Grenze International Journal of Engineering and Technology, June Issue

A Comprehensive Study on Capturing Complex Feature Interactions with Advanced Feature Selection and Extraction in IoT Intrusion Detection Systems

Sangapu Sreenivasa Chakravarthi[1], Sukaranam Mani Sarma[2] and Mutyala Karthik[3]
[1-3]Amrita School of Engineering, Chennai, India
Email: sschakravarthi@ch.amrita.edu, manisarma1401@gmail.com

***Abstract*—Efficient Intrusion Detection Systems (IDS) are required due to the growing cybersecurity threats that have been caused by the Internet of Things' (IoT) devices. Class imbalance, high-dimensional data, and feature redundancy are common challenges for traditional IDS, which affect detection accuracy and computational complexity. An improved feature selection and extraction method that enhances intrusion detection in Internet of Things networks is presented in this paper. Preprocessing, correlation-based feature selection, autoencoderbased feature extraction, and Synthetic Minority Over-sampling Technique (SMOTE) for balancing datasets are all part of the methodology. A better set of features is examined via a Gradient Boosting classifier that is less computationally expensive with high accuracy. The results show that IDS performance is enhanced through sensitive feature selection, thus qualifying for real-time IoT security applications**

***Index Terms*— Intrusion Detection System (IDS), Internet of Things (IoT), Feature Selection, Feature Extraction, Autoencoder, Correlation-based Feature Selection, Synthetic Minority Over-sampling Technique (SMOTE), Class Imbalance, Highdimensional Data, Gradient Boosting Classifier, Cybersecurity, Real-time IoT Security.**

I. INTRODUCTION

Cybercriminals have a wider attack surface with the sudden boom of IoT devices across industries. Below are challenges plaguing traditional intrusion detection systems (IDS): High-dimensional feature spaces, which result in inefficient computation. Unbalanced attack distributions, which may result in skewed learning models. Features that are redundant or irrelevant, decreasing the accuracy of detection. Methods for feature extraction and selection assist with dimensionality reduction, feature preservation, and enhancing computing efficiency. To enhance the accuracy and effectiveness of attack detection, this work introduces an optimized feature extraction and selection framework intended for IoT-IDS.

II. RESEARCH METHODOLOGY

Review Scope This article addresses issues like computationally inefficient high-dimensional data, class imbalance, and high-dimensional data by emphasizing sophisticated feature selection and extraction methods for Internet of Things Intrusion Detection Systems (IDS).

Grenze ID: 01.GIJET.11.2.503_1

For improving the accuracy of detection, it investigates several feature selection approaches, namely filter, wrapper, embedding, and hybrid methods. To provide a better representation of features, the research also investigates feature extraction strategies such as autoencoders, PCA, and deep learning-based approaches. Additionally, methods of handling class imbalance such as SMOTE as well as combined sampling are addressed. Investigation into how different methods affect IDS performance measures such as precision, false alarms, as well as computation cost is carried out. The work also provides future research directions for enhancing IoT security and identifies major concerns like scalability, increased cyber attacks, computational efficiency, and model explainability

III. Survey of Related Work

A few related studies have been conducted based on work surveys to improve feature selection-based IoT intrusion detection models by utilizing different techniques with the intention of enhancing accuracy in detection, reducing false positives, and dealing with high dimensional data. The case for the IoT environment is tricky because the environments are heterogeneous and resource-constrained; thus, traditional IDS is not very effective.

Sanjay Misra et al. [8] discussed the gaps of IoT IDS: datasets used were outdated, NSL-KDD, as well as the need for realistic datasets like BoT-IoT and UNSW-NB15; they report over 95% accuracy with Bi-LSTM RNN on UNSW-NB15.

They proposed a hybrid approach combining IG with PCA dimension reduction and indicated that diverse comparative studies are needed. For strengthening the performance of the IDS, the authors also proposed an ensemble model with the Voting Gray Wolf Optimizer (GWO).

Mouaad Mohy-eddine et al. [13] proposed an intrusion detection model based on election-based feature selection and K-Nearest Neighbors (K-NN) for IoT networks. Their model emphasized increased accuracy (ACC) and fewer false alarm rate (FAR) based on PCA, GA, and univariate statistical tests in feature selection. Again, with the election-based feature selection method, the number of features dropped to five, where using this set, the model bettered performance on the Bot-IoT dataset to 98.78%. This also reduced the training time and proved that feature selection enhances IoT IDS effectively[13].

