International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2024: 7.101

Enhanced Cyber Incident Detection and Prediction Using Machine Learning and SMOTE-Based Class Balancing

Dr. S. Gnanamurthy[1], M. Arun Kumar Naik[2]

[1]Associate Professor, HOD, Departmwnt of CSE (Data Science), Department of Computer Science and Engineering, Kuppam Engineering College, KES Nagar, Kuppam, Andhra Pradesh 51742, India

[2]PG Scholoar, Department of Computer Science and Engineering, Kuppam Engineering College, KES Nagar, Kuppam, Andhra Pradesh 51742, India

Abstract: ***Cybersecurity incidents continue to escalate in complexity and volume, necessitating advanced detection systems. This study investigates the application of machine learning models-specifically Decision Trees, Logistic Regression, Random Forest, Gradient Boosting, XGBoost, and LightGBM-to detect cyber threats. A dataset of 9.5 million entries with 45 features was used. Initial evaluation revealed XGBoost achieved the highest macro F1-score (0.91). To address class imbalance, the Synthetic Minority Over-sampling Technique (SMOTE) was applied, improving XGBoost's performance to 0.90 accuracy, 0.92 precision, recall, and F1-score. The findings support the integration of ensemble learning and data balancing for robust and scalable threat prediction. This research underscores the effectiveness of ensemble learning techniques, particularly XGBoost, in predicting and mitigating cyber threats, offering a scalable and efficient approach to cybersecurity incident detection.***

Keywords: Cybersecurity, Machine Learning, XGBoost, SMOTE, Threat Detection *etc.*

1. Introduction

Cybersecurity has become a critical concern in today's digital era, with cyber threats evolving in complexity and scale. The rise in cyberattacks, including malware infections, ransomware, phishing, and distributed denial-of-service (DDoS) attacks, poses a significant risk to individuals, organizations, and governments worldwide [1], [2]. Recent studies indicate that cyber incidents have surged in frequency, with financial and reputational damages exceeding billions of dollars annually [4], [5]. Consequently, robust mechanisms for detecting and predicting cyber incidents are necessary to mitigate risks and enhance security measures.

Machine learning (ML) has emerged as a powerful tool for cybersecurity applications, offering automated threat detection, anomaly identification, and proactive response capabilities [8], [9]. Traditional cybersecurity methods, such as rule-based intrusion detection systems (IDS) and signature-based malware detection, have limitations in identifying novel and sophisticated cyberattacks [13]. To address these gaps, researchers have explored various machine learning models, including decision trees, logistic regression, random forests, gradient boosting, and deep learning-based techniques [10], [14].

Despite the advancements in machine learning-driven cybersecurity, several challenges remain:

- *Class Imbalance in Cyber Incident Datasets:* Many cybersecurity datasets exhibit a severe imbalance, where normal instances significantly outnumber attack cases. This imbalance negatively impacts model performance, leading to biased predictions toward the majority class [9], [11].
- *Feature Selection and Model Optimization:* Identifying relevant features from high-dimensional cybersecurity datasets remains a challenge. Many studies use feature engineering techniques, but improper feature selection can lead to suboptimal model performance [12].
- *Generalization to Real-World Threats:* Most machine learning models are trained on specific datasets and may not generalize well to emerging cyber threats. Adversarial attacks further reduce the robustness of these models [16], [17].
- *Computational Complexity:* Advanced machine learning techniques, particularly deep learning models, often require substantial computational resources, making real-time cyber threat detection challenging [19].

Figure 1: Common Types of Cyber Incident

Given these challenges, the common types of cyber incidents are shown in Fig. 1., this research aims to enhance cyber incident detection and prediction using an improved machine learning framework. The study is motivated by the need for:

- More accurate and balanced cyber incident classification by addressing data imbalance using Synthetic Minority Over-sampling Technique (SMOTE).
- Optimized feature selection and model tuning to improve cybersecurity detection efficiency.

- Robust and scalable machine learning models that generalize effectively across different cyber threat scenarios.

The primary objectives of this study are:
- To evaluate various machine learning models for cybersecurity incident detection, including Decision Trees, Logistic Regression, Random Forest, Gradient Boosting, XGBoost, and LightGBM.
- To enhance model performance by addressing class imbalance using SMOTE.
- To compare pre-SMOTE and post-SMOTE model performance in terms of accuracy, precision, recall, and F1-score.
- To establish XGBoost as an effective model for cyber threat prediction and mitigation.

This paper makes the following key contributions:
- The study evaluates baseline and advanced ML models for cybersecurity incident detection.
- The application of SMOTE to improve class balance, leading to enhanced model performance.
- A detailed performance analysis demonstrating the superiority of XGBoost in cyber threat prediction.
- Insights into feature selection and model optimization for improved cybersecurity frameworks.

The rest of the paper is organized as follows: Section 2 provides a literature review on cybersecurity threat detection, incident response mechanisms, and machine learning applications. Section 3 describes the dataset, preprocessing techniques. Section 4 describes the dataset, preprocessing techniques, and machine learning models used in this study. Section 5 presents experimental results and performance comparisons.Section 6 concludes the paper with future research directions
.

