Forest was slightly lower in overall accuracy than the other two algorithms, it is still an established, reliable model for multi-class classification problems. Random Forest would have a more difficult time separating the classes, as compared to gradient boosting classes.

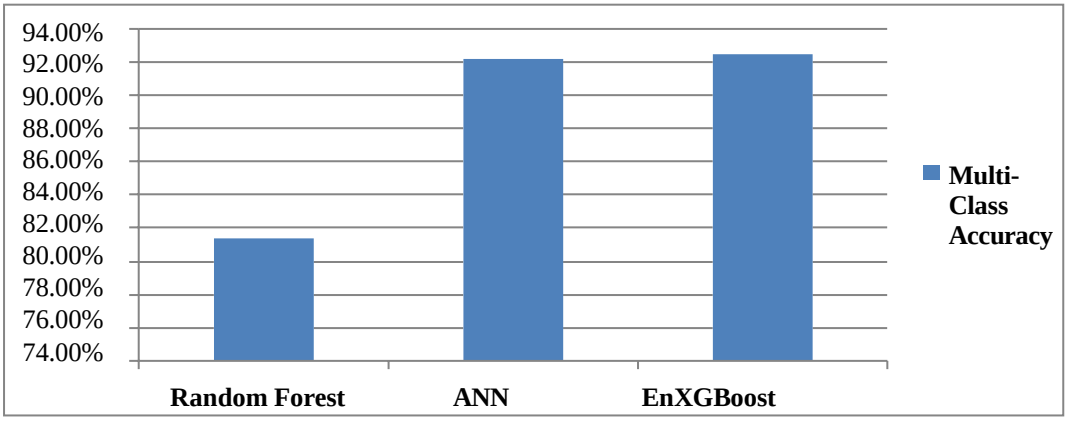

Figure 3: The Multi-Class Classification of Proposed Model

The data collected in this study indicates that EnXGBoost is the best -performing machine-learning model when compared to other models tested on this dataset to detect multiple classes of threats.

CONCLUSION

The predictive threat intelligence architecture created using machine learning techniques outlined in this paper develops proactive cybersecurity systems from the UNSW-NB15 dataset. The architecture comprises a complete methodology consisting of feature engineering, supervised training of models, preparation of data, and full evaluation of performance. During the analysis phase, we tested multiple classification techniques including EnXGBoost, Random Forest, Support Vector Machine, LogisticRegression, and Artificial Neural Network. The analysis showed that EnXGBoost was the most successful method for classifying data into two or more classes. In addition to demonstrating how ensemble and boosting methods perform well for intrusion detection (as evidenced by the previous finding), EnXGBoost has good memory efficiency (potentially reducing overall processing costs), good accuracy and good precision; thus, optimized machine learning models were capable of improving the predictive power of our threat assessment significantly. The design of the framework will help reduce false positive detection rates, improve operational security effectiveness by raising early detection rates (and thus enabling a faster response to an incident), and improve the ability to mitigate the impacts of an incident. The system has been designed with flexibility and scalability to support heterogeneous networks. It is suitable for enterprise networks, cloud platforms, and infrastructures. IoT: The modular architecture enhances reliability and maintainability. In general, the study highlights how crucial data- driven strategies are to cybersecurity. The findings are in favour of incorporating cognitive analytics into contemporary military systems. The suggested approach offers a workable and effective way to detect cyber threats in advance. It helps to fortify the next-generation cyber defence system.

FUTURE WORK

Future research can enhance the proposed predictive threat intelligence system by combining cutting- edge deep learning methods with new security innovations. By incorporating both spatial and temporal traffic patterns, CNN– LSTM models may increase detection accuracy. It is possible to use Explainable Artificial Intelligence (XAI) to improve model transparency and support better decision-making by security analysts. Federated learning offers a privacy preserving framework for collaborative threat detection across organizations. Blockchain-based platforms can enable secure and tamper- resistant threat information sharing. Graph Neural Networks (GNNs) may improve detection of coordinated and multi stage attacks through Page 15 of 16 - Integrity Sustructural modeling of network entities. Real- time large scale deployment should be conducted to evaluate system scalability and robustness Testing how an application performs under high user loads will help to guarantee that it will continue to operate reliably in the future. Also, as attackers continue to develop new

ways to attack organizations' networks, continually modifying to match new attack methods is necessary. Collectively, these changes will improve the scalability, transparency, and resiliency of predictive threat intelligence systems.

REFERENCES

1. Int. J. Adv. Sci. Res. Eng., 2024, N. G. Anoh et al., "IoT Intrusion Detection System based on Machine Learning Algorithms utilizing the UNSW- NB15 dataset."
2. S. More et al., "UnSW-NB15 Data Analysis Improves Intrusion Detection Performance," Algorithms, 2024. System
3. A. Al-Obaidi et al., "Effectiveness of ML Techniques to Detect Nine Attacks using UNSW NB15," MMEP, 2023.
4. "Intrusion Detection: A Comparison Study of Machine Learning Models Using Unbalanced Dataset," SN Comput. Sci., 2024.
5. "IGRF-RFE: Hybrid Feature Selection for MLP- based IDS on UNSW-NB15," by Y. Yin et al. Big Data Journal, 2023.
6. ML-based network intrusion detection for large and unbalanced data using oversampling and stacking, M. A. Talukder et al., 2024.
7. M. Karthi and colleagues, "The UNSW-NB15 Dataset: Network Intrusion Detection Using Stacked Machine Learning Models," 2025.
8. UNSW-NB15 Dataset," 2025. [8] A. K. Dasari et al., "Stacked Ensemble Learning Technique for NIDS," Eng. Res. Express, 2025.
9. "Comparative Analysis of NSL-KDD and UNSW-NB15 Using Deep Learning Resampling Methods," N. Dangol, 2025.
10. M. Luqman et al., "Intelligent Parameter- Based In-Network IDS for IoT utilizing UNSW NB15," Journal of Franklin Institute, 2025.
11. J. Biradar and colleagues, "Attention Augme
12. F. S. Alrayes et al., "Adaptive IDS using Model-Agnostic Meta-Learning on UNSW NB15," Sensors, 2025
13. Z. P. Putra, "Evaluating Classification Algorithms on the UNSW-NB15 Dataset," 2024.
14. I. H. Putro, "Performance Evaluation of ML Classifiers for Network Intrusion Detection," 2025.
15. T. P. J. Swaroop et al., "Enhanced IDSSVM and Random Forest applied to the UNSW- NB15 dataset, 2025.
16. S. Fatima, P. Upadhyay, P. Sharma, S. K K. Khadagade, S. Bihari and R . Tiwari, "Hybrid Machine Learning Framework for Intelligent Attack Detection in VANET Environments "2025 International Conference on Data, Energy and Communication Networks (DECoN), Bhopal, India,2025.