Bouke et al. [15] suggested a Decision Tree (DT)-based model with an enhanced Gini index-based feature selection procedure for discovering DDoS attacks on IoT networks. Out of 45 security features, they selected 13 features. It can be noticed that the data dimension was highly reduced with good improvement results. Their proposed model outperformed other more complex models like Random Forest and XGBoost models as the accuracy reached 98% with a lower false alarm rate of 2% on the UNSW-NB15 dataset. Feature selection technique helped avoid overfitting and reduced the computational complexity[15].

Wathiq Laftah Al-Yaseen et al. [34] claimed that wrapper techniques and, in specific, the genetic algorithm-based approach improve the accuracy of IDS systems. Lee et al. (2020) worked with NSL-KDD dataset used random forest with Sequential Floating Forward Search (SFFS) achieving up to 84.4% accuracy but implying low efficiency of computations. The authors present that the standard methods suffer from suboptimal feature sets and processing delay issues nowadays. A differential evolution-based feature selection with an extreme learning machine classifier is proposed by the authors, who achieved high accuracy and efficiency with 96.8% of normal traffic correctly classified.[34]

Bakır et al. [2] focus their research on the drawbacks with which machine learning-based IDS are often affected: suboptimal hyper-parameter tuning, as managed by genetic algorithms approach and features selection, which is one of the most common gaps with many studies that fail to combine methods properly, an aspect resolved by the hybrid approach used in this paper. The authors make a review of key advances and gaps in IDS and in particular the advances and gaps on IOS networks. Another aspect is that their approach works well on testing different kinds of datasets, such as CICIDS2017, and also deals with class imbalance. Based on their experiment, their approach results in reducing the time taken to train the models by more than 40% over XGBoost and 98% over RF, while still maintaining an accuracy of 98.67% for RF and 98% for XGBoost.

Louai A. Maghrab et al.[5] review methodologies for IoTbased Network Intrusion Detection Systems (NIDS). Key findings include the effectiveness of RNN, LSTM, and CNN, with AlZewairi et al. achieving 94.4% accuracy on the UNSW-NB15 dataset. Hybrid models like CNN and LSTM have proven successful, while adversarial detection mechanisms show the importance of handling evasion attacks.

The proposed Random Forest-based NIDS achieved 90.17% accuracy on unbalanced datasets and 98.83% on balanced ones, surpassing state-of-theart methods. Precision, recall, and F1 scores were all around 90%, with data resampling boosting performance for minority attacks.

Rehab Hosny et al. [41] reviewed feature selection techniques for improvement of the performance of IDS. The remarkable methods include Balakrishnan et al.'s Information Gain Ratiobased method and Ramakrishnan et al.'s entropy-based selection with a high accuracy value.

III. Experimental Setup and Dataset

- IoT-23 Dataset Overview The IoT-23 dataset is an open dataset used for IoT-based intrusion detection by Stratosphere Laboratory in collaboration with Avast AI and Cybersecurity. It is appropriate for Intrusion Detection System (IDS) testing due to the presence of 23 labeled network traffic captures of various IoT devices in benign and malicious settings.
- Dataset Characteristics The captured network traffic (PCAP files) comprising the IoT23 dataset includes both normal and attack scenarios. The dataset was collected from various IoT devices, including IP cameras, smart speakers, and Raspberry Pi-based IoT setups.
  - Diverse Attack Scenarios: Port scanning, botnets, malware infections, DDoS attacks, and command-and-control (C&C) communication.
  - Real IoT Activity: Grounded in actual deployments of IoT devices.
  - Labeled Data: All sessions are labeled as malicious or benign, with additional labels for specific attack types.
  - Flow-Based Features: The raw network packets are processed to extract meaningful statistical and behavioral features, transforming them into a machine-learningcompatible format.
- Attack Categories in IoT-23 The dataset contains multiple attack types originating from various malware families and attack vectors. The major attack categories include:
  - DDoS Attacks: IoT devices are overloaded with excessive traffic, disrupting services.
  - Port Scanning: Attackers probe IoT devices to identify open ports and potential vulnerabilities.
  - Botnets (Mirai, Tsunami, Gafgyt): IoT devices are compromised and remotely controlled for coordinated attacks.
  - Brute-Force Attacks: Automated password-guessing techniques are used to gain unauthorized access. • Command & Control (C&C): Malicious servers remotely control infected IoT devices to execute cyberattacks.
  - Malware Infections: IoT devices are infected with malware like Mirai or Gafgyt, enabling attackers to exploit them.
- Preprocessing Dataset Preprocessing steps are necessary to convert unstructured network packets into structured data for feature extraction since IoT-23 is delivered in PCAP format.
  Step 1: Feature Extraction: To extract flow-based features from raw packet captures, we utilize CICFlowMeter, which translates network packets into bidirectional network flow features, including
  - Basic Features: Packet count, byte count, and flow duration.
  - Time-Based Features: Packet arrival times between packets.
  - Statistical Characteristics: Mean packet size, packet size variation, minimum and maximum packet sizes.
  - Protocol-Based Features: Flow start information and TCP flags.

