ITU-ML5G-PS-006: Intrusion and Vulnerability Detection in Software-Defined Networks (SDN)

Nasik Sami Khan
Department of Computer Science
University of Regina, Canada
nku618@uregina.ca

Md. Shamim Towhid
Department of Computer Science
University of Regina, Canada
mty754@uregina.ca

Md Mahibul Hasan
Department of Computer Science
University of Regina, Canada
mhr993@uregina.ca

***Abstract*—The transition from conventional architectures to Software Defined Networks (SDNs) has revolutionized network management and control in contemporary networking. Nonetheless, the centralization of network control within SDNs has introduced a significant security risk, necessitating the implementation of robust intrusion detection systems. This paper examines intrusion detection within SDN-enabled networks, concentrating on the development of multiclass classifiers capable of identifying an array of intrusion types. A comprehensive dataset combining actual user data from an SD-WAN environment with established datasets is provided to facilitate research. The sample set includes Normal flow data, DDoS flow data, Malware flow data, and web-based flow data, among other types. Using machine learning techniques, our research aims to facilitate the development of effective intrusion detection models, thereby contributing to the protection of SDN-based networks against a wide range of threats.**

***Index Terms*—SDN, DDoS, Intrusion Detection, Machine Learning**

I. Introduction

With the advent of Software Defined Networks (SDNs) in recent years, the networking landscape has undergone a dramatic transformation. This paradigm shift has ushered in unprecedented flexibility and efficiency in network management, made possible by the centralization of control through a single entity — the SDN controller. While this architectural change offers numerous benefits, it has also introduced a major security concern: the potential vulnerability of the centralized controller, which, if compromised, could have catastrophic repercussions for the entire network.

In the context of SDNs, the security of the central controller is crucial to the network operations' integrity. Any unauthorized access or nefarious activity directed at the controller can have far-reaching effects on the network's functionality and security. In response, the implementation of effective intrusion detection systems (IDS) has become mandatory.

Traditional intrusion detection methods frequently rely on binary classifications, labelling instances as normal or malevolent. Nonetheless, the changing threat landscape necessitates a more nuanced approach. Intrusions can manifest as Distributed Denial-of-Service (DDoS) attacks, malware infiltrations, and web exploits, among others. Therefore, the development of multiclass classifiers capable of categorizing these diverse intrusion varieties is crucial.

In this paper, we conduct a thorough investigation of intrusion detection systems that are tailored for SDN-enabled networks. Our research capitalizes on a novel dataset that combines actual user data sourced from an SD-WAN environment with well-established datasets representing various intrusion scenarios. The dataset contains distinct sample types, including Normal flow data, DDoS flow data, Malware flow data, and web-based flow data. By leveraging the power of machine learning techniques, we aim to create intrusion detection models that transcend binary categorizations, thereby contributing to the protection of SDN architectures against a vast array of security threats. Through our research, we hope to promote a deeper understanding of intrusion detection in the context of SDN and to provide practical insights into the development of multiclass classifiers that have the potential to bolster network security in this era of dynamic networking.

The remainder of the paper is structured as follows: in Section II, we examine previous research on IDS. Section III describes the methodology of our research, while Section IV evaluates the results. Section V concludes the paper and discusses prospective research.

II. Related Work

In an effort to protect Software Defined Networks (SDNs) from intrusions, a vast amount of research has been devoted to the creation of effective intrusion detection systems (IDS). This section examines a selection of seminal research papers that advance intrusion detection techniques, particularly in SDN-enabled environments.

Maxime Labonne [1] explored anomaly-based network intrusion detection using machine learning techniques. The study highlighted the ability of machine learning algorithms to detect anomalies in network behaviour, providing insights into enhancing network security by recognizing deviations from the norm.

