Deep Learning-Driven Heuristic Intrusion Detection System

A Synopsis Report

For

<u>B.Tech Project-I (CO-401)</u>

Abhishek Panda 2K21/CO/19, **Pranay Avnish Kachhap** 2K21/CO/338, **Pritesh Das** 2K21/CO/348

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Delhi Technological University

(Formerly, Delhi College

of Engineering) Bawana

Road, Delhi-110042

UNDER THE SUPERVISION OF:

MR. NIPUN BANSAL

ACKNOWLEDGMENT

- We would like to express our heartfelt gratitude to our respected Mentor **Mr. Nipun Bansal** and our respected Supervisor **Dr. Rajeev Kumar** for their invaluable guidance and support throughout our project. Their expertise and insights have been instrumental in shaping our understanding of the complexities involved in intrusion detection systems. Their constructive feedback and encouragement motivated us to explore innovative approaches and refine our methodologies.

- Thank you for being a constant source of inspiration and for dedicating your time and resources to our learning. We are truly grateful for your mentorship, which has significantly enriched our project experience.

- We, the members of the project team, would like to acknowledge our collective efforts and contributions to the successful completion of our project, **“Deep Learning-Driven Heuristic Intrusion Detection System.”**
    - **Abhishek Panda 2K21/CO/19**
    - **Pranay Avnish Kachhap 2K21/CO/338**
    - **Pritesh Das 2K21/CO/348**

Signature of Mr. Nipun Bansal

TABLE OF CONTENTS

INTRODUCTION

The increasing reliance on internet connectivity has brought about significant advancements in various sectors, but it has also introduced substantial risks related to network security. Intrusion Detection Systems (IDS) have emerged as a critical component in safeguarding networks against unauthorized access and cyberattacks. Traditional IDS methods, which often rely on predefined signatures and rules, struggle to keep pace with the evolving nature of cyber threats. Consequently, there has been a growing interest in leveraging machine learning (ML) techniques to enhance the capabilities of IDS.

Machine learning-driven IDS can analyze vast amounts of network data to identify patterns indicative of malicious activities. These systems can adapt to new types of attacks by learning from historical data, making them more robust compared to traditional methods. Various ML techniques, including deep learning, support vector machines (SVM), and ensemble methods, have been explored to improve the detection accuracy and reduce false alarm rates in IDS.

Despite the promising results, several challenges remain in the implementation of ML-based IDS. One significant issue is the ability to detect low-frequency attacks, which are often overshadowed by more common attack patterns in the training data. Additionally, the dynamic nature of cyber threats necessitates continuous updates to the datasets used for training these models, ensuring they remain effective against new and evolving attack vectors. Moreover, the complexity of ML models can make it difficult for cybersecurity experts to interpret and trust the decisions made by these systems, highlighting the need for explainable AI frameworks.

Recent advancements have focused on optimizing ML algorithms to enhance their performance in IDS applications. For instance, hybrid models that combine different ML techniques have shown improved detection rates by leveraging the strengths of each method. Furthermore, the integration of big data technologies has enabled the processing of large-scale network traffic data, facilitating more accurate and timely intrusion detection.

In conclusion, while machine learning-driven IDS offer significant advantages over traditional methods, ongoing research is essential to address the challenges of low-frequency attack detection, dataset updates, and model interpretability. By continuing to refine these systems, we can develop more robust and reliable solutions to protect against the ever-evolving landscape of cyber threats.

LITERATURE SURVEY

Intrusion Detection Systems (IDS) are critical for maintaining network security by identifying unauthorized access and potential threats. The integration of machine learning (ML) techniques into IDS has been a significant focus of research, aiming to enhance detection accuracy and efficiency.

Overview of Machine Learning Techniques in IDS

Machine learning techniques have been extensively applied to develop IDS, leveraging their ability to learn from data and identify patterns indicative of intrusions. Various studies have explored single, hybrid, and ensemble classifiers to improve IDS performance. A comprehensive review of 55 studies from 2000 to 2007 highlights the evolution and current state of ML techniques in IDS, discussing the achievements and limitations of these approaches.

Detailed Analysis and Challenges

A detailed investigation into ML techniques for IDS reveals that while many methods have been developed, they often struggle to detect all types of intrusions effectively. Issues such as the detection of low-frequency attacks and the limitations of existing datasets are significant challenges. The study suggests improvements and compares different ML techniques in terms of their detection capabilities.