2. Literature Survey

Cybersecurity remains a critical field of research, evolving to counter increasingly sophisticated cyber threats. Various studies have explored cyber-attack detection, prediction, and prevention mechanisms, emphasizing the role of machine learning in strengthening cybersecurity frameworks.

1) Cybersecurity Threat Landscape and Trends
Li and Liu [1] provide a comprehensive review of cyber-attacks and security trends, highlighting emerging threats and recent advancements in cybersecurity measures. Similarly, Kaur and Ramkumar [2] discuss the latest developments in the field, offering insights into new challenges organizations face. The Cybersecurity Almanac [4] and Datareportal's Digital 2022 Global Statshot [5] provide statistical insights into cyber incidents, emphasizing the growing frequency and severity of attacks.

2) Incident Response and Cyber Threat Management
Hejase et al. [3] and Hodgson et al. [7] explore the response mechanisms for significant cyber incidents (SCIs), comparing event life cycles in cyber and non-cyber domains. They highlight the importance of an adaptive incident response strategy to mitigate damages. The CSIS Significant Cyber Incidents (SCI) database [11] provides a historical perspective on major cyberattacks, demonstrating trends in threat evolution.

3) Machine Learning in Cybersecurity
The integration of machine learning for cyber threat detection and prevention has gained prominence in recent years. Handa et al. [8] provide an extensive review of machine learning applications in cybersecurity, detailing techniques applied to strengthen threat detection mechanisms. Ibor et al. [9] discuss various cybersecurity approaches, including attack detection and prediction. Dar et al. [10] compare machine learning models for cyber threat detection, evaluating their performance across multiple datasets.

4) Intrusion and Malware Detection Using AI
Various studies have explored different machine learning models for detecting cyber intrusions and malware. Wressnegger et al. [13] analyze n-gram-based anomaly detection methods, while Pektaş et al. [12] propose an n-gram-based algorithm for malware classification. Alqahtani et al. [15] and Terai et al. [16] explore intrusion detection systems using machine learning classification techniques and support vector machines (SVMs). Similarly, Ghanem et al. [17] investigate the use of SVMs for cyber-attack detection in network security.

5) Cybersecurity Applications in Critical Infrastructure
Machine learning has also been applied to secure industrial control systems (ICS) and power grids. Bhusal et al. [18] investigate cyber-attack detection on voltage regulation in distribution systems, emphasizing the role of AI-driven security solutions.

6) Network Security and Botnet Detection
Bapat [19] focuses on detecting malicious botnet traffic using logistic regression, demonstrating the potential of lightweight ML models for real-time threat detection. Kajal and Sardana [20] highlight the effectiveness of intrusion detection systems (IDS) using the Random Forest classifier to mitigate cyber-attacks.

The reviewed literature underscores the critical role of machine learning in modern cybersecurity. From intrusion detection to malware classification and incident response, AI-driven approaches continue to enhance cyber threat mitigation strategies. Future research should focus on improving model robustness against adversarial attacks and optimizing computational efficiency for real-time threat detection.

3. Dataset

The dataset used in this study consists of over 9.5 million records with 45 features related to threat identification, device profiling, and response actions.. It includes key identifiers such as IncidentId, AlertId, and Timestamp to track events, while AlertTitle and Category classify incidents based on their nature. Critical attributes such as MitreTechniques and IncidentGrade provide insight into the severity and tactics associated with different threats. Additionally, network and device-related fields, including IpAddress, DeviceId, OSFamily, and OSVersion, help in

Paper ID: SR251024103823 DOI: https://dx.doi.org/10.21275/SR251024103823

profiling affected systems. User and account information, such as AccountUpn, AccountSid, and AccountObjectId, enable the identification of compromised or targeted entities.

One of the primary challenges in this dataset is the presence of missing values in several features. For instance, MitreTechniques has 57.44% missing values, while ActionGrouped and ActionGranular are missing in over 99.4% of the records, making them difficult to utilize. Similarly, attributes like ThreatFamily, ResourceType, and AntispamDirection exhibit extreme sparsity, limiting their relevance for predictive modeling. However, certain essential fields, such as IncidentGrade, have relatively low missing percentages (0.54%), making them viable for imputation.

To address these challenges, a structured data preprocessing approach will be applied, including feature selection, missing value imputation, and categorical encoding. Features with excessive missing data (>90%) may be excluded to prevent bias, while critical attributes will be imputed using statistical techniques. Given the likelihood of class imbalance in cybersecurity threats, SMOTE (Synthetic Minority Over-sampling Technique) will be used to enhance model robustness by balancing class distributions. This refined dataset will then be leveraged for machine learning models, particularly XGBoost, which has demonstrated high effectiveness in cybersecurity threat detection.

