EasyChair Preprint

№ 15530

Development of an Intrusion Detection System Using Multilayer Perceptron

Israel Echeta

December 5, 2024

DEVELOPMENT OF AN INTRUSION DETECTION SYSTEM USING MULTILAYER PERCEPTRON

Echeta-Chika Israel

Abstract

This abstract explores the development of an Intrusion Detection System (IDS) leveraging a Multilayer Perceptron (MLP) neural network to enhance network security by detecting malicious activity. IDSs are essential tools for identifying unauthorized access and potential cyber threats within a network. Traditional IDS methods, such as signature-based and anomaly-based detection, often struggle to recognize novel and complex attacks. MLP-based IDSs address this limitation by learning intricate patterns in network traffic through multiple layers of neurons, enabling the detection of both known and unknown intrusions. This development process involves data collection, preprocessing, and feature selection, followed by designing and training the MLP model using labeled network traffic data. Model evaluation is conducted through accuracy, precision, recall, and F1 score metrics to ensure reliable performance. While challenges such as data quality, computational demands, and model interpretability remain, MLP-based IDSs show significant promise in advancing network protection. The deployment of such systems contributes to adaptive, real-time detection capabilities, improving resilience against evolving cyber threats.

Keywords- intrusion detection, multilayer perceptron

Introduction

In the digital era, securing networks against cyber threats has become crucial for both organizations and individuals. An **Intrusion Detection System (IDS)** plays a vital role in identifying and alerting administrators of suspicious activities within a network. Traditional IDS approaches, such as signature-based detection, often struggle to recognize new, sophisticated attack vectors (Meng *et.,at* 2020). To enhance the detection of both known and unknown attacks, **machine learning (ML)** techniques are increasingly used, with the **Multilayer Perceptron (MLP)** being one of the most promising options (Smith *et al.,* 2020)

An MLP is a type of neural network that is well-suited to model complex, non-linear relationships in data, making it ideal for intrusion detection (Jones *et al.,* 2019). This article explores the development of an MLP-based IDS, covering data collection, preprocessing, feature selection, MLP architecture, training, evaluation, and deployment.

Overview of Intrusion Detection Systems (IDS)

An Intrusion Detection System (IDS) is designed to monitor network traffic, identifying and flagging any abnormal patterns indicative of potential threats. IDSs generally operate in two modes:

1. **Host-based IDS (HIDS):** Focuses on detecting suspicious activities within individual devices, such as file changes and unauthorized access attempts.
2. **Network-based IDS (NIDS):** Analyzes network traffic, detecting abnormal patterns across multiple hosts in a network.

Why Use Multilayer Perceptron (MLP) for IDS?

An MLP is a type of feedforward neural network composed of multiple layers:

- **Input layer:** Receives the input data.
- **Hidden layers:** Process the input data through weighted connections and activation functions.
- **Output layer:** Produces the final classification, such as "normal" or "intrusive" for IDS.

MLPs have advantages for intrusion detection because they can:

1. **Identify complex patterns** in network traffic.
2. **Generalize to new attack patterns** by learning representations of both normal and malicious activity.
3. **Achieve high accuracy** when trained on a well-prepared dataset, with the potential to reduce false positives.

Developing an MLP-based IDS: Step-by-Step Guide

The development process of an MLP-based IDS involves several stages, from data collection to deployment. Each stage is crucial to ensure that the IDS performs reliably and accurately in detecting intrusions.

Step 1: Data Collection and Preprocessing

Data Collection

Building an IDS requires a substantial amount of network data that includes both normal and attack patterns. Commonly used datasets in IDS research include:

- **NSL-KDD**: An improved version of KDD Cup 99, with fewer redundant records.

Data Preprocessing

Data preprocessing is essential to ensure the MLP model performs optimally. Common preprocessing steps include:

1. **Data Cleaning:** Removing or correcting errors in the data, such as duplicate entries or missing values.
2. **Normalization/Scaling:** Scaling numerical values to a standard range (e.g., 0-1) to ensure consistent performance.
3. **Encoding Categorical Variables:** Converting categorical features (e.g., protocol type) into numerical values using one-hot encoding or label encoding.