Deep Learning Approaches

Deep learning, a subset of ML, has shown promise in enhancing IDS. A proposed Big Data-based Hierarchical Deep Learning System (BDHDLS) aims to improve detection rates by focusing on unique data distributions within clusters. This approach addresses the limitations of single learning models in handling complex data distributions. Another study explores the use of deep neural networks (DNNs) to develop a scalable and flexible IDS, demonstrating superior performance over classical ML classifiers on various benchmark datasets.

Optimization Techniques

Optimization algorithms have been employed to enhance the performance of ML-based IDS. For instance, support vector machines (SVM) optimized using genetic algorithms (GA), particle swarm optimization (PSO), and ant colony optimization (ACO) have shown improved accuracy in vehicular ad hoc networks (VANETs). Additionally, combining Elman neural networks with robust SVMs has been proposed to reduce false alarm rates and improve detection accuracy.

Robustness and Early Classification

The robustness of ML-based IDS is a critical concern, particularly in detecting new and unknown attacks. A study proposes an early classification approach to prevent intrusions that fall outside the learned data scope, significantly enhancing the robustness of existing ML-NIDS.

Performance Analysis and Feature Selection

Performance analysis of various ML techniques, including Adaptive Boost (AdaBoost), Gradient Boosting, Random Forest, and Decision Tree, indicates that Gradient Boosting outperforms others in terms of F1-score. This suggests its potential for implementing intelligent IDS. Furthermore, a multi-stage optimized ML framework has been proposed to reduce computational complexity while maintaining high detection performance, demonstrating significant improvements in training sample size and feature set size.

Explainability in IDS

The complexity of ML models often leads to a lack of transparency in their decision-making processes. An explainable ML framework using SHapley Additive exPlanations (SHAP) has been proposed to improve the interpretability of IDS. This framework provides both local and global explanations, helping cybersecurity experts understand and optimize IDS decisions.

RESEARCH GAP

1. Dataset Limitations and Benchmarking

- **Outdated Datasets**: Many studies rely on older datasets like KDD Cup 99 and NSL-KDD, which may not reflect current network traffic and attack patterns. There is a need for more up-to-date and diverse datasets to evaluate IDS performance effectively.
- **Lack of Comprehensive Analysis**: There is a lack of detailed analysis comparing the performance of various deep learning algorithms across different publicly available datasets.

2. Model Evaluation and Comparison

- **Controlled Environment Comparisons**: There is a scarcity of objective comparisons of different deep learning models within a controlled environment, especially on recent datasets.
- **Performance Metrics**: Many studies do not use a consistent set of performance metrics, making it difficult to compare results across different research works.

3. Scalability and Real-Time Detection

- **Scalability Issues**: Existing models often struggle with scalability, especially when dealing with large volumes of data and real-time detection requirements.
- **Real-Time Implementation**: There is a need for more research on implementing scalable and hybrid deep learning frameworks that can operate in real-time environments.

4. Model Robustness and Adaptability

- **Dynamic Nature of Attacks**: The continuously evolving nature of cyberattacks requires IDS models to be highly adaptable. Current models often fail to generalize well to new, unseen attack types.
- **Feature Learning**: More research is needed to improve the feature learning capabilities of deep learning models to better capture the nuances of different types of network traffic and attacks.

5. Human Interaction and Automation

- **Human Interaction**: There is a need to reduce the level of required human interaction in the IDS process, making the systems more autonomous.
- **Automation**: Developing models that can automatically update and adapt to new threats without significant human intervention is a critical area for future research.

Conclusion

The primary research gaps in applying deep learning to IDS include the need for updated and diverse datasets, objective model comparisons, scalability and real-time detection capabilities, robustness to evolving threats, and reducing human interaction. Addressing these gaps will be crucial for advancing the effectiveness and reliability of IDS in modern network environments.

PROJECT METHODOLOGY

1. Introduction

Intrusion Detection Systems (IDS) are critical for maintaining the security and integrity of network infrastructures. With the increasing complexity and volume of cyberattacks, machine learning (ML) and deep learning (DL) techniques have emerged as powerful tools to enhance the detection capabilities of IDS. This project aims to develop a robust and scalable IDS using advanced ML and DL methodologies.