1	OrgId	int64
2	IncidentId	int64
3	AlertId	int64
4	Timestamp	object
5	DetectorId	int64
6	AlertTitle	int64
7	Category	object
8	MitreTechniques	object
9	IncidentGrade	object
10	ActionGrouped	object
11	ActionGranular	object
12	EntityType	object
13	EvidenceRole	object
14	DeviceId	int64
15	Sha256	int64
16	IpAddress	int64
17	Url	int64
18	AccountSid	int64
19	AccountUpn	int64
20	AccountObjectId	int64
21	AccountName	int64
22	DeviceName	int64
23	NetworkMessageId	int64
24	EmailClusterId	float64
25	RegistryKey	int64
26	RegistryValueName	int64
27	RegistryValueData	int64
28	ApplicationId	int64
29	ApplicationName	int64
30	OAuthApplicationId	int64
31	ThreatFamily	object
32	FileName	int64
33	FolderPath	int64
34	ResourceIdName	int64
35	ResourceType	object
36	Roles	object
37	OSFamily	int64
38	OSVersion	int64
39	AntispamDirection	object

Figure 2: Dataset

This stacked bar chart shown in fig.3, depicts the hourly distribution of cybersecurity incidents, categorized as Benign Positives, False Positives, and True Positives, revealing that overall incident counts, particularly true malicious incidents, tend to peak during afternoon and evening hours, while false positives remain relatively consistent throughout the day, suggesting potential patterns in attack timing and the need for optimized detection strategies.

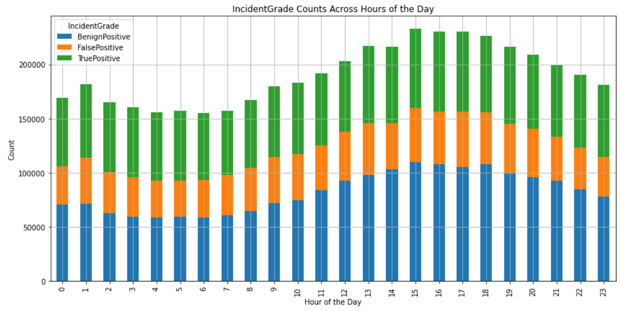

Figure 3: Incident grade counts across hours of the day

In Fig. 4. Incident grade counts across hours of the month extends the daily view to a monthly timescale, pinpointing specific days or weeks with heightened incident activity. It helps identify cyclical attack patterns or periods where increased security vigilance is crucial. You might see specific weeks or days where attacks are higher than others.

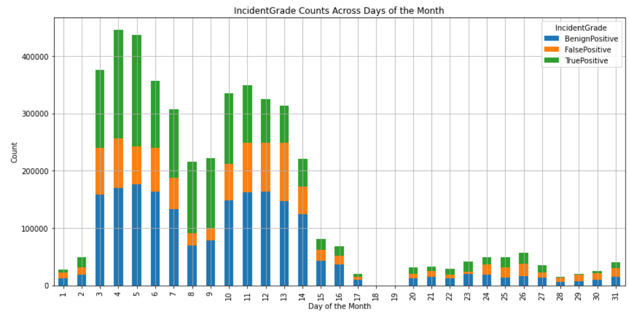

Figure 4: Incident grade counts across Days of the month

In Fig. 5. Incident grade counts across hours of the year provides a broad, annual overview, revealing seasonal trends in cyberattacks. Expect to observe if certain months or seasons exhibit higher incident rates, possibly linked to holidays, business cycles, or emerging threats. This helps to understand the long term trends of attacks.

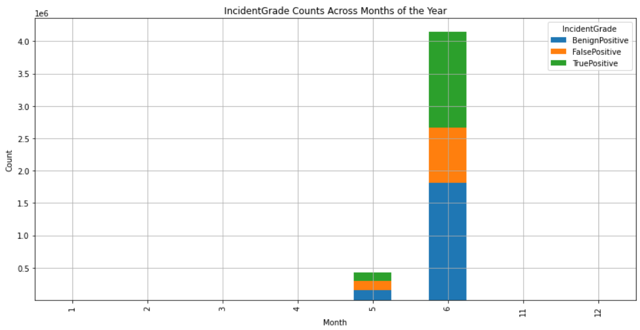

Figure 5: Incident grade counts across Months of the year

In Fig. 6. Distribution of category illustrates the relative frequency of different categories of cyber incidents. It shows the overall prevalence of various attack types, helping prioritize security efforts and understand the most common threats faced. This is a count of the types of attacks.

Paper ID: SR251024103823 DOI: https://dx.doi.org/10.21275/SR251024103823

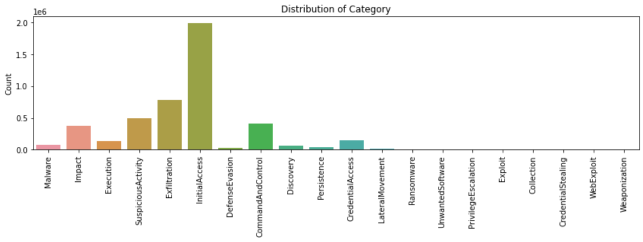

Figure 6: Disstribution of category

In Fig. 7. Distribution of incidentGrade shows the proportions of Benign, False, and True Positives. It reveals the accuracy of the incident classification system, highlighting potential areas for improvement in detection and response strategies. This shows the count of the results of the security system

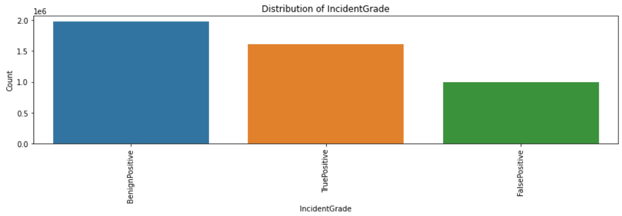

Figure 7: Distribution of incidentGrade

Fig. 8. Correlation Map This visualization displays the relationships between different features in the incident data. It helps identify potential dependencies and redundancies, informing feature selection for machine learning models and revealing hidden patterns in attack characteristics. This shows which datapoints relate to each other.

