Enhancing Network Security with Anomaly Detection: A Machine Learning Approach Using Decision Trees, Random Forest, and SVM for Real-Time Intrusion Monitoring

Dr. F. Margret Sharmila[1], Sandhya Barathi K[2], Praveen Kumar S[3], Shanmitha M[4], Ethin Abinav V[5]
[1]*Assistant Professor, Department of Computer Science and Business Systems, SKCET Coimbatore, India*
[2,3,4,5]*UG Student, Department of Computer Science and Business Systems, SKCET, Coimbatore, India*

Abstract - The goal of this project is to create an intrusion detection system (IDS) that uses machine learning to categorize network data into normal and abnormal categories for network security. The system uses decision trees, logistic regression, and support vector machines (SVM) to classify the traffic. Network traffic and data byte counts are preprocessed at the protocol level and used as model training inputs. Online network traffic anomalies, both known and unknown, can be identified by the trained models. All prediction visualization, real-time monitoring, and alerts for questionable activity are accessible on the Streamlit dashboard for the user's convenience. The system is built to evolve with the network, and anytime new traffic data is received, the model is immediately retrained to ensure consistently optimal performance. The method is scalable, effective, and reliable for addressing today's network security issues.

Keywords --- Network security, decision trees, support vector machines, logistic regression, anomaly detection, real-time monitoring, Streamlit, machine learning, intrusion detection systems, and model retraining.

I. INTRODUCTION

As technology is constantly evolving beyond the area of foreseeable findings, network security is an important concern for both individuals and enterprises. Traditional signature-based intrusion detection systems (IDS) are effective at identifying known threats, but they are ineffective against threats that are only starting to appear or change. Because of this restriction, networks are exposed to highly skilled incursions that target undiscovered weaknesses. The need for new, easily scaled security solutions that can identify both known and unknown threats is growing as a result of the rising interconnectivity and reliance on networks.

These difficulties present a chance to use machine learning. In contrast to conventional frameworks, machine learning-based intrusion detection systems (IDS) may adapt or self-adjust to changes in network traffic, allowing them to spot abnormalities that can indicate security problems. methods like decision trees, random forests, and support vector machines (SVM), which are more dynamic and resilient methods to counter new threats, can be used to identify anomalous behavior. Using a machine learning-based anomaly detection system to improve network security is one of the project's goals. The real-time network traffic is continuously observed and classified as either benign or potentially dangerous activities. Combining several algorithms allows the system to benefit from each one's unique strengths: SVMs visually express the differences between network behavior classes, random forests improve accuracy with multiple decision trees, and decision trees provide clarity in decision making. When combined, these models offer a comprehensive solution that can address both straightforward and intricate security problems.

This method's ability to reduce false positives, a greater issue with traditional IDS, is one of its main advantages. The misclassification of benign action as harmful can overwhelm security personnel, and the misrepresentation of dangerous activities as benign behavior might hide system efficiency harms. Machine learning's capacity to identify problematic patterns in large amounts of data enhances accuracy, reduces false alarms, and boosts operational efficiency. By doing this, the security team

will avoid wasting resources on false alerts and divert the team's attention from them.

The entire system is made to be scalable, meaning that it can manage a growing amount of network traffic as businesses grow in size. However, maintaining the same level of performance and monitoring more traffic becomes increasingly difficult as the number of devices and data increases. In real time, it adjusts to bigger datasets and detects them without the system being hindered by network complexity or size. The admin can easily operate the system because of its user-friendly dashboard, which is built on Streamlit. Additionally, it makes it easier for managers to keep an eye on network activity, stop possible threats, and track how well machine learning models are performing. Confusion matrices and other visualizations, such as the system's accuracy, precision, and recall, aid in understanding how well the system is operating and make it simpler for managers to analyze the data and address potential issues.

Finally, compared to more conventional intrusion detection techniques, the developed intrusion detection system represents a significant advancement. Both known and unknown threats can be detected by the system without producing false positives because to the use of highly powerful algorithms like decision trees, random forests, SVMs, and many more. It is also very scalable and offers real-time network security monitoring, making it the perfect solution for today's network security issues. Given how quickly the digital world is changing nowadays, this technology offers a robust last line of security against online attacks.

