Advanced Machine Learning Techniques for Intrusion Detection System Development

N. Vijayalakshmi*, K. Sujith

PG and Research Department of Computer Science, Annai College of Arts and Science, Kovilacheri, Kumbakonam - 612 503

*Corresponding author

doi: https://doi.org/10.21467/proceedings.173.14

ABSTRACT

Intrusion Detection Systems (IDS) play a critical role in safeguarding digital infrastructures against evolving cyber threats. This research explores the integration of advanced machine learning (ML) techniques to enhance the detection accuracy and adaptability of IDS. Traditional IDS methodologies often struggle with scalability, false positives, and identifying novel attack patterns. To address these challenges, this study leverages advanced ML algorithms, including deep learning architectures, ensemble models, and anomaly detection techniques, for effective threat identification. A comprehensive dataset is utilized to evaluate system performance, ensuring robust benchmarking across various metrics such as accuracy, precision, recall, and F1-score. Furthermore, feature selection and dimensionality reduction methods are employed to optimize computational efficiency while maintaining predictive performance. The results demonstrate that advanced ML techniques significantly improve intrusion detection capabilities compared to conventional methods, particularly in detecting zero-day attacks and minimizing false positives. This paper concludes with a discussion on the implications of integrating advanced ML models into real-world IDS applications and outlines future directions for research in this domain.

1. INTRODUCTION

The rapid evolution of technology and the increasing reliance on interconnected systems have made digital infrastructures highly susceptible to cyber-attacks. Intrusion Detection Systems (IDS) are pivotal in identifying and mitigating such threats, acting as a first line of defense against malicious activities in networks and systems. However, traditional IDS methods often suffer from limitations such as high false positive rates, poor scalability, and difficulty in detecting novel or zero-day attacks. These challenges necessitate the exploration of more sophisticated techniques to enhance IDS performance and resilience. Machine Learning (ML) has emerged as a transformative approach in the field of cyber security, offering the ability to analyze large volumes of data, uncover patterns, and adapt to evolving threats. The integration of advanced ML techniques, such as deep learning, ensemble models, and anomaly detection, into IDS frameworks has shown promise in addressing the shortcomings of traditional methods. These techniques not only improve detection accuracy but also enable real-time decision-making and reduce the dependency on predefined signatures. In this paper, we investigate the application of advanced ML techniques for the development of robust IDS. Our approach emphasizes leveraging cutting-edge algorithms to address key challenges, including handling imbalanced datasets, improving detection rates, and minimizing false alarms. Furthermore, we discuss the role of feature engineering and dimensionality reduction in optimizing the computational efficiency of ML-based IDS. The objective of this research is twofold: to demonstrate the superiority of ML-driven IDS over conventional approaches and to provide insights into practical implementation strategies for real-world scenarios. By evaluating the performance of these techniques on comprehensive datasets, we aim to establish a benchmark for the design and deployment of intelligent IDS [2].

Proceedings DOI: 10.21467/proceedings.173; Series: AIJR Proceedings; ISSN: 2582-3922; ISBN: 978-81-970666-x-x

2 .ALGORITHM: ML-BASED INTRUSION DETECTION SYSTEM

Input:

Network traffic dataset D (labeled or unlabeled)

Features F={f1,f2,...,fn}

ML algorithms A={A1,A2,...,Am}

Output:

Optimized ML model M for detecting intrusions

Step 1: Data Collection and Preprocessing

Dataset Collection:

Acquire a benchmark dataset (e.g., NSL-KDD, CICIDS2017) containing network traffic and labeled intrusions.

Data Cleaning:

Handle missing values and remove duplicate entries.

Feature Encoding:

Convert categorical features (e.g., protocols, attack types) to numerical values using techniques like one-hot encoding or label encoding.

Step 2: Data Partitioning

Split the dataset into:

Training set (70%−80%70%−80%)

Validation set (10%−15%10)

Test set (10%−15%)

Ensure class balance during splitting to address data imbalance.

Step 3: Model Development

Model Selection:

Choose suitable ML techniques based on the problem type:

Supervised Learning: Random Forest, Support Vector Machines (SVM), XGBoost, or Neural Networks (for labeled data).

Unsupervised Learning: k-Means Clustering or Auto encoders (for unlabeled data).

Hybrid Models: Combine supervised and unsupervised methods (e.g., ensemble models).

Step 4: Model Evaluation

Evaluate the trained models on the test set using metrics such as:

Accuracy

Proceedings DOI: 10.21467/proceedings.173
ISBN: 978-81-970666-x-x

Series: AIJR Proceedings
ISSN: 2582-3922

Precision

Recall

F1-Score

Area Under Curve (AUC)

Step 5: Deployment and Monitoring

System Integration:

Deploy the optimized model M in a real-time IDS environment.

Real-Time Inference:

Process incoming network traffic through the model for intrusion detection.

Continuous Learning:

Periodically retrain the model with new data to adapt to evolving attack patterns.

