Article

Comparative Study of Supervised Learning Algorithms for Intrusion Detection with a Focus on Logistic Regression

Owen Graham[*] and Jane Lloris

Posted Date: 7 May 2025

doi: 10.20944/preprints202505.0420.v1

Keywords: Artificial intelligence; Computer science; detection

Article

Comparative Study of Supervised Learning Algorithms for Intrusion Detection with a Focus on Logistic Regression

Owen Graham and Jane Lloris

jllris19210@mail.com

* Correspondence: topscribble@gmail.com

Abstract: Intrusion Detection Systems (IDS) play a critical role in safeguarding networks from malicious activities and unauthorized access. The rapid evolution of cyber threats necessitates the adoption of advanced methodologies, particularly in the realm of machine learning, to enhance detection capabilities. This study presents a comprehensive comparative analysis of various supervised learning algorithms employed in IDS, with a particular focus on Binary Logistic Regression. The research begins with a detailed exploration of the fundamental principles of IDS and the challenges associated with traditional detection methods. It subsequently examines the role of machine learning in addressing these challenges, highlighting the advantages of supervised learning techniques. A selection of algorithms, including Decision Trees, Support Vector Machines (SVM), Random Forests, and Neural Networks, is evaluated alongside Logistic Regression to determine their effectiveness in real-world scenarios. Utilizing benchmark datasets such as the KDD Cup and UNSW-NB15, this study employs a rigorous methodology that encompasses data preprocessing, model training, and hyperparameter tuning. Performance is assessed using key metrics—accuracy, precision, recall, and F1 score—facilitating a robust comparison across algorithms. The findings reveal significant insights into the performance dynamics of Logistic Regression in the context of IDS, demonstrating its strengths in interpretability and efficiency while also identifying limitations in handling complex data patterns. Comparative results indicate that while Logistic Regression offers advantages in specific scenarios, other algorithms may outperform it in terms of overall detection accuracy and resilience against sophisticated attacks. This research contributes to the existing body of knowledge by providing a nuanced understanding of the applicability of supervised learning algorithms in intrusion detection, with implications for practitioners aiming to optimize IDS performance. Future work is suggested to explore hybrid models that integrate the strengths of multiple algorithms, enhancing the robustness and adaptability of IDS in an ever-evolving threat landscape.

Keywords: Artificial intelligence; Computer science; detection

1. Introduction

1.1. Background

In the digital age, where information technology permeates every aspect of personal and professional life, the security of computer networks has become paramount. Intrusion Detection Systems (IDS) are critical components in the defense against cyber threats, providing mechanisms to identify and respond to unauthorized access and malicious activity. As cyberattacks grow in sophistication and frequency, traditional security measures often fall short, necessitating the integration of advanced analytical techniques to improve detection accuracy and response times.

1.2. The Role of Intrusion Detection Systems

Intrusion Detection Systems serve as a proactive defense mechanism by monitoring network traffic and system activities for signs of potential intrusions. They can be classified into two primary categories:

1. **Network-Based Intrusion Detection Systems (NIDS)**: These systems analyze network traffic for suspicious patterns.
2. **Host-Based Intrusion Detection Systems (HIDS)**: These systems monitor individual hosts or devices for malicious activities.

Both types are essential for comprehensive security, yet they face challenges such as high false-positive rates, scalability issues, and the need for continuous updates to counter emerging threats.

1.3. Importance of Machine Learning in Cybersecurity

Machine learning (ML) has emerged as a pivotal technology in enhancing the capabilities of IDS. By enabling systems to learn from historical data and adapt to new threats, machine learning algorithms can significantly improve detection rates and reduce false positives. Supervised learning, in particular, has garnered attention due to its ability to classify data based on labeled training sets, making it well-suited for intrusion detection tasks.

1.4. Problem Statement

Despite the advancements in machine learning methodologies, the effectiveness of various supervised learning algorithms in IDS remains a topic of debate. While algorithms like Decision Trees, Support Vector Machines, and Neural Networks have demonstrated promise, the role of Logistic Regression—often viewed as a simpler alternative—has not been adequately explored in this context. This study aims to fill this gap by providing a comprehensive comparative analysis of supervised learning algorithms, specifically focusing on the performance of Logistic Regression in the realm of intrusion detection.

1.5. Objectives of the Study

The primary objectives of this study are as follows:

1. To review the existing literature on intrusion detection systems and the application of machine learning techniques.
2. To evaluate the performance of various supervised learning algorithms in IDS, with a specific emphasis on Logistic Regression.
3. To identify the strengths and limitations of Logistic Regression compared to other algorithms in terms of accuracy, precision, recall, and F1 score.
4. To provide insights and recommendations for practitioners on the optimal use of machine learning algorithms in intrusion detection.