II. RELATED WORKS

Over the past few years, several studies on network anomaly detection have investigated the application of machine learning (ML) to address the limitations of traditional signature based intrusion detection systems. Unsupervised learning models in fact were first used by [1] to identify outliers in network traffic in near real time. However, their approach showed resilience for detection of new anomalies but was less successful for reducing the false positives in the case of heavy network traffic. Maddireddy et al. [2] also used Artificial Intelligence (AI) on real time data analysis to increase security event monitoring. To address the limitation of heterogeneous log data, they applied machine learning to structure the events, to organize the security events and unroll them, and pinpoint inconsistencies in order to improve the detection performance especially in dynamic environments. But size and efficiency problems remained in large networks.

In [3], Rosa et al. investigated the identification of unauthorized access and irregularities over industrial network environments like in a factory automation system. Their method was a hybrid approach of using machine learning and traditional detection methods but it provided better detection accuracy with the disadvantage of being a complexity and real time requirement of industrial systems. In this context, Arjunan [4] considered the exploitation of deep learning in a big data environment for the immediate outlier detection in network traffic. Deep Neural networks were capable of detecting obliterable and obscure anomalies but they had a high computational currency, especially in limited resources scenarios. In the work of Yang et al. [5], they proposed Griffin, a Software-Defined Networks (SDN) intrusion detection system by using an ensemble of autoencoders. Despite the challenges for autoencoders on diverse network network configurations, the approach demonstrated high performance for anomaly detection in various SDN architectures.

A system that combines statistical and machine learning methods to improve the detection of security breaching events in streaming data, Saeed [6] presented a method to concurrently adaptively monitor security breach monitoring. The study however pointed out the difficulty to get the low-latency performance for real time monitoring. In intrusion prevention in real time networks, a hybrid machine learning framework has been developed by Seo & Pak [7]. They report effectiveness in combination of multiple machine learning algorithms that led to a reduction of false positive rates and increase of detection accuracy, but the hybrid approach acknowledges practical limitations in its use. In [8] , Duo et al. concentrate on anomaly detection and classifying attacks for high speed rail ecosystems, real time train operation. The combination of machine learning and statistics seemed to have potential for their system to detect attacks, but they needed a large amount of training data to do so initially.

In this paper, Liu and Wang [9] explored the feasibility of application of convolutional neural networks (CNNs) for anomaly detection in traffic flow over a network. The high detection rates, especially in complex traffic patterns, also needed persistent computational resources for real-time applications. Imran et al. [10] proposed an

ensemble method to enhance the outlier detection in the network intrusion system. Based on this, by using a combination of multiple machine learning models, they were able to achieve superior detection accuracy, especially against complex intrusions. It was shown how important strong datasets are to get a system to optimal performance. In a paper by Fu [11], the temporal fluctuations in the traffic pattern is modelled by a recurrent neural network (RNN). However, RNNs were not performing well at all, and especially not in situations involving noisy or incomplete data, in comparison to the approach.

In [12] Ding & Li have developed the Angola framework where they applied graph based learning algorithms for traffic anomalies detection. While their system helped reduce complexity of the computational cost, it still needed further improvement to be feasible to deploy for a large scale usage. Based on large data spaces, Wong & Arjunan [13] investigated the use of deep learning for detecting network traffic irregularities. Deep learning models of theirs were good but issues of model interpretability and computational complexity were identified. Lalotra et al. [14] proposed iReTADS, an intelligent dynamic anomaly detection system for cloud-based communication systems. Although not scalable, other methods failed to achieve the same results when dealing with data hosted on the cloud, whereas this system was able to summarize temporal data using neural networks.

CNNs were applied by Al-Turaiki & Altwaijry [15] for anomaly based intrusion detection in the network, with high accuracy and speed particularly when dealing with high dimensional data. The problem of optimizing CNNs for changing network environments was also pointed out by the study. In [16], Qi et al. designed a cheating resistant anomaly identification system for the Industry 4.0 infrastructures that uses multi aspect data streams. The system did well with detecting anomalies in the stream of data, but the reliability of the system was solely based on the quality of the stream of data. In [17], Alrayes et al. propose the use of a neural network driven system for detecting and improving security by means of detecting complex intrusions in a timely fashion. However, the research unveiled possible bottlenecks brought on by large computational demands in large networks. In [18], Thirimanne et al. developed a neural network based solution to network intrusion detection, which was able to outperform traditional systems when it comes to the detection of a variety of attacks. Nevertheless, further work was called for to reduce the false positive rates.