  Step 2: Data Cleaning and Formatting:
  - Handling Missing Values: Using imputation methods.
  - Removing Irrelevant Columns: Such as timestamps and uninformative metadata.
  - Categorical Feature Encoding: Encoding protocol types using one-hot encoding or label encoding.

Step 3: Feature Selection and Engineering:
  - Autoencoder-based feature extraction: Extracts important patterns.
  - Correlation-based feature selection: Eliminates redundant features.
  - Synthetic Minority Over-sampling Technique (SMOTE): Handles class imbalance.
- Why IoT-23 Must Be Used
  - Realistic IoT Security Threats: Records actual attack scenarios.
  - A Variety of IoT Devices: Offers generalizability by embracing a variety of IoT devices.
  - Handling of Imbalanced Data: Enables testing of sampling algorithms such as SMOTE.
  - Flow-Based Network Analysis: Focuses on network activity rather than fixed malware signatures.
- Summary of Preprocessing Steps

Key preprocessing processes that were applied to the IoT-23 dataset are summarized in Table I. The IoT-23 dataset is effectively preprocessed and optimized for training an IDS model that accurately detects IoT-based cyber threats due to this systematic approach.

IV. Understanding Complex Data Relationships

Understanding complex data relationships is crucial for effective feature selection in IoT intrusion detection systems. IoT network traffic often involves nonlinear interactions and diverse patterns, making traditional selection methods insufficient. With the growing variety of IoT data, advanced techniques like ensemble learning, deep learning, and hybrid models are needed to capture these dependencies.
Methods such as correlation analysis and mutual information help identify the most relevant features while considering their interconnections. This comprehensive approach enhances detection accuracy and strengthens system resilience against evolving threats.

A. Types of Relationships

Understanding the kinds of relationships that exist among data features is essential in the area of IoT Intrusion Detection Systems (IDS) for effective feature selection and performance of a model. The most general and all-encompassing categorization of the types of relationships is into three:
Linear Relationships: These occur when there is a direct proportional relationship between the two variables. Indeed, if one of the variables increases or decreases, the other one does in the same proportion. It is often possible to capture linear relationships using linear regression techniques, making them particularly easy to model and interpret. However, relying solely on linear assumptions might miss more complex interactions that could be going on in a real-world dataset.
Non-Linear Relationships: We speak of relationships in a much more complicated form where the effect is not constant, i.e., one variable does not always affect another in the same amount as seen in linear cases. Polynomial relationships and interactions would generally require models like decision trees or even neural networks to explain specific underlying patterns and interactions. Identifying and modeling these relationships properly may be critical for bettering the predictability of IDSs.

B. Degrees of Complexity in Data Relationships

This lays out the nuances of complexity in data relationships and how these would evolve towards developing an efficient Intrusion Detection Systems for IoT networks. Because a piece of data can be described with a spectrum of complexity levels, data relationship descriptions will have a strong impact on performance and feature selection methods/models.
Low-Degree Relationships: These relationships refer to the associations between just two attributes, which is a very naive understanding of the way data interacts. A good grasp of such simple relationships is essential to build basic models that may capture simple patterns in network behavior.

TABLE I: PREPROCESSING STEPS FOR IOT-23

Step	Process	Tools Used
Extracting Features	Convert PCAP files to flow-based features	CICFlowMeter
Data Cleaning	Remove null values, drop unnecessary features	Pandas, NumPy
Feature Selection	Select relevant features using autoencoders and correlation analysis	Scikit-learn, Autoencoders, Correlation Filters
Handling Class Imbalance	Apply SMOTE to balance attack classes	Imbalanced-learn
Model Training	Train classifier for intrusion detection	TensorFlow, GradientBoosting

Challenges in Capturing Complex Relationships

- Curse of Dimensionality: High-dimensional data leads to sparsity, making it difficult to identify meaningful patterns, especially in IoT datasets.
- Overfitting and Generalization: Complex models may fit noise instead of patterns, harming performance on unseen data—crucial in dynamic security environments.
- Computational Complexity: Analyzing intricate feature interactions demands high computational power, potentially causing delays in threat detection for IoT systems.