2. Data Collection and Preprocessing

Datasets: Utilize publicly available benchmark datasets such as KDDCup 99, NSL-KDD, UNSW-NB15, Kyoto, WSN-DS, and CICIDS 2017.

Preprocessing: Implement data cleaning, normalization, and transformation techniques to handle missing values, noise, and inconsistencies in the datasets. Use feature selection methods like information gain and correlation-based techniques to reduce dimensionality and improve model performance.

3. Model Selection and Training

Model Selection: Explore various ML and DL models including Deep Neural Networks (DNN), Recurrent Neural Networks (RNN), Support Vector Machines (SVM), Random Forest (RF), and Decision Jungle (DJ).

Hyperparameter Tuning: Optimize hyperparameters using techniques such as grid search and random search to enhance model performance. Parameters to tune include learning rate, number of neurons, and epochs.

Training: Train the models on the preprocessed datasets. Implement oversampling techniques to handle class imbalance and ensure robust training.

4. Model Evaluation

Performance Metrics: Evaluate models using metrics such as accuracy, precision, recall, and false alarm rate. Compare the performance of different models to identify the best-performing one.

Cross-Dataset Validation: Validate the trained models on multiple datasets to ensure generalizability and robustness. For instance, a model trained on KDDCup 99 can be tested on NSL-KDD and CICIDS 2017 datasets.

5. Implementation of Hybrid Framework

Hybrid Model: Develop a scalable and hybrid IDS framework, such as the scale-hybrid-IDS-AlertNet, which combines the strengths of different ML and DL models to improve detection accuracy and reduce false alarms.

Early Classification: Implement early classification techniques to detect intrusions before they can cause significant damage. This involves analyzing active sessions and classifying them early in the process.

6. Explainability and Transparency

Model Interpretation: Use SHapley Additive exPlanations (SHAP) to provide local and global explanations for the model's decisions. This helps in understanding the important features and the rationale behind the model's predictions.

Comparison of Classifiers: Compare the interpretations between different classifiers, such as one-vs-all and multiclass classifiers, to optimize the IDS structure and improve decision-making.

7. Deployment and Monitoring

Real-Time Monitoring: Deploy the IDS in a real-time environment to monitor network traffic and host-level events. The system should proactively alert possible cyberattacks and adapt to new threats.

Continuous Learning: Implement a feedback loop to continuously update the model with new data and retrain it to handle evolving attack patterns.

PROJECT WORK TIMELINE

Phase 1: Literature Review and Problem Definition (Weeks 1-4)

Week 1-2: Conduct a comprehensive literature review on existing machine learning techniques for intrusion detection systems (IDS). Focus on single, hybrid, and ensemble classifiers, and their performance on various datasets.

Week 3: Identify the gaps and limitations in current IDS approaches, such as issues with detecting low-frequency attacks and the need for scalable solutions.

Week 4: Define the specific problem your project will address, including the types of attacks to be detected and the machine learning techniques to be employed.

Phase 2: Data Collection and Preprocessing (Weeks 5-8)

Week 5-6: Collect relevant datasets for training and testing the IDS, such as KDDCup 99, NSL-KDD, UNSW-NB15, Kyoto, WSN-DS, and CICIDS 2017.

Week 7: Preprocess the data by cleaning, normalizing, and splitting it into training and testing sets. Address issues like noise elimination and feature selection.

Week 8: Perform exploratory data analysis to understand the data distribution and identify key features for intrusion detection.

Phase 3: Model Development (Weeks 9-14)

Week 9-10: Develop initial machine learning models using various algorithms such as Support Vector Machines (SVM), Decision Trees, and Random Forests. Optimize these models using techniques like Genetic Algorithm (GA) and Particle Swarm Optimization (PSO).

Week 11-12: Implement deep learning models, such as Deep Neural Networks (DNNs), and explore their performance on the preprocessed datasets.

Week 13: Compare the performance of classical machine learning models with deep learning models. Evaluate metrics like accuracy, detection rate, and false alarm rate.

Week 14: Fine-tune the hyperparameters of the best-performing models to enhance their detection capabilities.

Phase 4: Model Evaluation and Validation (Weeks 15-18)

Week 15-16: Validate the models using cross-validation techniques and test them on unseen data to ensure robustness and generalizability.