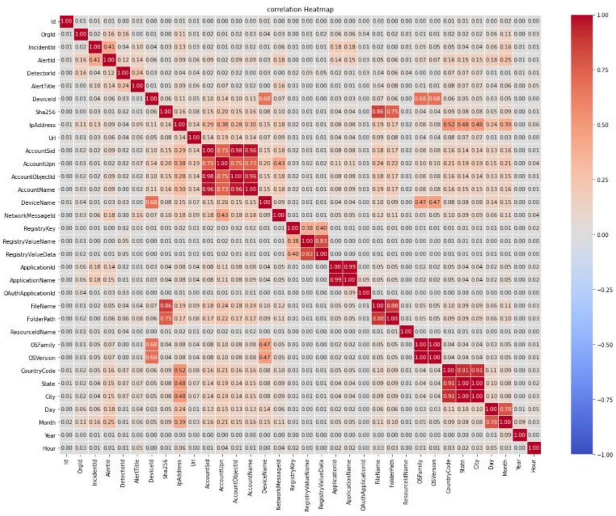

Figure 8: Correlation Map

4. Proposed Method

The proposed method enhances cyber incident detection using machine learning, addressing class imbalance via SMOTE (Synthetic Minority Over-sampling Technique). The study evaluates both baseline and advanced machine learning models, including Decision Trees, Logistic Regression, Random Forest, Gradient Boosting, XGBoost, and LightGBM, to determine the most effective approach. Among these models, XGBoost achieves the highest macro F1-score of 0.91 before addressing class imbalance, indicating its superior performance in cybersecurity threat detection.

To further improve detection accuracy, the dataset undergoes pre-processing steps, including handling missing values, feature selection, and encoding categorical variables. Given the presence of severe class imbalance, SMOTE is applied to generate synthetic samples of the minority class, thereby improving the model's ability to detect rare but critical cyber incidents. After applying SMOTE, the XGBoost model achieves an accuracy of 0.90, precision of 0.92, recall of 0.92, and an F1-score of 0.92, demonstrating significant performance improvement.

This approach leverages ensemble learning techniques to enhance robustness while mitigating data imbalance issues, ensuring that the system effectively predicts and mitigates cybersecurity threats. The results highlight the importance of combining advanced machine learning models with data-balancing techniques to develop a more effective and reliable cyber incident detection system. The system architecture shown in fig.9.

System Architecture

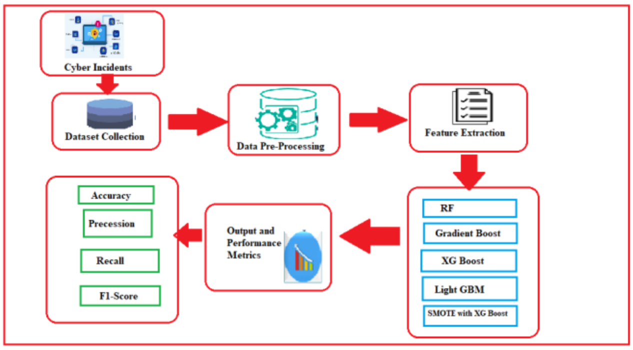

Figure 9: System Architecture

The proposed methodology for enhanced cyber incident detection and prediction consists of multiple steps, including data preprocessing, feature engineering, model selection, class balancing using SMOTE, and model evaluation. The framework aims to effectively process a large-scale cybersecurity dataset, mitigate class imbalance issues, and leverage machine learning models for accurate prediction.

5. Data Preprocessing

The dataset contains over 9.5 million records with 45 features, capturing various cybersecurity events. However, several attributes exhibit high missing values, which require preprocessing. The key steps include:

- *Handling Missing Values:* Features with more than 90% missing values (e.g., ActionGrouped, ThreatFamily, ResourceType) are removed. Features with moderate missing data (e.g., MitreTechniques with 57.44% missing values) are imputed using statistical techniques.
- *Encoding Categorical Variables:* Text-based categorical attributes (e.g., Category, AlertTitle) are converted into numerical format using label encoding and one-hot encoding where necessary.
- *Feature Scaling:* Numerical attributes are normalized using Min-Max Scaling to improve model performance.

1) Feature Engineering

- *Feature Selection:* Highly correlated and irrelevant features are removed to reduce dimensionality and improve efficiency.

- *Derived Features:* Additional meaningful features are created from timestamps, incident severity, and entity types to enhance predictive capability.