V. Feature Interaction Loss

Losing key feature interactions during selection or extraction can degrade model performance, particularly in IoT, where correlated data streams from many devices can form complex behavioral patterns. For instance, network latency, packet size, and device response might jointly indicate an attack—interactions easily missed by simple techniques.

Types of Feature Interactions:

- First-Order: Linear pairwise effects (e.g., CPU usage affecting network speed), sometimes captured by methods like PCA.
- Higher-Order: Interactions among three or more features (e.g., CPU, battery, packet size), often requiring advanced models like Random Forests or Gradient Boosting Machines to detect.

VI. Results and Discussion

Using the IoT-23 dataset, the proposed Enhanced Feature Selection and Extraction Framework improved intrusion detection accuracy, reduced computational load, and addressed class imbalance. Key evaluation areas included feature importance, model performance under various strategies, and redundancy reduction.

A. Impact of Feature Selection on Extraction

Feature selection and extraction reduce dimensionality, eliminate noise, and enhance generalization by focusing on the most informative features.

- Correlation-Based Selection: Highly correlated features (Pearson > 0.9) were removed to avoid redundancy and multicollinearity. This preserved accuracy while improving efficiency and interpretability.

B. Model Performance Evaluation

Model performance was assessed with and without preprocessing using metrics like accuracy, precision, recall, F1-score, and ROC-AUC.

- Autoencoder-Based Extraction: Autoencoders compressed high-dimensional data into a latent space, capturing complex, non-linear relationships missed by traditional methods. These encoded features enhanced the model's effectiveness and generalization.

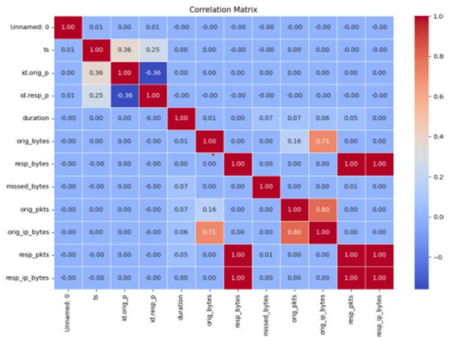

Figure 1: Correlation Heatmap Showing Feature Relationship

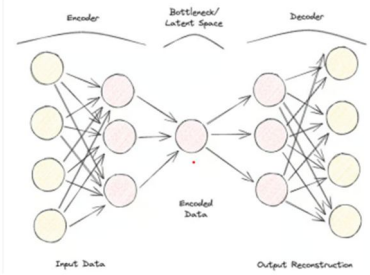

Fig. 2. Autoencoder-Based Feature Representation

Figure 2: Autoencoder based Feature Representation

TABLE II: Comparison of Feature Selection Approaches

Work	Method	Acc.	Strengths	Weaknesses
Bakır et al. [2]	GA, AOA, XGBoost, RF	98%	Fast training (XGBoost), handles class imbalance	Complex feature selection
Maghrab et al. [5]	CNN, LSTM, RNN	94.4%	Hybrid model, good adversarial detection	Poor performance on unbalanced data
Misra et al. [8]	IG, PCA, Bi-LSTM	95%	Effective ensemble model	Uses outdated datasets
Mohy-eddine et al. [13]	Election-based FS, K-NN	98.78%	High accuracy, reduced features	Limited dataset evaluation
Bouke et al. [15]	DT, Gini FS	98%	Low false alarms (2%), feature reduction	May not generalize well
Fraihat et al. [19]	AOA FS, ML models	98% (binary)	Strong feature reduction	Low detection for certain attacks
Mananayaka et al. [19]	Ensemble, THE-AFS-DT	94.31%	High TPR, better detection	Lower accuracy on large datasets
Cao et al. [22]	Stacked ensemble, GWO	97.1%	High accuracy, low FPR (0.64%)	High resource demands for IoT
Prakash et al. [26]	M2SFS framework	95.36%	High precision, low false positives	Scalability issues for large IoT
Al-Yaseen et al. [34]	GA, RF	96.8%	Good normal traffic handling	Expensive for real-time
Halim et al. [31]	GA, DT, PCA	96%	Efficient FS, high accuracy	GA may face convergence issues
Hosny et al. [41]	ReliefF, Weighted KNN	97.1%	Good across models, high accuracy	Limited dataset applicability