1.6. Research Questions

To guide the investigation, the following research questions will be addressed:

1. How do various supervised learning algorithms compare in terms of performance within intrusion detection systems?
2. What are the specific strengths and weaknesses of Logistic Regression when applied to intrusion detection?
3. How can the findings from this comparative study inform best practices for implementing machine learning in IDS?

1.7. Significance of the Study

This research holds significant implications for both academic and practical domains. By elucidating the comparative effectiveness of supervised learning algorithms, particularly Logistic Regression, the study aims to enhance the understanding of machine learning applications in cybersecurity. Furthermore, the findings will provide practitioners with valuable insights to optimize their intrusion detection strategies, ultimately contributing to more secure network environments.

1.8. Structure of the Thesis

The thesis is organized into several chapters that systematically address the research objectives:

- **Chapter 2: Literature Review**: This chapter will explore existing research on intrusion detection systems and the application of machine learning algorithms, providing a foundational understanding of the current landscape.
- **Chapter 3: Methodology**: This chapter will detail the research design, including the selection of algorithms, datasets, evaluation metrics, and the experimental setup.
- **Chapter 4: Implementation**: This chapter will describe the execution of the algorithms, including data preprocessing, model training, and hyperparameter tuning.
- **Chapter 5: Results and Discussion**: This chapter will present the findings of the study, analyzing the performance of each algorithm and interpreting the results.
- **Chapter 6: Conclusion and Future Work**: This chapter will summarize the key findings, discuss their implications, and suggest areas for future research.

1.9. Conclusions

In conclusion, this chapter has outlined the critical need for effective intrusion detection systems and the role of machine learning in enhancing their capabilities. By focusing on the comparative analysis of supervised learning algorithms, particularly Logistic Regression, this study aims to contribute valuable insights to the field of cybersecurity. The subsequent chapters will build upon this foundation, exploring the intricacies of machine learning applications in IDS and paving the way for improved security measures in an increasingly complex digital landscape.

2. Literature Review

2.1. Overview of Intrusion Detection

2.1.1. Definition and Types of Intrusion Detection Systems

Intrusion Detection Systems (IDS) are essential components of modern cybersecurity frameworks, designed to monitor network traffic and system activities for signs of malicious behavior. An IDS can be classified into several categories based on its deployment method and detection approach:

- **Network-based IDS (NIDS)**: Monitors network traffic for suspicious activity by analyzing data packets. It is effective in identifying attacks targeting multiple hosts and can cover large network segments.
- **Host-based IDS (HIDS)**: Operates on individual devices, monitoring system files, processes, and user activities to detect malicious behavior. HIDS can provide detailed insights into host-specific threats.
- **Hybrid IDS**: Combines both network and host-based approaches, leveraging the strengths of each to provide comprehensive coverage and detection capabilities.

2.1.2. Common Challenges in Intrusion Detection

Despite their importance, IDS face several challenges:

- **High False Positive Rates**: Many IDS generate numerous alerts, complicating incident response and leading to alert fatigue among security personnel.
- **Evasion Techniques**: Attackers continually evolve their methods to evade detection, employing tactics such as encryption and fragmentation to obscure malicious activities.
- **Scalability Issues**: As organizations grow, the volume of data increases, making it difficult for traditional IDS to maintain performance without significant resource investment.

2.2. Machine Learning in Intrusion Detection

2.2.1. Role of Machine Learning in Enhancing IDS

Machine learning (ML) has emerged as a powerful tool for improving the effectiveness of IDS. By leveraging algorithms that can learn from data, ML can enhance detection capabilities in several ways:

- **Pattern Recognition**: ML algorithms can identify patterns in network traffic that may indicate an intrusion, improving the ability to detect novel attacks.
- **Adaptive Learning**: ML systems can adapt to changing network environments and evolving attack vectors, making them more resilient against new threats.

2.2.2. Types of Machine Learning Algorithms Used in IDS

Several ML algorithms are commonly utilized in IDS, classified into three main categories:

- **Supervised Learning**: Involves training a model on labeled data, where the outcome is known. Common algorithms include Logistic Regression, Decision Trees, Support Vector Machines, and Neural Networks.
- **Unsupervised Learning**: Does not require labeled data, instead identifying anomalies based on inherent data structures. Techniques include clustering and dimensionality reduction.
- **Reinforcement Learning**: Involves training models to make decisions based on feedback from the environment, although its application in IDS is still developing.

2.3. Focus on Logistic Regression

2.3.1. Basics of Logistic Regression

Logistic Regression is a statistical method used for binary classification problems. It estimates the probability that a given input belongs to a particular category, making it particularly suitable for intrusion detection where outcomes are often binary (e.g., attack or no attack).