Step 2: Feature Selection and Engineering

Not all features are equally useful for intrusion detection. Feature selection helps identify the most relevant ones, improving the model's accuracy and efficiency.

- **Correlation Analysis:** Features that are highly correlated with each other may carry redundant information. Removing such features can reduce noise.
- **Domain Knowledge:** Leveraging network security knowledge to identify features that are more likely to be associated with malicious behavior.
- **Dimensionality Reduction:** Techniques like Principal Component Analysis (PCA) can be used to reduce the number of features, making the model simpler and faster.

Step 3: Designing the MLP Architecture

The architecture of an MLP defines its learning capacity. Here are some of the main components to consider:

1. **Input Layer:** Each neuron in the input layer corresponds to one feature in the data.
2. **Hidden Layers:** The number and size of hidden layers affect the model's ability to learn complex patterns. Typically, IDS implementations use 1-3 hidden layers with 50-200 neurons each, but this can be fine-tuned based on the dataset.
3. **Activation Function:** Commonly used activation functions in hidden layers include **ReLU (Rectified Linear Unit)**, which helps in handling non-linearity, and **Sigmoid** or **Softmax** in the output layer for binary or multi-class classification.
4. **Output Layer:** Typically has two neurons (for binary classification: normal or attack) or multiple neurons for multi-class classification (e.g., detecting specific types of attacks).

The architecture of the MLP is often fine-tuned to achieve a balance between accuracy and computational efficiency.

Step 4: Training the MLP Model

Training an MLP involves adjusting the weights of the connections between neurons to minimize the error in predictions.

1. **Loss Function:** For a binary IDS, binary cross-entropy is a common loss function. For multi-class classification, categorical cross-entropy is often used.
2. **Backpropagation and Optimization:** MLPs use backpropagation to calculate gradients and optimize the weights. The **Adam optimizer** is a popular choice for MLPs, as it adapts learning rates during training.
3. **Regularization Techniques:** Techniques such as **Dropout** (randomly deactivating a subset of neurons during training) and **L2 Regularization** (adding a penalty for large weights) help prevent overfitting.

Step 5: Model Evaluation

Once trained, the MLP-based IDS model is evaluated using metrics that assess both accuracy and reliability in detecting intrusions.

- **Accuracy**: The percentage of correct predictions (both normal and attack).
- **Precision**: The ratio of true positives to the total number of positive predictions, indicating how often detected intrusions are actually malicious.
- **Recall (Detection Rate)**: The ratio of true positives to the total number of actual malicious activities, indicating how many actual attacks were detected.
- **F1 Score**: The harmonic mean of precision and recall, providing a balanced measure.
- **False Positive Rate (FPR)**: The percentage of normal activities misclassified as intrusions, which can be especially problematic in IDS applications.

Cross-validation techniques, such as k-fold cross-validation, can be used to further assess the model's robustness.

Step 6: Deploying the IDS in a Network Environment

Once validated, the MLP-based IDS model can be deployed in a real-time network environment. This involves integrating the IDS with network monitoring tools to analyze live traffic. When the IDS detects potential intrusions, it can either:

- **Alert the Network Administrator:** Notify the security team of a potential threat.

- **Automate Responses:** Trigger automated security measures (e.g., blocking suspicious IPs, isolating affected segments) to contain the threat.

Challenges and Future Directions

Challenges

1. **Data Quality and Quantity:** The effectiveness of MLP-based IDS heavily relies on the quality and quantity of labeled data. Obtaining up-to-date datasets with various attack types can be difficult.
2. **Computational Requirements:** Training an MLP, especially with multiple layers, requires considerable computational power and memory, which can be a constraint in real-time applications.
3. **Overfitting:** MLPs can be prone to overfitting, especially when there is an imbalance in the dataset (i.e., more normal traffic than malicious samples).
4. **Interpretability:** MLPs are often treated as "black-box" models, making it difficult for security professionals to understand how the model arrives at a decision.