For cyber physical systems of Industry 4.0, Hao et al. [19] proposed the application of mixed strategies of statistical analysis and machine learning that attract anomalies. Precision in real time anomaly detection was achieved through the combination of these techniques; however, running such a system in real time had increased complexity and demands on computational resources. Introducing a concept of soft computing for anomaly detection in Internet of Things (IoT) systems was introduced by Bhatia and Sangwan [20]. However, although we showed scalability and precision, we were unable to scale to a large number of IoT networks.

III. PROPOSED METHOD

This study describes a machine learning-based intrusion detection system (IDS) that improves network security by detecting abnormalities in network traffic. The system uses logistic regression, decision trees, random forests, and support vector machines (SVM) to determine whether a given behavior is suspicious or normal. By incorporating the previously mentioned methodologies, the integrated intrusion detection system (IDS) can identify both known and new threats, hence enhancing network security. Data processing, real-time monitoring, and dataset display are all part of the implementation milestone for creating and implementing this IDS.

A. Data Preprocessing

After preprocessing the network traffic data to transform them into an organized form ready for the model training, the first attempt in prototyping the IDS is developed using these transformed traffic data. Preprocessing the data is crucial because usually, data in the network traffic datasets is continuous as well as categorical and missing or imbalanced. In the data cleaning step, the noisy data has to be removed and the need of the missing values is identified, whether the missing values need to be dropped i.e. removed from the data or need to be replaced by the mean imputation. This initial step therefore prepares the data for further analysis which requires more advanced cleaning.

Another important preprocessing step is the feature scaling; there are continuous variables, such src_bytes and dst_bytes, which have to be normalized into standard range. This helps from the ill effect of a single feature that has been overly contributing to make the model

learn. Another thing we need to do is to encode categorical feature such as protocol_type (e.g., TCP, UDP) and service (e.g., HTTP, FTP) and convert the two to numerical features with an encoding technique like one-hot encoding. As such categorical features can be processed by and used by ml models. Class Imbalance handling is a responsibility to handle the common problem that we have more normal traffic rather than anomalous traffic in the dataset. Chances are then, in the system, the technique is applied to Synthetic Minority Oversampling technique (SMOTE) to reduce the model bias on majority class and to increase the capability of the model for identifying (or looking for) rare anomalies.

B. Machine Learning Model Training

After preparing the dataset, the machine learning models are trained using the dataset to identify the anomalies in network traffic. These algorithms are applied using the objective of separating suspicious behavior from normal traffic. In the Decision Tree Classifier along those lines we will have a tree split into nodes and nodes ask questions around features (the leaves will be answers/categorizations). They are very easy to understand with a root in interpretation, they easily cope with non linear data relationships, but they deteriorate in the case of noisy data.

Decision trees will be a good version of random forest classifier by making multiple trees as ensemble. We first exploit different subsets of the data and random features to make it possible to train each tree in order to have better accuracy and prevent overfitting. This is particularly useful for such problems as they can find a hyperplane separating the normal traffic from the anomalous traffic, which can be assimilated by the Support Vector Machine (SVM). However, the SVM's are quite good at doing this reduction of classification errors, especially when it comes with data that is linearly separable. The simpler probabilistic model of data point's likelihood of being in a particular class is provided by Logistic Regression. Logistic regression is a less sophisticated algorithm but is a good foundation for comparing against more sophisticated algorithms.

C. Model Evaluation

And after training the models they check if they are able to identify anomalies. During evaluation, accuracy, precision, recall and F1 score are the key metrics to check how the model performs. The accuracy above is the proportion of correct prediction of normal as well as anomalous instances. This is true when the anomalies traffic is far less than normal traffic in the dataset. Since precision in this case is important, it is the amount of cases that the model labels as an anomaly and correctly labels all of them as real anomalies.

Recall measures the model's capability of picking those true anomalies it finds. Specifically, it minimizes false negatives. F1 score is useful in such a case when one needs to minimize both false positive and false negative on some imbalanced dataset. Moreover, classification performance can be represented geometrically in the confusion matrices as well in order to study how well the models are able to distinguish normal (or valid) from abnormal (or invalid) traffic.