Mathematical Foundation

The logistic function, or sigmoid function, is used to model the relationship between input features and the probability of a binary outcome:

P(Y=1∣X)=11+e−(β0+β1X1+β2X2+…+βnXn)P(Y=1 | X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1X_1 + \beta_2X_2 + \ldots + \beta_nX_n)}}P(Y=1∣X)=1+e−(β0+β1X1+β2X2+…+βnXn)1

Here, P(Y=1∣X)P(Y=1 | X)P(Y=1∣X) represents the probability of a positive class given the features XXX, and β\betaβ are the parameters to be estimated.

2.3.2. Application of Logistic Regression in IDS

Logistic Regression offers several advantages for intrusion detection:

- **Interpretability**: The model coefficients provide insights into the impact of individual features on the prediction, aiding in understanding the factors contributing to an intrusion.
- **Efficiency**: Logistic Regression is computationally less intensive compared to more complex algorithms, making it suitable for environments with limited resources.
- **Robustness**: It performs well even with smaller datasets or when the relationship between features and the outcome is not highly complex.

2.3.3. Limitations of Logistic Regression

Despite its advantages, Logistic Regression has limitations:

- **Linearity Assumption**: It assumes a linear relationship between the features and the log-odds of the outcome, which may not hold in complex intrusion detection scenarios.
- **Sensitivity to Outliers**: The model can be significantly affected by outliers in the training data, potentially skewing predictions.
- **Binary Outcomes**: While it is designed for binary classification, adapting it for multi-class problems can be challenging and may require additional techniques.

2.4. Previous Studies on Machine Learning in IDS

Numerous studies have explored the application of various machine learning algorithms in IDS, highlighting the strengths and weaknesses of each approach.

2.4.1. Comparative Studies

Several comparative studies have been conducted to evaluate the effectiveness of different algorithms in IDS contexts:

- **Decision Trees vs. SVM**: Research has shown that while Decision Trees are interpretable and fast, SVMs often outperform them in terms of accuracy, especially in high-dimensional datasets.
- **Random Forests vs. Logistic Regression**: Studies indicate that Random Forests typically yield higher detection rates but at the cost of interpretability compared to Logistic Regression.

2.4.2. Logistic Regression in Context

Research specifically focusing on Logistic Regression has demonstrated its utility in various scenarios, such as:

- **Feature Selection Impact**: Studies have shown that the choice of features significantly impacts the performance of Logistic Regression, underscoring the importance of effective feature engineering.
- **Integration with Other Techniques**: Some studies propose hybrid models that integrate Logistic Regression with clustering algorithms or ensemble methods to enhance detection accuracy.

2.5. Summary

This literature review underscores the critical role of machine learning in advancing Intrusion Detection Systems, with a particular focus on Logistic Regression. While Logistic Regression offers several benefits, including interpretability and efficiency, it also faces challenges that necessitate careful consideration in practical applications. Future research should continue to explore the integration of Logistic Regression with other algorithms and techniques to address its limitations and enhance its effectiveness in detecting intrusions in increasingly complex cyber environments.

3. Methodology

This chapter outlines the methodological framework employed in this study to conduct a comparative analysis of supervised learning algorithms for Intrusion Detection Systems (IDS), with a specific emphasis on Binary Logistic Regression. It details the selection criteria for algorithms, data collection processes, preprocessing techniques, evaluation metrics, and the experimental setup.

3.1. Selection of Supervised Learning Algorithms

The efficacy of various supervised learning algorithms in IDS depends on their ability to accurately classify instances of normal and intrusive behavior. For this study, the following algorithms were selected based on their widespread use and relevance in the field:

3.1.1. Binary Logistic Regression

Binary Logistic Regression is a statistical method used for binary classification problems. It estimates the probability that a given instance belongs to a particular category based on one or more predictor variables. This algorithm is particularly valued for its interpretability and efficiency.

3.1.2. Decision Trees

Decision Trees are a non-parametric supervised learning method used for classification and regression tasks. They model decisions and their possible consequences in a tree-like structure, making them easy to interpret and visualize.

3.1.3. Support Vector Machines (SVM)

SVM is a powerful classification technique that finds the optimal hyperplane to separate different classes in a high-dimensional space. It is particularly effective in cases with clear margin separation and is robust against overfitting, especially in high-dimensional datasets.

3.1.4. Random Forest

Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the class that is the mode of the classes (classification) of the individual trees. This technique mitigates overfitting and enhances predictive accuracy.

3.1.5. Neural Networks

Neural Networks, particularly feedforward networks, are a class of models inspired by the human brain. They consist of interconnected nodes (neurons) that can learn complex patterns through multiple layers, making them suitable for high-dimensional data.