Step 6: Data Preprocessing

```
data = load_dataset("CICIDS2017")
data = clean_data(data)
data = encode_features(data)
data = scale_features(data)
```

Step 7: Split the data

```
X_train, X_test, y_train, y_test = train_test_split(data, test_size=0.2, stratify=True)
```

Step 8: Model Training

```
est_model = None
best_score = 0
for model in [RandomForest(), XGBoost(), NeuralNetwork()]:
 model.fit(X_train, y_train)
 score = evaluate_model(model, X_test, y_test)
 if score > best_score:
 best_model = model
 best_score = score
```

Step 9: Evaluate and Deploy

```
evaluate_model(best_model, X_test, y_test)
deploy_model(best_model)
```

3 .METHODOLOGIES

3.1 ENSEMBLE FEATURE SELECTION

Feature selection plays a vital role in machine learning tasks due to the capacity of alleviating the curse of dimensionality and improving robustness. In the intrusion detection domain, the filter-based feature selection technique is the mainstream, which de- couples the process of feature selection and the training of the classifier. Although the strategy indeed saves time consumption in decision-making, the feature selection procedure is generally volatile. To overcome the weakness, we propose an ensemble feature selection algorithm to ensure the robustness of the selected optimal subset.

3.2 LIGHT GBM

Light GRM is an gradient boosting decision tree (GDDT) algorithm, which is applied in various fields final feature importance. Finally, we apply forward search to select the optimal feature combination. In the procedure of forward search, we first sort the feature importance in descending order to ensure that the most significant feature is posed in the first position. The sort procedure is necessary since the final feature important.

3.3 FEATURE SELECTION FRAMEWORK.

Light GBM based ensemble feature selection algorithm. The framework consists of five steps, i.e., sampling subsets, training base selectors, getting rankings. Aggregating rankings, and performing forward search.

3.4 BATCH NORMALIZATION

Batch normalization is a universal technique in training neural networks. The technique works since it rescales the inputs in a batch to the learnable scale, which indeed facilitates the procedure of stochastic gradient descend. **3.5 EMBEDDING**

In this study, the embedding technique is used to map the representation of categorical features from high-dimensional vector space to low-dimensional vector space. However, the simple approach will lead to the high sparsity of the feature space, especially when the number of values is large. Besides, in the one-hot space, we cannot identify the similarity among values since the distance between all value pairs is the same.

4. CONCLUSION

The increasing sophistication of cyber threats necessitates the development of more robust and intelligent Intrusion Detection Systems (IDS). This research has demonstrated the potential of advanced machine learning (ML) techniques in addressing the limitations of traditional IDS, such as high false positive rates and limited adaptability to novel attack patterns. By leveraging state-of-the-art algorithms, including deep learning, ensemble methods, and anomaly detection, IDS can achieve higher accuracy, improved detection rates, and better scalability. Through comprehensive experimentation, this study has highlighted the importance of feature engineering, data preprocessing, and model optimization in enhancing IDS performance. Advanced ML techniques not only provide superior threat detection capabilities but also enable real-time processing and adaptability, making them highly suitable for modern, dynamic network environments. Moreover, addressing challenges such as imbalanced datasets and computational efficiency has further strengthened the feasibility of deploying these systems in real-world scenarios. While the results are promising, the evolving nature of cyber threats requires continuous research and innovation. Future work could focus on hybrid models that combine multiple ML techniques, integration with other cyber

Proceedings DOI: 10.21467/proceedings.173
ISBN: 978-81-970666-x-x

Series: AIJR Proceedings
ISSN: 2582-3922

security tools, and the development of datasets that better reflect real-world traffic patterns and attack diversity. Additionally, incorporating explainable AI (XAI) frameworks could improve the interpretability of ML based IDS, fostering trust and adoption in critical sectors. In conclusion, the integration of advanced machine learning techniques marks a significant step forward in the development of intelligent and efficient IDS, paving the way for more secure and resilient digital infrastructures.

5 Declarations

5.1 Competing Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] Amiri, F., Yousefi, M. R., Lucas, C., Shakery, A., & Yazdani, N. (2011). Mutual information-based feature selection for intrusion detection systems. Journal of Network and Computer Applications, 34(4), 1184–1199.

[2] Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cybersecurity intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153–1176.

[3] Chandrasekhar, N. & Raghuveer, K. (2013). Intrusion detection system using feature selection andclassification technique. International Journal of Computer Applications, 84(3), 30–36.

[4] Liao, H. J., Lin, C. H. R., Lin, Y. C., & Tung, K. Y. (2013). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36(1), 16–24.

[5] Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). 2015 Military Communications and Information Systems Conference (MilCIS), 1–6

[6] Roy, S. S., & Cheung, W. K. (2018). A deep learning approach for intrusion detection in Internet of Things using bi-directional LSTM. IEEE Internet of Things Journal, 5(6), 4621–4631.

[7] Sahu, R., & Parsai, M. P. (2013). A survey on intrusion detection system using machine learning techniques. International Journal of Advanced Research in Computer and Communication Engineering, 2(11), 4349–4355