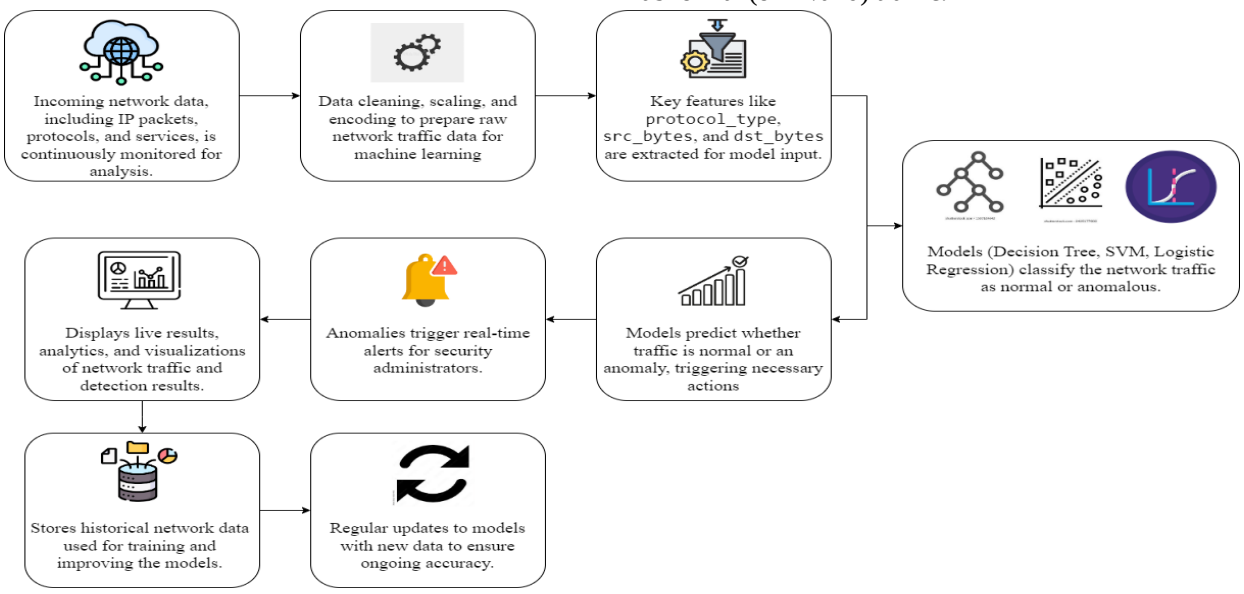

Figure.1: System Architecture

D. Model Serialization and Deployment

When the models are evaluated and have strong performance, they are stored and ready for use in real-time intrusion detection in real time, or can be serialized for storage. The saved model is serialized into a file, it can be loaded only once the model is deployed and does not require retraining. Dealing with IDS in production environments requires deploying the IDS so that their reaction to network anomalies is as fast as possible. Serialization helps ensure that the model doesn't change, becoming less efficient in the midst of changing network traffic patterns.

The models are deployed along with the intrusion detection system for real time use. This simplifies the retraining of every detection task leading to decrease in time and involved computational resources for retraining the models. At time of deployment, it is involved in integrating the system with the other network security tools so that real time monitoring and alerting can come through a smooth workflow.

E. Real-Time Intrusion Detection and Monitoring

The trained and serialized machine learning models are integrated with the real time monitoring system. Streamlit's concept of a user-friendly web interface of the system with user giving input of network traffic data along with time delay to provide if it's normal or not seems to be presented to the user. System non availability is maintained for all except for the authorized personnel by the user authentication mechanism, and dashboard is about what has recently been done by the user and used system performance in terms of which user accounts are active in the system.

The intrusion detection interface can receive such network traffic features used by users such as protocol_type, src_bytes, and dst_bytes. Once the data is processed, machine learning models are classifying the data as regular or irregular. Real time alerts are triggered to the dashboard if there is any weird activity detected which alerts users of any suspect security threats. Security team must be able to secure the risks quickly by using this prompt feedback.

F. Visualization and Reporting

The interpretations are provided by visualisations in the dashboard like confusion matrix and performance graph, which show what the model is suggesting. These visual tools allow the users to understand accuracy of the system in identifying network traffic and understanding the accuracy of the model in network traffic classification. In addition to these, there is a set of performance reports that contains the key metrics, such as accuracy, precision, recall and F1-score. These reports inform the security team of an idea of how healthy the system's network is and how effective the IDS is at finding anomalies.