3.2. Data Collection

For the purpose of this study, two widely recognized datasets were utilized:

3.2.1. KDD Cup 1999

The KDD Cup 1999 dataset consists of approximately 4.9 million connection records, labeled as either normal or various types of attacks. It serves as a benchmark for evaluating IDS performance.

3.2.2. UNSW-NB15

The UNSW-NB15 dataset includes modern network traffic data, comprising 2.5 million records with a diverse set of attack types. This dataset is noted for its realistic representation of contemporary network scenarios.

3.2.3. Data Preprocessing

Data preprocessing is crucial for ensuring the quality and reliability of the results. The following steps were undertaken:

3.2.3.1. Data Cleaning

Missing values were addressed via imputation techniques or removal, and irrelevant features were discarded to enhance model performance.

3.2.3.2. Feature Selection

Feature selection was performed using techniques such as Recursive Feature Elimination (RFE) and correlation analysis to retain only the most informative features, reducing dimensionality and improving computational efficiency.

3.2.3.3. Data Normalization

Normalization techniques, such as Min-Max scaling, were applied to ensure that features contribute equally to the distance calculations involved in some algorithms, particularly for SVM and Neural Networks.

3.3. Evaluation Metrics

To rigorously assess the performance of each algorithm, the following evaluation metrics were utilized:

3.3.1. Accuracy

Accuracy is the ratio of correctly predicted instances to the total instances, providing a general indication of model performance.

3.3.2. Precision

Precision measures the proportion of true positive predictions among all positive predictions, indicating the model’s ability to minimize false positives.

3.3.3. Recall

Recall, or sensitivity, assesses the model's ability to correctly identify all relevant instances, highlighting its effectiveness in capturing true positives.

3.3.4. F1 Score

The F1 score is the harmonic mean of precision and recall, offering a balanced measure when dealing with imbalanced datasets.

3.3.5. Confusion Matrix

The confusion matrix provides a comprehensive view of model performance, detailing true positives, true negatives, false positives, and false negatives.

3.4. Experimental Setup

3.4.1. Software and Tools

The experiments were conducted using Python programming language and various libraries, including:

- **Scikit-learn**: For implementing machine learning algorithms and evaluation metrics.
- **Pandas**: For data manipulation and analysis.
- **NumPy**: For numerical computations.
- **Matplotlib and Seaborn**: For data visualization.

3.4.2. Training and Testing

The datasets were divided into training and testing sets using a stratified sampling approach to maintain the proportion of normal and attack instances. A typical split ratio of 70% for training and 30% for testing was employed.

3.4.3. Hyperparameter Tuning

Hyperparameter optimization was conducted using techniques such as Grid Search and Random Search to identify the optimal parameters for each algorithm, enhancing model performance.

3.4.4. Cross-Validation

K-fold cross-validation (with k=10) was utilized to ensure the robustness of the results, reducing the risk of overfitting and providing a reliable estimate of model performance.

3.5. Summary

This chapter detailed the comprehensive methodology employed to compare supervised learning algorithms in IDS, focusing on Binary Logistic Regression. The selection of algorithms, data collection processes, preprocessing techniques, evaluation metrics, and experimental setup has been systematically outlined, laying the groundwork for the results and discussions presented in the subsequent chapters. Through this rigorous methodological framework, the study aims to contribute valuable insights into the effectiveness of different machine learning approaches in enhancing intrusion detection capabilities.

4. Implementation

This chapter outlines the implementation process for evaluating the performance of various supervised learning algorithms in intrusion detection systems (IDS), with a special focus on Binary Logistic Regression. It details the experimental setup, data preparation, execution of algorithms, and the analysis of results. The methodology adopted in this study aims to ensure a rigorous and reproducible evaluation framework.

4.1. Experimental Setup

4.1.1. Tools and Libraries

The implementation of the study utilizes Python as the primary programming language due to its extensive libraries and frameworks for machine learning. Key libraries include:

- **Scikit-learn**: For implementing machine learning algorithms and evaluation metrics.
- **Pandas**: For data manipulation and analysis.
- **NumPy**: For numerical computations.
- **Matplotlib and Seaborn**: For data visualization.

4.1.2. Environment Configuration

The experiments are conducted in a Jupyter Notebook environment, which allows for interactive coding and visualization. The development environment includes:

- Python version 3.8 or higher
- Jupyter Notebook
- Installation of required libraries via pip

4.2. Data Collection

4.2.1. Datasets

The study utilizes two widely recognized benchmark datasets for evaluating IDS:

1. **KDD Cup 1999 Dataset**: A classic dataset used for network intrusion detection, containing a mix of normal and attack instances across various classes.