Week 17: Conduct a detailed performance analysis, focusing on the models' ability to detect various types of attacks, including low-frequency and novel attacks.

Week 18: Use explainability tools like SHapley Additive exPlanations (SHAP) to interpret the models' decisions and improve transparency.

Phase 5: Implementation and Deployment (Weeks 19-22)

Week 19-20: Develop a scalable and hybrid IDS framework that can be deployed in real-time to monitor network traffic and host-level events.

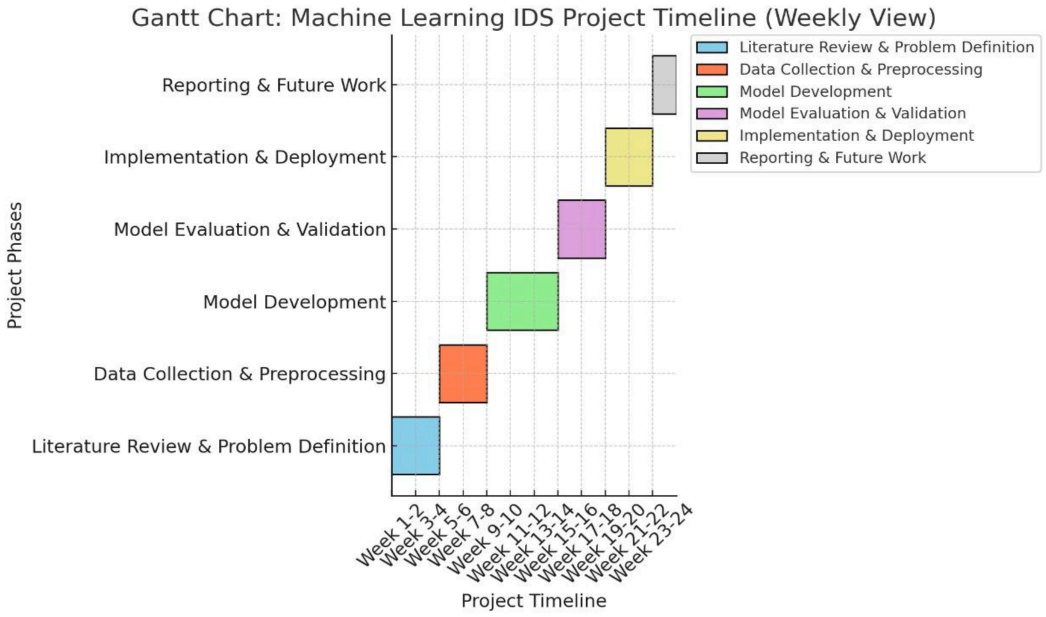
Week 21: Test the deployment in a controlled environment to ensure it performs well under real-world conditions.

Week 22: Finalize the deployment and prepare documentation for the IDS, including user manuals and technical reports.

Phase 6: Reporting and Future Work (Weeks 23-24)

Week 23: Compile the results and findings into a comprehensive project report. Highlight the achievements, limitations, and potential future research directions.

Week 24: Present the project to stakeholders and gather feedback for further improvements. Plan for future work based on the feedback and identified research gaps.



EXPECTED RESULTS

Applying deep learning models such as LSTM, GRU, and other feedback-based neural networks to Intrusion Detection Systems (IDS) can significantly enhance their performance compared to traditional industry methods. Here are the expected results and insights based on recent research:

Performance Improvements

- **Detection Rates and Accuracy**: Deep learning models, particularly GRU and LSTM, have shown high detection rates and accuracy. For instance, a GRU-based IDS achieved detection rates of 99.42% on the KDD 99 dataset and 99.31% on the NSL-KDD dataset, with false positive rates as low as 0.05% and 0.84%, respectively. Similarly, LSTM models have demonstrated superior performance with accuracies around 98.09% and high precision and F1-scores.

- **Comparison with Traditional Methods**: Traditional IDS methods often struggle with zero-day attacks and novel threats. Deep learning models, especially those combining CNN with LSTM or GRU, have shown to outperform traditional methods by effectively learning hierarchical patterns and reducing false positives.

Specific Model Insights

- **GRU vs. LSTM**: Comparative studies indicate that GRU can be more suitable for IDS than LSTM due to its simpler architecture and effective performance. For example, GRU-based models have been found to be more efficient in detecting denial of service attacks with detection rates as high as 99.98%.