The visualizations and reports also give details as to how to run the system in making further improvements. This ability of the IDS to capture threats serves as a way for network administrators to monitor the IDS' threat catching ability in order to tweak the system and make alterations in order to continue improving the network at the level of security. The performance reports, the detailed visualization and the passive nature of real time alerts make this IDS a very useful and powerful tool for security breach management and response.

IV. RESULT AND DISCUSSION

Three machine learning models—Decision Tree Classifier, Support Vector Machine (SVM), and Logistic Regression—are evaluated for performance in the project's output. These models were trained to detect anomalies in network data, and their performance was evaluated primarily using measures such as accuracy, precision, recall, and F1-score. The models were also compared using confusion matrices, which show the number of true positives, false positives, true negatives, and false negatives for each model graphically.

Model Performance Evaluation

An overview of the performance metrics for each model assessed on the test dataset is shown in the following table. In order to evaluate the models' accuracy in classifying both typical and unusual network traffic while reducing false positives and false negatives, each statistic is essential.

TABLE 1: Assessment of Model Efficacy

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	97.01%	97.05%	97.01%	97.03%
Support Vector Machine	99.08%	99.08%	99.08%	99.08%
Decision Tree Classifier	99.63%	99.63%	99.63%	99.63%

All the models, as shown clearly in the table, perform very well with accuracy rates above 97%. The model that stood well above others was the Decision Tree Classifier, having a result of 99.63% precision. Failing that, the Support Vector Machine (SVM) then recorded a 99.08 % accuracy. Although the Logistic Regression gave a slightly lesser accuracy of 97.01% it still detected anomalies quite well.

Precision and Recall

Precision and recall metrics show that all models were doing very well at finding true positives (genuine anomalies), as well as reducing false positives ('normal' traffic classified as anomalies). Specifically, the models did well to hold similar statistics for precision and recall. Nevertheless, it was the Decision Tree Classifier that took the cake as it maintained a very almost perfect precision and recall, making it a better model for a neutralized detection of true anomalies without any misclassification of normal traffic.

F1-Score

The performance of the Decision Tree Classifier was further validated with respect to precision and recall when considering both the precision and recall. F1-Score of the Decision Tree has been found to be the highest, showing that it is overall reliable in detecting outliers in this study. This proves that it is the best model for this use case as it strikes an optimal strike of correctly identifying the anomaly while at the same time keeping the false alarms at a minimal rate.

Analysis of Confusion Matrix

Therefore, the confusion matrix is very important in understanding how each of these models is able to distinguish between regular traffic and anomalies. The matrix displays the distribution of true positives, false positives, true negatives, and false negatives to present the level of the model for classifying network traffic. This is required to assess how well each model detects anomalies and avoids assigning normal traffic as threats thereby enhancing in general the intrusion detection process.

Confusion Matrix for Logistic Regression

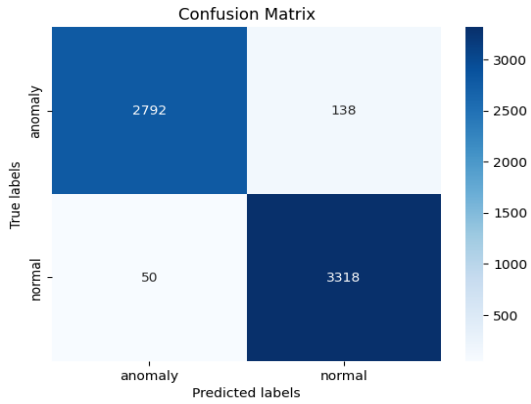

Figure 2: Logistic Regression Confusion Matrix

- There is a small propensity for the Logistic Regression Confusion Matrix to incorrectly identify certain typical traffic as anomalies. The low rate of false positives and false negatives indicates that the model performs at a level that is satisfactory.