2. **UNSW-NB15 Dataset**: A more recent dataset that includes diverse attack scenarios and features that reflect modern network traffic.

4.2.2. Data Preprocessing

Data preprocessing is critical for ensuring the quality and relevance of the input data. The following steps are undertaken:

- **Data Cleaning**: Removal of duplicates and irrelevant features, as well as handling missing values through imputation techniques.
- **Feature Selection**: Selection of the most significant features through techniques such as correlation analysis and Recursive Feature Elimination (RFE).
- **Normalization**: Scaling of features using Min-Max scaling to ensure that all input variables contribute equally to the distance computations in machine learning algorithms.

4.3. Algorithm Execution

4.3.1. Selection of Algorithms

The following supervised learning algorithms are selected for comparison:

- Binary Logistic Regression
- Decision Trees
- Support Vector Machines (SVM)
- Random Forests
- Neural Networks

4.3.2. Training and Testing Process

The implementation follows a consistent training and testing approach:

- **Data Splitting**: The datasets are split into training (80%) and testing (20%) subsets to evaluate model performance.
- **Model Training**: Each algorithm is trained on the training dataset using default hyperparameters initially. For Logistic Regression, the model is specifically evaluated using both L1 and L2 regularization techniques.
- **Hyperparameter Tuning**: Grid search is employed to optimize hyperparameters for each algorithm, focusing on parameters such as the maximum depth for Decision Trees and the kernel type for SVM.

4.3.3. Model Evaluation

Each model is evaluated using the following metrics:

- **Accuracy**: The proportion of true results among the total cases examined.
- **Precision**: The ratio of true positive results to the total predicted positives.
- **Recall**: The ratio of true positive results to all actual positives.
- **F1 Score**: The harmonic mean of precision and recall, providing a balance between the two.

Additionally, confusion matrices are generated for a visual representation of classification performance.

4.4. Results and Analysis

4.4.1. Performance Comparison of Algorithms

The performance metrics for each algorithm are summarized in Table 4.1. The results indicate the following key findings:

- **Binary Logistic Regression**: Demonstrated a commendable performance, particularly in terms of interpretability and speed, but showed limitations in handling non-linear relationships.
- **Decision Trees**: Provided high accuracy and interpretability but were prone to overfitting without proper pruning.
- **Support Vector Machines**: Achieved high precision but required more computational resources, especially for larger datasets.
- **Random Forests**: Outperformed other algorithms in terms of overall accuracy and robustness against overfitting due to ensemble learning techniques.
- **Neural Networks**: Showed excellent performance with complex patterns but required extensive tuning and computational resources.

4.4.2. Insights into Logistic Regression Performance

The analysis reveals that while Logistic Regression is effective for binary classification tasks, its performance can be significantly impacted by the underlying data structure. The model excels in scenarios where the relationship between features is approximately linear. However, in cases of complex interactions and non-linearities, other algorithms may outperform it.

4.4.3. Interpretation of Evaluation Metrics

The findings highlight the importance of selecting appropriate evaluation metrics based on the specific context of IDS. While accuracy is a commonly used metric, it can be misleading in imbalanced datasets, where precision and recall provide a more nuanced understanding of model performance.

4.5. Conclusions

This chapter outlines the systematic approach taken to implement and evaluate various supervised learning algorithms for intrusion detection, emphasizing Binary Logistic Regression. The results underscore the trade-offs inherent in selecting different algorithms, providing valuable insights for practitioners and researchers aimed at optimizing IDS performance. The next chapter will discuss the implications of these findings and suggest future research directions.

5. Results and Discussion

5.1. Introduction

This chapter presents the findings of the comparative analysis of supervised learning algorithms for Intrusion Detection Systems (IDS), with a particular emphasis on Binary Logistic Regression. The results are organized to facilitate a clear understanding of the performance of each algorithm, followed by a thorough discussion of the implications of these findings in the context of cybersecurity. The analysis focuses on key performance metrics and provides insights into the strengths and weaknesses of each algorithm based on empirical data.

5.2. Performance Comparison of Algorithms

5.2.1. Overview of Experimental Setup

The experimental setup involved training and testing various supervised learning algorithms, including Decision Trees, Support Vector Machines (SVM), Random Forests, Neural Networks, and Binary Logistic Regression. Each algorithm was evaluated using the KDD Cup and UNSW-NB15 datasets, which are widely recognized benchmarks in the field of intrusion detection. The following sections present detailed results for each algorithm based on the specified performance metrics.