TABLE III: EVALUATION METRICS FOR IDS MODEL

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Without Feature Selection	94.2%	91.5%	89.7%	90.6%	92.1%
With Feature Selection	**97.8%**	**96.3%**	**95.9%**	**96.1%**	**98.4%**

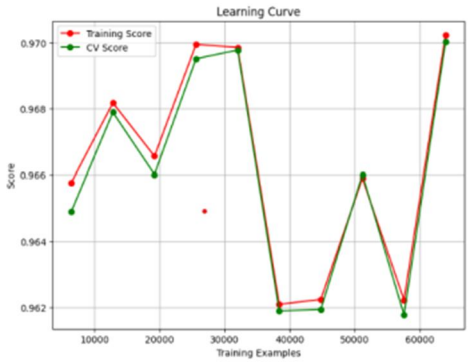

Figure 3. Learning Curve: Training vs. Validation Score

C. Learning Curve and Generalization

The learning curve was used to assess the model's training behavior and generalization ability. It showed a smooth convergence between training and validation loss, indicating minimal overfitting. SMOTE oversampling addressed early instability caused by class imbalance. As training data grew, validation performance improved, highlighting the effectiveness of feature selection in reducing variance.

D. Addressing Class Imbalance with SMOTE

IoT intrusion detection often suffers from class imbalance, with attack data under-represented. SMOTE was used to generate synthetic attack samples, balancing the dataset and improving detection.
Before SMOTE: High false negatives on rare attacks.
After SMOTE: Improved recall and F1-score without harming precision, enhancing detection of both common and rare attacks.

E. Comparison with Existing IDS Models

Compared to traditional IDS approaches, the proposed method showed higher accuracy and efficiency:
Legacy models struggled with high-dimensional, noisy data.The use of autoencoders and correlation-based filtering improved feature quality and generalization.
Enhanced models were both more accurate and computationally efficient.

F. Discussion

Why Gradient Boosting?
Effective on high-dimensional, imbalanced data.
Promotes generalization via smart feature selection.
Benefits of Feature Selection
Cuts computation time.
Improves model clarity.
Enables real-time IDS use.
Limitations & Future Work
Needs real-world validation.
Future studies should explore adaptive feature selection and online learning.

V. CONCLUSION

This framework integrates SMOTE, autoencoder-based extraction, and correlation-based feature selection to address key challenges in IoT security: feature redundancy, high dimensionality, and class imbalance. It improves accuracy, reduces training time, and is well-suited for real-time deployment. Future research should focus on real-world testing and adaptive, online techniques to further enhance intrusion detection.

REFERENCES

[1] Li, Jing, Mohd Shahizan Othman, Hewan Chen, and Lizawati Mi Yusuf. ”Optimizing IoT intrusion detection system: feature selection versus feature extraction in machine learning.” Journal of Big Data 11, no. 1 (2024): 36.

[2] Bakır, Halit, and ¨ Ozlem Ceviz. ”Empirical enhancement of intrusion detection systems: a comprehensive approach with genetic algorithm based hyperparameter tuning and hybrid feature selection.” Arabian Journal for Science and Engineering (2024): 1-19.

[3] Osman, Musa, Jingsha He, Nafei Zhu, and Fawaz Mahiuob Mohammed Mokbal. ”An ensemble learning framework for the detection of RPL attacks in IoT networks based on the genetic feature selection approach.” Ad Hoc Networks 152 (2024): 103331.

[4] Eljialy, A. E. M., Mohammed Yousuf Uddin, and Sultan Ahmad. ”Novel framework for an intrusion detection system using multiple feature selection methods based on deep learning.” Tsinghua Science and Technology 29, no. 4 (2024): 948-958.

[5] Maghrabi, Louai A.”Automated Network Intrusion Detection for Inter net of Things Security Enhancements.” IEEE Access (2024).

[6] Azimjonov, Jahongir, and Taehong Kim. ”Designing accurate lightweight intrusion detection systems for IoT networks using f ine-tuned linear SVM and feature selectors.” Computers Security 137 (2024): 103598.

[7] Alshaeaa, Hyder Yahya, and Zainab Mohammed Ghadhban. ”Develop ing a hybrid feature selection method to detect botnet attacks in IoT devices.” Kuwait Journal of Science 51, no. 3 (2024): 100222.