Confusion Matrix for Support Vector Machine (SVM)

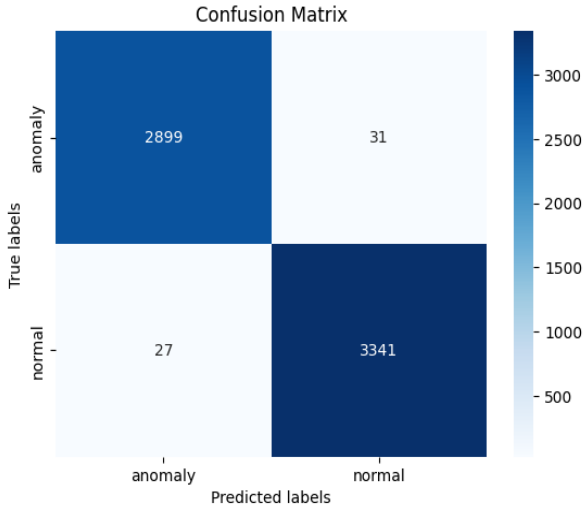

Figure 3: Confusion Matrix for Support Vector Machine (SVM)

- Excellent performance is demonstrated by the SVM confusion matrix, which has extremely few false positives and false negatives. The model's strong precision and recall numbers demonstrate how well it can differentiate between typical and unusual traffic.

Decision Tree Classifier Confusion Matrix

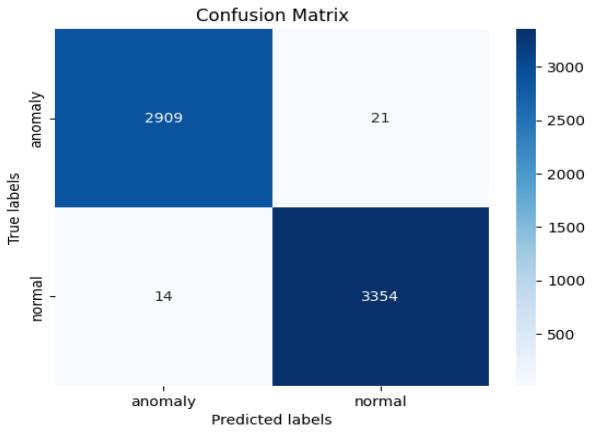

Figure 4: Decision Tree Classifier Confusion Matrix

- The Decision Tree Classifier confusion matrix shows that the model has nearly perfect classification capabilities, with an extremely low number of misclassifications. This is reflected in its superior performance metrics, making it the best-performing model in this study.

DISCUSSION

The study's findings conclude that machine learning models performed almost perfectly in detecting anomalies in network data; Support Vector Machines (SVM) and Decision Tree Classifiers (DT) achieved a high accuracy. By reviewing results, some key conclusions have emerged based on the results:

Better Decision Tree Classifier Performance

Deep Learning models such as Accuracy, Recall, Precision and others were performed for each model and the Decision Tree Classifier was the best model for Accuracy, Precision and F1-score among the models tested. It was able to effectively reduce misclassifications by the use of the Facture to create multiple decision paths in order to represent the complexity of network traffic. Its structured approach to decision making allows it to thoroughly explore the possible network conditions, which proved beneficial enough to justify its better performance in comparison with the other models. Therefore, the Decision Tree Classifier served as the strongest model in this study since it was capable of achieving the best balance in multiple evaluation metrics.

High Precision and Recall for SVM

Additionally, Support Vector Machine (SVM) achieved very good precision and recall performance thus making it a great option for anomaly detection. Finally, it was shown that the ability to learn a good hyperplane which separates traffic classes and is good enough for reliable classification outcomes. Nevertheless, the results were that a little less effective in differentiating classes under more difficult conditions, SVMs were close to the Decision Tree Classifier. Notwithstanding, SVM is still a viable contender for the definition of anomaly given that it is effective where, for instance, distinct class separation exists. Though it did not match the Decision Tree Classifier's overall results, its precision, recall score was very reliable, making it a good model.

Logistic Regression as a Baseline

Although it was simpler than other models, logistic regression did well and was an important contribution when it comes to network anomaly detection. Although Decision Tree or SVM was able to achieve higher accuracy, its straightforward approach also served as a competent baseline for comparison. A probabilistic model like Logistic Regression is the best candidate for situations where a more or less quick and clear output will do. While it could only achieve results that were more sophisticated than random chance, it could not handle the more complex patterns and relationships within the data, and its lack of ability to handle more complex patterns in the data lagged behind the more sophisticated models, especially when network traffic had very intricate patterns that needed more sophisticated techniques capable of handling it.