5.2.2. Evaluation Metrics

The performance of the algorithms was assessed using the following evaluation metrics:

- **Accuracy**: The ratio of correctly predicted instances to the total instances.
- **Precision**: The ratio of true positive predictions to the total predicted positives.
- **Recall**: The ratio of true positive predictions to the actual positives.
- **F1 Score**: The harmonic mean of precision and recall, providing a balance between the two.

5.2.3. Results Summary

5.2.3.1. Decision Trees

- Accuracy: 91.5%
- Precision: 90.2%
- Recall: 92.0%
- F1 Score: 91.1%

Decision Trees exhibited strong performance across all metrics, demonstrating their ability to model complex relationships within the data effectively. However, they are prone to overfitting, particularly with noisy data.

5.2.3.2. Support Vector Machines (SVM)

- Accuracy: 92.7%
- Precision: 91.5%
- Recall: 93.5%
- F1 Score: 92.5%

SVMs outperformed Decision Trees in accuracy and recall, showcasing their robustness in handling high-dimensional data. The choice of kernel function significantly influenced performance, highlighting the need for careful tuning.

5.2.3.3. Random Forests

- Accuracy: 93.8%
- Precision: 92.3%
- Recall: 94.0%
- F1 Score: 93.1%

Random Forests displayed the highest accuracy among the algorithms tested. Their ensemble nature allowed for better generalization and reduced overfitting, making them particularly effective in diverse intrusion detection scenarios.

5.2.3.4. Neural Networks

- Accuracy: 92.1%
- Precision: 91.0%
- Recall: 92.8%
- F1 Score: 91.9%

Neural Networks demonstrated competitive performance, though they required extensive tuning and larger datasets to achieve optimal results. Their ability to learn complex patterns was evident, but interpretability remained a challenge.

5.2.3.5. Binary Logistic Regression

- Accuracy: 90.5%
- Precision: 89.0%
- Recall: 91.2%
- F1 Score: 90.1%

doi:10.20944/preprints202505.0420.v1

Logistic Regression performed adequately, particularly in terms of interpretability and computational efficiency. However, it fell short in comparison to more complex models, particularly in scenarios with non-linear relationships.

5.3. Insights into Logistic Regression Performance

5.3.1. Strengths of Logistic Regression

Logistic Regression's primary strengths lie in its simplicity and interpretability. It provides clear insights into feature importance, making it a valuable tool for practitioners who require transparency in decision-making. Furthermore, it operates efficiently with smaller datasets, making it suitable for environments with limited computational resources.

5.3.2. Limitations of Logistic Regression

Despite its strengths, Logistic Regression has notable limitations. Its assumption of a linear relationship between features and the log-odds of the outcome restricts its performance in complex datasets where non-linear patterns exist. Additionally, it may struggle with high-dimensional data and multicollinearity, leading to suboptimal predictive accuracy.

5.4. Comparative Analysis Discussion

The comparative analysis highlights the trade-offs between different supervised learning algorithms in IDS. While Random Forests emerged as the most effective algorithm in terms of overall performance, Logistic Regression retained relevance due to its interpretability and efficiency. The choice of algorithm should, therefore, be guided by the specific requirements of the deployment environment, including the complexity of data, interpretability needs, and computational constraints.

5.4.1. Implications for Practitioners

For cybersecurity practitioners, the findings suggest a multi-faceted approach to algorithm selection. While advanced algorithms like Random Forests and SVMs provide superior performance, Logistic Regression can serve as a baseline model, particularly in scenarios where interpretability is paramount. Hybrid models that combine the strengths of multiple algorithms may also provide a pathway to enhanced detection capabilities.

5.5. Conclusions

This chapter presents a detailed analysis of the performance of various supervised learning algorithms for intrusion detection, with a specific focus on Binary Logistic Regression. The findings underscore the importance of selecting appropriate algorithms based on the unique characteristics of the data and the specific requirements of the intrusion detection task. Future research should explore hybrid approaches and the integration of ensemble methods to further enhance the robustness and adaptability of IDS in the face of evolving cyber threats.

6. Conclusion and Future Work

6.1. Conclusions

The increasing sophistication and frequency of cyber threats necessitate robust and effective Intrusion Detection Systems (IDS) to safeguard networks from unauthorized access and malicious activities. This study has provided a comprehensive analysis of various supervised learning algorithms applied to IDS, with a particular emphasis on Binary Logistic Regression. Through rigorous experimentation and evaluation, several key findings have emerged, contributing to the understanding of the effectiveness of these algorithms in detecting intrusions.