2) *Model Selection*

The study evaluates both baseline and advanced machine learning models for cyber incident detection:

a) ***Random Forest***

Random Forest is an ensemble learning algorithm that constructs multiple decision trees and combines their predictions to enhance accuracy and prevent overfitting the architecture shown in fig. 10. In cyber incident detection, it works as follows:

- The dataset is randomly split into multiple subsets.
- A decision tree is trained on each subset using a random selection of features.
- During inference, the majority vote from all trees determines the final prediction.
- This approach reduces variance and improves robustness against noisy data.

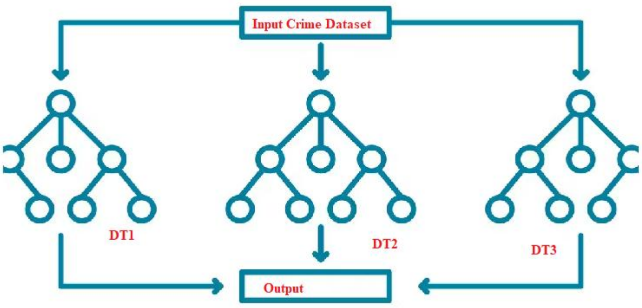

Figure 10: Cyber Incidents with random forest Architeture

b) ***LightGBM***

LightGBM (Light Gradient Boosting Machine) is an optimized gradient boosting framework designed for high-speed performance and efficiency. Tthe architecture shown in fig. 11, Its consists of:

- Leaf-wise growth strategy: Unlike traditional level-wise tree growth, LightGBM expands nodes selectively, reducing computation time.
- Histogram-based feature selection: Bins continuous features into discrete values, improving training efficiency.
- Handling of large datasets: Optimized for handling millions of entries with reduced memory consumption.
- Cyber Incident Detection: LightGBM helps identify attack patterns with minimal training time while maintaining high accuracy.

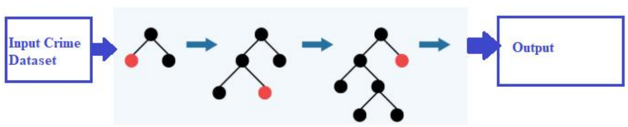

Figure 11: Cyber Incidents with LightGBM Architeture

c) ***Gradient Boosting***

Gradient Boosting is an iterative ensemble learning method that builds trees sequentially, where each tree corrects the errors of the previous one. the architecture shown in fig. 12 Its structure includes:

- Weak learners (decision trees) trained iteratively.
- Loss minimization using gradient descent, refining predictions at each step.
- Final prediction is an additive combination of all weak learners.
- Cybersecurity application: Helps detect subtle cyber threats by improving prediction accuracy iteratively.

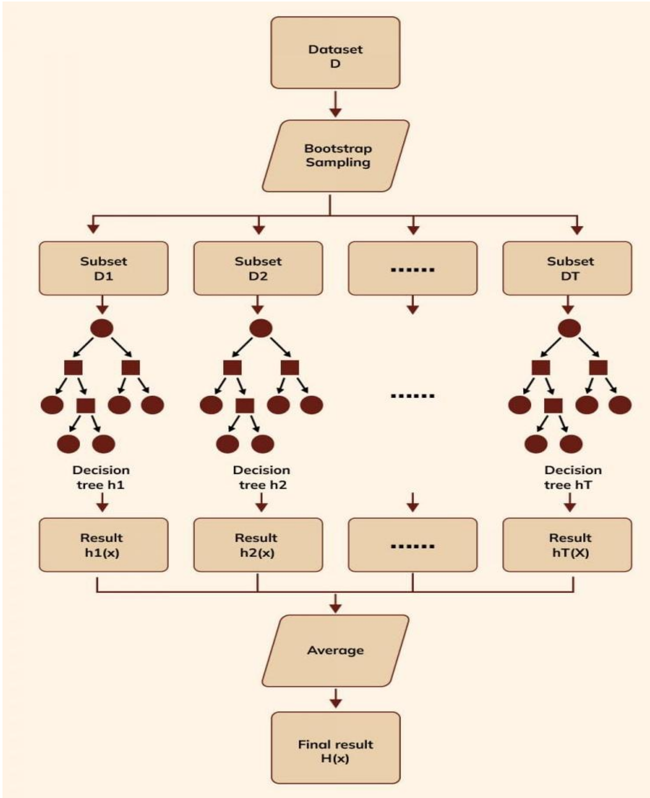

Figure 12: Cyber Incidents with Gradient Architeture

d) ***XG Boost***

XGBoost (Extreme Gradient Boosting) is an optimized version of Gradient Boosting, incorporating the architecture shown in fig. 13:

- Regularization (L1 & L2) to prevent overfitting.
- Tree pruning techniques to improve efficiency.
- Parallel computation, making it faster than traditional boosting techniques.
- Handling of missing values automatically.
- Cyber Incident Detection: XGBoost is highly effective in distinguishing between attack types, achieving the best F1-score (0.92) after SMOTE balancing in this study.