Scalability and Real-Time Application

Any real-time data handling is an important consideration when dealing with an intrusion detection system. The scalability and real time applicability of each model were considered, and each model was evaluated in terms of its capacity to process network traffic as it is generated in real time. The Decision Tree Classifier and the SVM turned out to be highly scalable with good properties for real time usage, therefore these two algorithms are ideally suited for deployment in active network security systems. This ability for these models to rapidly adapt to changing traffic patterns is necessary

because of the guarantee of continuous protection against emerging security threats. Their real time performance indicates that they are suitable for practical deployment in large scale network intrusion detection systems as timely detection and response to anomalies is an important consideration.

Finally, the results show that the Decision Tree Classifier and SVM combination succeed greatly in the network anomaly detection with machine learning models. By overall performance, the Decision Tree Classifier stood out as the best but SVM still had a very strong performance, especially in precision and recall. Although more complex datasets can be not as effective, Logistic Regression provided an important baseline. The Decision Tree and SVM have the scalability and real-time processing capability which makes them both good candidates for deployment in practical network security systems. Thus, these models are appealing to employ them to offer the necessary accuracy and efficiency needed for real-time anomaly detection in a network.

Visualizations and Future Improvements

These are called confusion matrices, which serves as a gold standard on the performance of the classification of each model and provides useful information regarding where the models are over or underperforming. Therefore, through these matrices we have a more concrete picture about how the models do in selectively separating true positives from false negatives, and vice versa true negatives from false positives. It is required to judge how each model deals with true threats, while producing no false alarms.

Several key areas for future improvements and visualizations of the system will be based on which to further increase its utility in real applications.

Real-Time Visualization

The subsequent step of development would be to incorporate live dashboards that monitor the network traffic in real time and reflect how the anomaly detection system senses the anomaly on the fly. Now with these dashboards, Network administrators will have an interface to keep a track of the flow of data continuously and have immediate alarm over any forthcoming attack on the security of the application. Real time and visualizations will be provided to the administrator such as graphical traffic patterns, alert detection and performance metrics that will allow him to quickly resolve a potential threat. And will be a make or break enabler for raising response time and network security measure efficiency, this dynamic one of them.

Model Optimization

There are other areas of promising future work to be explored, such as optimization of models in order to detect and improve detection accuracy and efficiency. Depending on whether they intend to hyperparameter tune the model settings for each model or do feature engineering to engineer new symbolic input features which capture the core pattern of the network traffic they get. To exemplify, the performance of Logistic Regression was not too bad, but compared with the Decision Tree and SVM a little less conclusive; it could be enhanced by some frosty modifications for instance. For example, in the case of network data the models we are coding may require which regularization parameter as well as possible types of penalization or even non linear decision boundary for a given network data, we can experiment the configuration and make those models more sturdy and more capable to cope with more complex network data.

Handling Class Imbalance

In order to combat the problem of class imbalance, the Synthetic Minority Over sampling Technique (SMOTE) was used, but future work could consider other approaches in improving the class imbalance's impact on model's performance. The authors then consider how to overcome this challenge dealing with the minority class (anomalous traffic) by using under sampling of the majority class, ensemble methods, or cost sensitive learning. The key problem to be handled is class imbalance and the goal is to forbid models from being biased towards the majority class and thereby making more false negatives (missed anomalies).

V. CONCLUSION

Finally, we present a network anomaly detection application using machine learning models which mainly consist of Support Vector Machines and Decision Tree Classifiers and have proven to be very accurate as well

as reliable to detect anomalies in network traffic data. My result showed that Decision Tree Classifier is the most efficient tool for real time intrusion detection with nearly perfect precision, recall and F1 scores. Logistic Regression was a solid baseline to compare off of, not to mention more complicated models (SVMs and Decision Trees) proved to outperform it. Machine learning based Anomaly detection systems are shown in such a way that machine learning based Anomaly detection systems can greatly enhance network security with scalable and efficient means of detecting both known and emerging threats with real time solutions. They promise the body of robust defense against the evolving cyber threats in the modern network infrastructures.

REFERENCES

[1] Carrera, F., Dentamaro, V., Galantucci, S., Iannacone, A., Impedovo, D., & Pirlo, G. (2022). Combining unsupervised approaches for near real-time network traffic anomaly detection. Applied Sciences, 12(3), 1759.