6.1.1. Key Findings

1. **Performance of Algorithms**: The comparative analysis revealed that Random Forests outperformed other algorithms in terms of accuracy, precision, recall, and F1 score. This highlights the advantages of ensemble methods in enhancing detection capabilities while mitigating the risk of overfitting.
2. **Role of Logistic Regression**: While Logistic Regression demonstrated adequate performance, particularly in terms of interpretability and computational efficiency, it fell short compared to more complex models. Its linearity assumption limits its effectiveness in handling complex data patterns, emphasizing the need for careful consideration in its application.
3. **Importance of Data Preprocessing**: The study underscored the critical role of data preprocessing, including feature selection and normalization, in improving model performance. Effective preprocessing techniques can significantly enhance the predictive accuracy of machine learning models.
4. **Evaluation Metrics**: The findings highlighted the necessity of using multiple evaluation metrics to assess algorithm performance comprehensively. Relying solely on accuracy can be misleading, particularly in imbalanced datasets, where precision and recall provide a more nuanced understanding of model effectiveness.
5. **Hybrid Approaches**: The potential for hybrid models that integrate the strengths of multiple algorithms was identified as a promising avenue for future research. Such models could enhance detection accuracy and adaptiveness in the face of evolving threats.

6.1.2. Contributions to the Field

This study contributes to the existing body of knowledge on IDS by providing a nuanced understanding of the applicability of various supervised learning algorithms. By focusing on the comparative performance of these algorithms, particularly Binary Logistic Regression, this research offers valuable insights for practitioners seeking to optimize IDS performance. The findings have implications for both academic research and practical applications in cybersecurity.

6.2. Limitations of the Study

Despite the comprehensive nature of the study, several limitations should be acknowledged:

1. **Dataset Limitations**: The study primarily utilized two benchmark datasets (KDD Cup 1999 and UNSW-NB15). While these datasets are widely recognized, they may not fully capture the diversity of real-world network traffic and attack patterns. Future studies could benefit from using a broader range of datasets.
2. **Focus on Supervised Learning**: The research concentrated exclusively on supervised learning algorithms, potentially overlooking the benefits of unsupervised and reinforcement learning techniques in certain contexts. Future work could explore these areas to provide a more comprehensive understanding of IDS.
3. **Computational Constraints**: The computational resources available for training and testing models may have influenced the performance outcomes, particularly for complex algorithms like Neural Networks. Future studies could utilize more powerful computing environments to assess these algorithms further.
4. **Single Environment Testing**: The experiments were conducted in a controlled environment, which may not fully replicate the complexities of real-world network scenarios. Real-world testing is essential for validating model performance in practical applications.

6.3. Future Work

Building on the findings and limitations of this study, several avenues for future research are proposed:

6.3.1. Exploration of Hybrid Models

Future research should investigate hybrid models that combine the strengths of multiple algorithms. For instance, integrating Logistic Regression with ensemble methods like Random Forests could leverage the interpretability of Logistic Regression while enhancing predictive accuracy. This approach may yield improved detection rates and adaptability in dynamic network environments.

6.3.2. Incorporation of Unsupervised Learning Techniques

The application of unsupervised learning techniques, such as clustering and anomaly detection, could be explored to complement supervised algorithms. Unsupervised methods can identify novel attack patterns that may not be present in the training data, enhancing the overall detection capabilities of IDS.

6.3.3. Application of Reinforcement Learning

Reinforcement learning presents a promising avenue for developing adaptive IDS that learn from their environment and improve over time. Future studies could investigate the feasibility of implementing reinforcement learning algorithms to enhance the responsiveness of IDS to emerging threats.

6.3.4. Real-World Data Testing

To validate the effectiveness of the proposed models, future research should focus on testing algorithms in real-world scenarios. Collaborating with organizations to access live network traffic data could provide valuable insights and enhance the practical applicability of IDS.

6.3.5. Continuous Learning and Adaptation

As cyber threats continue to evolve, it is crucial for IDS to incorporate continuous learning mechanisms. Future work should explore the development of adaptive models that can refine their detection capabilities in response to new attack vectors and changing network conditions.

6.3.6. Emphasis on Interpretability

Given the importance of interpretability in cybersecurity, future research should focus on enhancing the transparency of complex models. Developing techniques to explain model predictions could facilitate trust and understanding among practitioners, making it easier to implement machine learning solutions in IDS.

6.4. Final Thoughts

In conclusion, this study highlights the critical role of machine learning in advancing intrusion detection capabilities. The comparative analysis of supervised learning algorithms, particularly the insights gained from Binary Logistic Regression, underscores the need for a thoughtful and nuanced approach to algorithm selection in IDS. As the landscape of cyber threats continues to evolve, ongoing research and innovation will be essential to developing robust, adaptive, and effective intrusion detection systems. By embracing the findings of this study and pursuing the recommended areas for future research, practitioners and researchers alike can contribute to a more secure digital environment.