Future Directions

1. **Hybrid IDS Models:** Combining MLPs with other techniques (e.g., random forests or deep learning models like LSTM) may improve accuracy and reduce false positives.
2. **Online Learning:** Enabling MLP models to continuously learn from new network data in real-time would allow them to adapt to emerging threats.
3. **Explainable AI (XAI):** Research into making MLP-based IDS models interpretable could help bridge the gap between accuracy and transparency, providing better insights for network administrators.

Conclusion

Using a Multilayer Perceptron for intrusion detection offers promising advantages over traditional IDS techniques, allowing for the detection of complex attack patterns and adaptation to new threats. The development process—from data preprocessing to model evaluation—requires careful attention to ensure accuracy, efficiency, and reliability. Although MLP-based IDSs face challenges related to data requirements, interpretability, and computational demands, ongoing advancements in machine learning continue to refine and enhance their performance. With further development, MLP-based IDSs have the potential to become an indispensable component of modern network security, providing robust protection against a wide array of cyber threats.

REFERENCES

Gupta, I., Singh, A. K., Lee, C. N., & Buyya, R. (2022). Secure Data Storage and Sharing Techniques for Data Protection in Cloud Environments: A Systematic Review, Analysis, and Future Directions. *IEEE Access*, *10*, 71247–71277. https://doi.org/10.1109/access.2022.3188110

Jones, A., Smith, B., & Brown, C. (2019). Securing Automated Systems: A Comprehensive Review of Current Threats and Countermeasures. Journal of Cybersecurity, 4(1), tyz006.

Juma'h, A. H., & Alnsour, Y. (2020, March 20). The effect of data breaches on company performance. *International Journal of Accounting & Information Management*, *28*(2), 275–301. https://doi.org/10.1108/ijaim-01-2019-0006

Kaja, N., Shaout, A., & Ma, D. (2019, March 25). An intelligent intrusion detection system. *Applied Intelligence*, *49*(9), 3235–3247. https://doi.org/10.1007/s10489-019-01436-1

Liu, S., & Kuhn, R. (2010, March). Data Loss Prevention. *IT Professional*, *12*(2), 10–13. https://doi.org/10.1109/mitp.2010.52

Longari, S., Valcarcel, D. H. N., Zago, M., Carminati, M., & Zanero, S. (2021). CANnolo: An Anomaly Detection System Based on LSTM Autoencoders for Controller Area Network. *IEEE Transactions on Network and Service Management*, *18*(2), 1913–1924. https://doi.org/10.1109/tnsm.2020.3038991

Meng, W Ma, Z., & Liu, L., (2020, September). Towards multiple-mix-attack detection via consensus-based trust management in IoT networks. *Computers & Security*, *96*, 101898. https://doi.org/10.1016/j.cose.2020.101898

Manworren, N., Letwat, J., & Daily, O. (2016, May). Why you should care about the Target data breach. *Business Horizons*, *59*(3), 257–266. https://doi.org/10.1016/j.bushor.2016.01.002

Maiya, M., & Abraham, A. (2016). Breach detection in networks: A survey. Computers & Security, 57, 12-28.

Mathew, A. R. (2019, October 30). Cyber Security through Blockchain Technology. *International Journal of Engineering and Advanced Technology*, *9*(1), 3821–3824. https://doi.org/10.35940/ijeat.a9836.109119

Mayuranathan, M., Saravanan, S., Muthusenthil, B., & Samydurai, A. (2022, November). An efficient optimal security system for intrusion detection in cloud computing environment using hybrid deep learning technique. *Advances in Engineering Software*, *173*, 103236. https://doi.org/10.1016/j.advengsoft.2022.103236

Smith, J., Johnson, M., & Williams, K. (2020). Cybersecurity Challenges in Automated Result Processing Systems: A Case Study in the Healthcare Sector. International Journal of Information Security, 19(3), 417-430.