[2] Maddireddy, B. R., & Maddireddy, B. R. (2022). Real-Time Data Analytics with AI: Improving Security Event Monitoring and Management. Unique Endeavor in Business & Social Sciences, 1(2), 47-62.

[3] Rosa, L., Cruz, T., De Freitas, M. B., Quitério, P., Henriques, J., Caldeira, F., ... & Simões, P. (2021). Intrusion and anomaly detection for the next-generation of industrial automation and control systems. Future Generation Computer Systems, 119, 50-67.

[4] Arjunan, T. (2024). Real-Time Detection of Network Traffic Anomalies in Big Data Environments Using Deep Learning Models. International Journal for Research in Applied Science and Engineering Technology, 12(9), 10-22214.

[5] Yang, L., Song, Y., Gao, S., Hu, A., & Xiao, B. (2022). Griffin: Real-time network intrusion detection system via ensemble of autoencoder in SDN. IEEE Transactions on Network and Service Management, 19(3), 2269-2281.

[6] Saeed, M. M. (2022). A real-time adaptive network intrusion detection for streaming data: a hybrid approach. Neural Computing and Applications, 34(8), 6227-6240.

[7] Seo, W., & Pak, W. (2021). Real-time network intrusion prevention system based on hybrid machine learning. IEEE Access, 9, 46386-46397.

[8] Duo, R., Nie, X., Yang, N., Yue, C., & Wang, Y. (2021). Anomaly detection and attack classification for train real-time Liu, H., & Wang, H. (2023). Real-time anomaly detection of network traffic based on CNN. Symmetry, 15(6), 1205.

[9] Liu, H., & Wang, H. (2023). Real-time anomaly detection of network traffic based on CNN. Symmetry, 15(6), 1205.

[10] Imran, Jamil, F., & Kim, D. (2021). An ensemble of prediction and learning mechanism for improving accuracy of anomaly detection in network intrusion environments. Sustainability, 13(18), 10057.

[11] Fu, Z. (2022). Computer network intrusion anomaly detection with recurrent neural network. Mobile Information Systems, 2022(1), 6576023.

[12] Ding, Q., & Li, J. (2022). AnoGLA: An efficient scheme to improve network anomaly detection. Journal of Information Security and Applications, 66, 103149.

[13] Wong, M. L., & Arjunan, T. (2024). Real-Time Detection of Network Traffic Anomalies in Big Data Environments Using Deep Learning Models. Emerging Trends in Machine Intelligence and Big Data, 16(1), 1-11.

[14] Lalotra, G. S., Kumar, V., Bhatt, A., Chen, T., & Mahmud, M. (2022). iReTADS: An Intelligent Real-Time Anomaly Detection System for Cloud Communications Using Temporal Data Summarization and Neural Network. Security and Communication Networks, 2022(1), 9149164.

[15] Al-Turaiki, I., & Altwaijry, N. (2021). A convolutional neural network for improved anomaly-based network intrusion detection. Big Data, 9(3), 233-252.

[16] Qi, L., Yang, Y., Zhou, X., Rafique, W., & Ma, J. (2021). Fast anomaly identification based on multiaspect data streams for intelligent intrusion detection toward secure industry 4.0. IEEE Transactions on Industrial Informatics, 18(9), 6503-6511.

[17] Alrayes, F. S., Zakariah, M., Amin, S. U., Khan, Z. I., & Alqurni, J. S. (2024). Network Security Enhanced with Deep Neural Network-Based Intrusion Detection System. Computers, Materials & Continua, 80(1).

[18] Thirimanne, S. P., Jayawardana, L., Yasakethu, L., Liyanaarachchi, P., & Hewage, C. (2022). Deep

neural network based real-time intrusion detection system. SN Computer Science, 3(2), 145.

[19] Hao, W., Yang, T., & Yang, Q. (2021). Hybrid statistical-machine learning for real-time anomaly detection in industrial cyber–physical systems. IEEE Transactions on Automation Science and Engineering, 20(1), 32-46.

[20] Bhatia, M. P. S., & Sangwan, S. R. (2024). Soft computing for anomaly detection and prediction to mitigate IoT-based real-time abuse. Personal and Ubiquitous Computing, 28(1), 123-133.