- **Ensemble Models**: Combining different deep learning models, such as CNN-LSTM and CNN-GRU, can further enhance IDS performance. These ensemble models have achieved accuracies of 99.7% and 99.6%, respectively, with exceptional F1-scores.

Application in IoT and Real-Time Systems

- **IoT Security**: Deep learning models, including LSTM and GRU, have been effectively applied to IoT intrusion detection, achieving high accuracy and robustness against diverse intrusion types. For instance, a CNN-LSTM-GRU integrated approach achieved near-perfect accuracy in both binary and multi-class classifications.

CONCLUSION

In conclusion, the implementation of deep learning algorithms in Intrusion Detection Systems (IDS) offers a promising approach to enhancing cybersecurity measures. The dynamic and evolving nature of cyber threats necessitates advanced methods capable of adapting to new and unforeseen attack patterns. Deep learning models, such as Deep Neural Networks (DNNs), Recurrent Neural Networks (RNNs), and Deep Reinforcement Learning (DRL) algorithms, have demonstrated superior performance in detecting and classifying intrusions compared to traditional machine learning techniques.

The scalability and flexibility of deep learning models allow them to handle large volumes of data and complex network behaviors effectively. For instance, the proposed scale-hybrid-IDS-AlertNet framework leverages DNNs to monitor network traffic and host-level events in real-time, providing proactive alerts for potential cyberattacks. Similarly, the use of nonsymmetric deep autoencoders (NDAEs) and stacked NDAEs has shown significant improvements in detection accuracy and reduced human interaction requirements.

Moreover, the integration of big data techniques with hierarchical deep learning systems enhances the ability to capture unique intrusion patterns, thereby increasing the detection rate of sophisticated attacks.

Comparative studies and surveys have highlighted the effectiveness of various deep learning approaches across different datasets, including KDD Cup 99, NSL-KDD, CIC-IDS2017, and CIC-IDS2018. These studies emphasize the importance of selecting appropriate datasets and evaluation metrics to benchmark the performance of IDS models accurately.

Overall, the adoption of deep learning techniques in IDS represents a significant advancement in cybersecurity, offering robust and adaptive solutions to counter the ever-increasing threat of cyberattacks. Future research should focus on addressing the remaining challenges, such as improving the interpretability of deep learning models and developing more comprehensive datasets to further enhance the effectiveness of IDS.

REFERENCES

1. Zhong, W., Yu, N., & Ai, C. (2020). Applying big data based deep learning system to intrusion detection. Big Data Min. Anal., 3, 181-195. https://doi.org/10.26599/bdma.2020.9020003.

2. Assis, M. V., Carvalho, L. F., Lloret, J., & Proença, M. L. (2021). A GRU deep learning system against attacks in software defined networks. Journal of Network and Computer Applications, 177, Article 102942. http://dx.doi.org/10.1016/j.jnca.2020.102942.

3. Vitor, G., Daniel M., Mateus K., Vinícius F., Marcos V., Luiz F., and Mario L. (2024). Anomaly and Intrusion Detection Using Deep Learning for Software-Defined Networks: A Survey. Expert Systems with Applications, 256, Article 124982. https://www.sciencedirect.com/science/article/pii/S0957417424018499.

4. Ferrag, M., Maglaras, L., Moschoyiannis, S., & Janicke, H. (2020). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *J. Inf. Secur. Appl.*, 50. https://doi.org/10.1016/j.jisa.2019.102419.

5. Aldweesh, A., Derhab, A., & Emam, A. (2020). Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues. *Knowl. Based Syst.*, 189. https://doi.org/10.1016/j.knosys.2019.105124.

6. Gamage, S., & Samarabandu, J. (2020). Deep learning methods in network intrusion detection: A survey and an objective comparison. *J. Netw. Comput. Appl.*, 169, 102767. https://doi.org/10.1016/j.jnca.2020.102767.

7. Melis, A., Sadi, A. A., Berardi, D., Callegati, F., & Prandini, M. (2023). A systematic literature review of offensive and defensive security solutions with software defined network. IEEE Access, 11, 93431–93463. http://dx.doi.org/10.1109/ACCESS.2023.3276238.