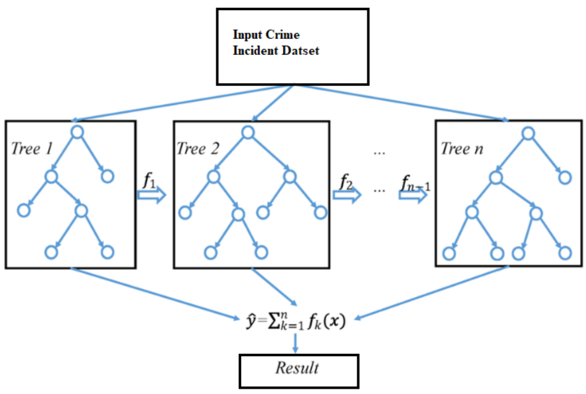

Figure 13: Cyber Incidents with XG Boost Architeture

Paper ID: SR251024103823 DOI: https://dx.doi.org/10.21275/SR251024103823

3) Handling Class Imbalance Using SMOTE

Cybersecurity datasets often suffer from class imbalance, where critical cyber incidents are underrepresented. To address this:

- Synthetic Minority Over-sampling Technique (SMOTE) is applied to generate synthetic samples for the minority class.
- This balances the dataset distribution, ensuring that models learn to detect both frequent and rare cyber incidents effectively.

4) Model Training and Evaluation

After applying SMOTE, the models are trained using optimized hyper parameters. Performance is evaluated using:

- Accuracy, Precision, Recall.F1-score

Post-SMOTE, the XGBoost model achieves an accuracy of 0.90, precision of 0.92, recall of 0.92, and an F1-score of 0.92, significantly improving cybersecurity threat detection.

This methodology effectively integrates data preprocessing, class balancing, and ensemble learning models to develop a robust cybersecurity threat detection system. The combination of XGBoost with SMOTE enhances the detection of critical cyber incidents, making this approach highly suitable for real-time cybersecurity applications

a) Algorithm

---

1) Input:

- Cybersecurity dataset with 9.5 million entries and 45 features
- Target variable (cyber incident classification)

2) Output:

- Trained machine learning model with optimized performance metrics
- Predicted cyber incident labels
  Step 1: Data Preprocessing
  Load Dataset
- Read the cybersecurity dataset into a dataframe.
- Handle Missing Values
- Remove features with >90% missing values.
- For features with moderate missing values (e.g., MitreTechniques), use imputation techniques.
- Encode Categorical Variables
- Apply label encoding and one-hot encoding to categorical features.
- Feature Scaling
  Numerical features are normalized using Min-Max Scaling to maintain consistency in model input ranges.

Step 2: Feature Engineering

- Feature Selection
- Remove highly correlated or irrelevant features.
- Select the most relevant predictive features.
- Feature Creation
- Generate new features (e.g., time-based patterns, severity levels, aggregated indicators).

Step 3: Train Machine Learning Models

- Split Data
- Divide dataset into training (80%) and testing (20%) sets.
- Initialize Models
- Train baseline models:
- Decision Trees
- Logistic Regression
- Train ensemble learning models:
- Random Forest, Gradient Boosting, XGBoost, LightGBM
- Evaluate Initial Performance
- Compute metrics: Accuracy, Precision, Recall, F1-score.
- Select best-performing model (XGBoost with macro F1-score = 0.91).
  Step 4: Handle Class Imbalance Using SMOTE
- Apply Synthetic Minority Over-sampling Technique (SMOTE)
- Generate synthetic samples for the minority class.
- Balance the dataset to improve model robustness.
- Retrain Model
- Train the XGBoost model on the balanced dataset.
- Evaluate Post-SMOTE Performance

End of Algorithm.

---

b) Implementation flow chart

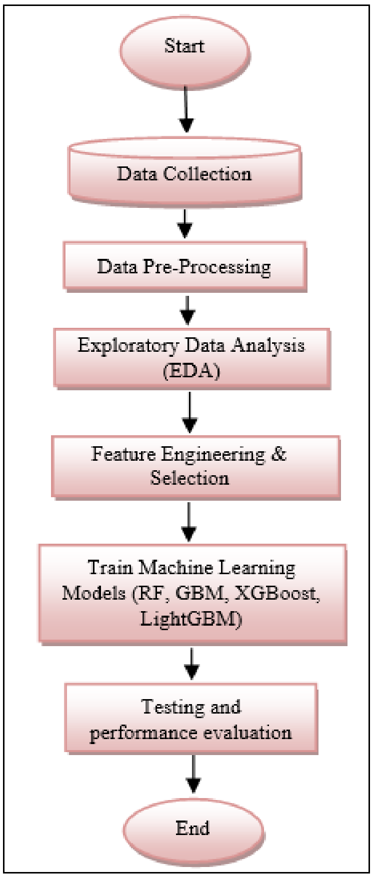

Figure 14: Implmentation Flow Diagram

5) Performance Metrics

Performance measures are used to evaluate the network performance of the proposed model. This work uses accuracy, precision, recall and f1-score as performance measure, which are formulated.

a) Accuracy:

This measures the proportion of correct predictions (both true positives and true negatives) out of the total number of predictions. It's a general indicator of the model's performance but can be misleading in imbalanced datasets.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ Number\ of\ Predictions} \quad (1)$$

b) Precision:

This metric indicates the proportion of true positive predictions among all positive predictions made by the model. High precision means that the model has a low rate of false positives.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Where
TP=True Positives
FP= False Positives

c) *Recall:*

Also known as sensitivity, recall measures the proportion of actual positive cases that the model correctly identified. High recall indicates that the model has a low rate of false negatives.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

Where
TP=True Positives
FP= False Positives
FN=False Negatives

d) *F1-Score:*

The F1 score is the harmonic mean of precision and recall, providing a single metric that balances both concerns. It's particularly useful when dealing with imbalanced datasets, as it considers both false positives and false negatives.

$$F1 - Score = 2\ X \frac{Precision\ X\ Recall}{Precision + \ Recall} \quad (4)$$

6. Results and Discussion

The simulation results highlight the effectiveness of different machine learning models in detecting and predicting cyber incidents. The evaluation was performed using various performance metrics, including Accuracy, Precision, Recall, and F1-score. The dataset contained 9.5 million entries with 45 features, and a key challenge was handling class imbalance, which was addressed using SMOTE (Synthetic Minority Over-sampling Technique).

1) *Baseline Model Performance (Before SMOTE)*

Initially, several machine learning models were tested on the imbalanced dataset. The XGBoost model outperformed others, achieving the highest macro F1-score of 0.91, demonstrating its capability to handle complex cybersecurity patterns. However, due to class imbalance, some minority classes were underrepresented, leading to lower recall.

- Random Forest: Moderate performance due to high variance.
- Gradient Boosting: Performed well but slightly overfit.
- LightGBM: Faster training but slightly lower recall.
- XGBoost: Best F1-score (0.91) but still affected by class imbalance.

2) *Performance After Applying SMOTE*

To mitigate class imbalance, SMOTE was applied, which generated synthetic samples for underrepresented classes. This led to significant improvements in model performance.

Table 1: Comparison of Performace Metrics with Different Models

Model	Accuracy	Precision	Recall	F1-score
Random Forest	0.88	0.89	0.87	0.88
Gradient Boosting	0.89	0.90	0.88	0.89
LightGBM	0.89	0.91	0.89	0.90
XGBoost	0.90	0.92	0.92	0.92

XGBoost achieved the highest accuracy (0.90), precision (0.92), recall (0.92), and F1-score (0.92), making it the most robust model for cyber incident detection. LightGBM showed competitive performance but was slightly outperformed by XGBoost. Gradient Boosting and Random Forest improved but did not surpass XGBoost.
Performance Analysis.

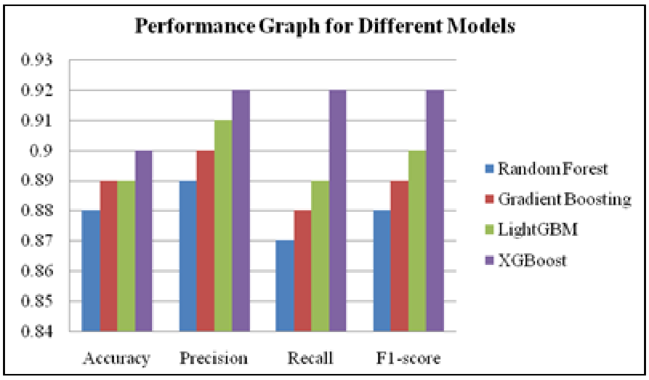

Figure 15: Performace graph for diferent models

Class balancing with SMOTE significantly improved recall and F1-score, reducing bias toward majority classes. XGBoost remained the best model due to its superior feature selection and boosting mechanism. LightGBM performed well but had slightly lower precision compared to XGBoost. Random Forest showed improvement, but its variance made it less stable for cybersecurity data shown in fig.15.

7. Conclusion

This paper explored the application of machine learning techniques to enhance cyber incident detection and prediction, addressing the prevalent issue of class imbalance through the use of the Synthetic Minority Over-sampling Technique (SMOTE). Our comprehensive evaluation encompassed various models, including Decision Trees, Logistic Regression, Random Forest, Gradient Boosting, XGBoost, and LightGBM. Among these, XGBoost demonstrated superior performance, achieving a macro F1-score of 0.91 prior to addressing class imbalance. Post-SMOTE application, the XGBoost model exhibited further improvements, attaining an accuracy of 0.90, precision of 0.92, recall of 0.92, and an F1-score of 0.92. These findings underscore the efficacy of combining ensemble learning techniques with data balancing methods to develop robust cybersecurity threat detection systems.

8. Future Scope

In future the proposed method can be extended with Incorporating real-time data feeds and threat intelligence can enhance the model's ability to detect emerging threats and adapt to evolving attack patterns.

References

[1] Y. Li and Q. Liu, ''A comprehensive review study of cyber-attacks and cyber security; emerging trends and recent developments,'' Energy Rep., vol. 7, pp. 8176–8186, Nov. 2021, doi: 10.1016/j.egyr.2021. 08.126.