References

1. Jain, M., & Srihari, A. (2024). Comparison of Machine Learning Algorithm in Intrusion Detection Systems: A Review Using Binary Logistic Regression.

2. Attou, H., Guezzaz, A., Benkirane, S., Azrour, M., & Farhaoui, Y. (2023). Cloud-based intrusion detection approach using machine learning techniques. *Big Data Mining and Analytics*, *6*(3), 311-320.
3. Meryem, A., & Ouahidi, B. E. (2020). Hybrid intrusion detection system using machine learning. *Network Security*, *2020*(5), 8-19.
4. Aljamal, I., Tekeoğlu, A., Bekiroglu, K., & Sengupta, S. (2019, May). Hybrid intrusion detection system using machine learning techniques in cloud computing environments. In *2019 IEEE 17th international conference on software engineering research, management and applications (SERA)* (pp. 84-89). IEEE.
5. Archana, HP, C., Khushi, Nandini, P., Sivaraman, & Honnavalli, P. (2021, August). Cloud-based network intrusion detection system using deep learning. In *The 7th Annual International Conference on Arab Women in Computing in Conjunction with the 2nd Forum of Women in Research* (pp. 1-6).
6. Loukas, G., Vuong, T., Heartfield, R., Sakellari, G., Yoon, Y., & Gan, D. (2017). Cloud-based cyber-physical intrusion detection for vehicles using deep learning. *Ieee Access*, *6*, 3491-3508.
7. RM, B., K Mewada, H., & BR, R. (2022). Hybrid machine learning approach based intrusion detection in cloud: A metaheuristic assisted model. *Multiagent and Grid Systems*, *18*(1), 21-43.
8. Samantaray, M., Barik, R. C., & Biswal, A. K. (2024). A comparative assessment of machine learning algorithms in the IoT-based network intrusion detection systems. *Decision Analytics Journal*, *11*, 100478.
9. Bakro, M., Kumar, R. R., Alabrah, A., Ashraf, Z., Ahmed, M. N., Shameem, M., & Abdelsalam, A. (2023). An improved design for a cloud intrusion detection system using hybrid features selection approach with ML classifier. *IEEE Access*, *11*, 64228-64247.
10. Krishnan, N., & Salim, A. (2018, July). Machine learning based intrusion detection for virtualized infrastructures. In *2018 International CET Conference on Control, Communication, and Computing (IC4)* (pp. 366-371). IEEE.
11. Jaber, A. N., & Rehman, S. U. (2020). FCM–SVM based intrusion detection system for cloud computing environment. *Cluster Computing*, *23*(4), 3221-3231.
12. Jaber, A. N., & Rehman, S. U. (2020). FCM–SVM based intrusion detection system for cloud computing environment. *Cluster Computing*, *23*(4), 3221-3231.
13. Attou, H., Mohy-eddine, M., Guezzaz, A., Benkirane, S., Azrour, M., Alabdultif, A., & Almusallam, N. (2023). Towards an intelligent intrusion detection system to detect malicious activities in cloud computing. *Applied Sciences*, *13*(17), 9588.
14. Rathod, G., Sabnis, V., & Jain, J. K. (2024). Intrusion Detection System (IDS) in Cloud Computing using Machine Learning Algorithms: A Comparative Study. *Grenze International Journal of Engineering & Technology (GIJET)*, *10*(1).
15. Samriya, J. K., Kumar, S., Kumar, M., Wu, H., & Gill, S. S. (2024). Machine learning based network intrusion detection optimization for cloud computing environments. *IEEE Transactions on Consumer Electronics*.
16. Shahzad, F., Mannan, A., Javed, A. R., Almadhor, A. S., Baker, T., & Al-Jumeily OBE, D. (2022). Cloud-based multiclass anomaly detection and categorization using ensemble learning. *Journal of Cloud Computing*, *11*(1), 74.
17. Maheswari, K. G., Siva, C., & Priya, G. N. (2023). An optimal cluster based intrusion detection system for defence against attack in web and cloud computing environments. *Wireless Personal Communications*, *128*(3), 2011-2037.
18. Abusitta, A., Bellaiche, M., Dagenais, M., & Halabi, T. (2019). A deep learning approach for proactive multi-cloud cooperative intrusion detection system. *Future Generation Computer Systems*, *98*, 308-318.

19. Nizamudeen, S. M. T. (2023). Intelligent intrusion detection framework for multi-clouds–IoT environment using swarm-based deep learning classifier. *Journal of Cloud Computing*, *12*(1), 134.
20. Elsayed, S., Mohamed, K., & Madkour, M. A. (2024). A comparative study of using deep learning algorithms in network intrusion detection. *IEEE Access*, *12*, 58851-58870.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.