[2] J. Kaur and K. R. Ramkumar, ''The recent trends in cyber security: A review,'' J. King Saud Univ.-Comput. Inf. Sci., vol. 34, no. 8, pp. 5766–5781, Sep. 2022, doi: 10.1016/j.jksuci.2021. 01.018.

[3] H. Hejase, H. Kazan, A. Hejase, and I. Moukadem, ''Cyber secu rity paper,'' Comput. Inf. Sci., vol. 14, pp. 10–25, Mar. 2021, doi: 10.5539/cis.v14n2p10.

[4] Cybersecurity Almanac by Cyber Security Ventures. [Online]. Available: https://cybersecurityventures.com/cybersecurity-almanac-2022

[5] Digital 2022 October Global Statshot, by Datareportal. [Online]. Avail able: https://datareportal.com/

[6] P. S. Seemma, S. Nandhini, and M. Sowmiya, ''Overview of cyber secu rity,'' Int. J. Adv. Res. Comput. Commun. Eng., vol. 7, no. 11, pp. 125–128, Nov. 2018, doi: 10.17148/IJARCCE.2018.71127.

[7] Q. E. Hodgson, A. Clark-Ginsberg, Z. Haldeman, A. Lauland, and I. Mitch, Managing Response to Significant Cyber Incidents: Comparing Event Life Cycles and Incident Response Across Cyber and Non-Cyber Events. anta Monica, CA, USA: RAND Corp., 2022, doi: 10.7249/RRA1265-4.

[8] Handa, A. Sharma, and S. K. Shukla, ''Machine learning in cyberse curity: A review,'' WIREs Data Mining Knowl. Discovery, vol. 9, no. 4, p. e1306, Jul. 2019, doi: 10.1002/widm.1306.

[9] Ibor, F. A. Oladeji, and O. B. Okunoye, ''A survey of cyber security approaches for attack detection, prediction, and preven tion,'' Int. J. Secur. Appl., vol. 12, no. 4, pp. 15–28, Jul. 2018, doi: 10.14257/ijsia.2018.12.4.02.

[10] K. Shaukat Dar, S. Luo, S. Chen, and D. Liu, ''Cyber threat detection using machine learning techniques: A performance evaluation perspective,'' in Proc. Int. Conf. Cyber Warfare Secur. (ICCWS), Oct. 2020, pp. 1–6, doi: 10.1109/ICCWS48432.2020.9292388.

[11] Significant Cyber Incidents (SCIs). [Online]. Available: https://www. csis.org/programs/strategic-technologies-program/significant-cyber incidents

[12] Pektaş, M. Eris, and T. Acarman, ''Proposal of n-gram based algorithm for malware classification,'' in Proc. 5th Int. Conf. Emerg. Secur. Inf., Syst. Technol., Jan. 2011, pp. 14–18.

[13] Wressnegger, G. Schwenk, D. Arp, and K. Rieck, ''A close look on n-grams in intrusion detection: Anomaly detection vs. classification,'' in Proc. ACM workshop Artif. Intell. Secur., Nov. 2013, pp. 14–18, doi: 10.1145/2517312.2517316.

[14] S. Soni and B. Bhushan, ''Use of machine learning algorithms for designing efficient cyber security solutions,'' in Proc. 2nd Int. Conf. Intell. Comput., Instrum. Control Technol. (ICICICT), vol. 1, Jul. 2019, pp. 1496–1501, doi: 10.1109/ICICICT46008.2019. 8993253.

[15] *Alqahtani, I. H. Sarker, A. Kalim, S. M. M. Hossain, S. Ikhlaq, and S. Hossain, ''Cyber intrusion detection using machine learning classifi cation techniques,'' in Proc. Int. Conf. Comput. Sci., Commun. Secur., Singapore, 2020, pp. 121–131.*

[16] Terai, S. Abe, S. Kojima, Y. Takano, and I. Koshijima, ''Cyber attack detection for industrial control system monitoring with sup port vector machine based on communication profile,'' in Proc. IEEE Eur. Symp. Secur. Privacy Workshops, Apr. 2017, pp. 132–138, doi: 10.1109/EuroSPW.2017.62.

[17] K. Ghanem, F. J. Aparicio-Navarro, K. G. Kyriakopoulos, S. Lambotharan, and J. A. Chambers, ''Support vector machine for network intru sion and cyber-attack detection,'' in Proc. Sensor Signal Process. Defense Conf. (SSPD), Dec. 2017, pp. 1–5, doi: 10.1109/SSPD.2017. 8233268.

[18] N. Bhusal, M. Gautam, and M. Benidris, ''Detection of cyber attacks on voltage regulation in distribution systems using machine learning,'' IEEE Access, vol. 9, pp. 40402–40416, 2021, doi: 10.1109/ACCESS.2021.3064689.

[19] R. Bapat, ''Identifying malicious botnet traffic using logistic regression,'' in Proc. Syst. Inf. Eng. Design Symp. (SIEDS), Apr. 2018, pp. 266–271, doi: 10.1109/SIEDS.2018.8374749.

[20] Kajal and G. Sardana, ''Protection from cyber attacks using IDS security mechanism with random forest classifier: A review,'' J. Crit. Rev., vol. 7, no. 19, p. 8516, 2020.