In SDN environments, Junhong Li [2] focused on the detection of Distributed Denial-of-Service (DDoS) attacks. Li proposed an innovative method for effectively identifying DDoS attacks that combines dense neural networks, autoencoders, and the Pearson correlation coefficient. The study demonstrated how neural network architectures and correlation metrics can be utilized to improve the accuracy of intrusion detection systems.

Naveen Bindra and Manu Sood [3] investigated the impact of feature selection techniques on the efficacy of machine learning models designed for the detection of DDoS attacks. Their research emphasized the importance of preprocessing stages in enhancing the precision of intrusion detection models. The research provided vital insights into the optimization of machine learning-based intrusion detection systems through the evaluation of a variety of feature selection methods.

K. Muthamil Sudar and P. Deepalakshmi [4] introduced a novel intrusion detection system based on flow analysis and customized for SDN environments. Their research utilized hybrid machine learning techniques to identify intrusions in SDN environments. This study highlighted the importance of employing flow-based analysis and hybrid machine learning models to address the unique security challenges that SDN architectures present.

Zhen Yang et al. [5] conducted a comprehensive systematic literature review, surveying anomaly-based network intrusion detection methodologies and datasets. Their research, which was published in Computers and Security, Volume 116 (2022), offered a comprehensive overview of cutting-edge techniques and data sets in this field. This contribution is a valuable resource for researchers pursuing a comprehensive comprehension of the landscape and methodologies surrounding anomaly-based network intrusion detection.

Collectively, the cited works emphasize the importance of machine learning in developing resilient intrusion detection systems in SDN environments. Incorporating techniques such as anomaly detection, neural networks, and hybrid machine learning, researchers have made significant advances in enhancing the security of SDNs against a variety of intrusion scenarios. In conjunction with Zhen Yang et al.'s systematic literature review, these studies contribute to a greater understanding of the evolving threat landscape and the corresponding defences in the dynamic domain of SDN-based networking.

III. Research Methodology

A. Dataset

The provided dataset, which is central to a competition, contains 1.78 million rows, each of which is characterized by 77 distinct columns. A crucial component of the dataset is its solitary labelled column, which serves as the competition's output variable. The structure of this dataset reflects the intent to use machine learning to predict or classify outcomes based on the interaction of the various independent features. This dataset is notable for its pronounced class disparity, a situation in which some classes in the labelled column are considerably underrepresented in comparison to others. This disparity can present difficulties for machine learning algorithms, potentially resulting in biased model performance in which the majority class dominates prediction accuracy. This imbalance necessitates the application of specialized techniques to ensure that all classes are treated fairly. Some classes within the labelled column exhibit an extremely sparse representation, with only a minimal two-digit total, adding to the complexity. This scarcity heightens the need for cautious model management, as the limited number of instances of these classes may hinder the algorithm's ability to generalize accurately. Table I contains overall summary.

TABLE I
Dataset Summary

	Class	Count
0	BENIGN	1,432,050
1	DoS Hulk	145,575
2	PortScan	100,125
3	DDoS	80,656
4	DoS GoldenEye	6,484
5	FTP-Patator	5,000
6	SSH-Patator	3,714
7	DoS slowloris	3,651
8	DoS Slowhttptest	3,464
9	Bot	1,238
10	Web Attack Brute Force	949
11	Web Attack XSS	410
12	Infiltration	22
13	Web Attack Sql Injection	12
14	Heartbleed	6
	Total	1,783,356

B. Feature Selection

The process of feature selection was instrumental in refining the original 77 features of the dataset. Using the robust Random Forest (RF) classifier, we evaluated each attribute's significance in relation to the target variable with great care. Through this exhaustive evaluation, we identified and retained a subset of 28 carefully curated features with significant predictive power. To validate and strengthen our choices, we subjected these 28 characteristics to Principal Component Analysis (PCA), a well-known dimensionality reduction technique. Remarkably, the results derived from both RF and PCA exhibited remarkable coherence, further validating the effectiveness of the feature subset we chose. This consolidated subset not only improves the predictive ability of our models, but also assures an essential level of interpretability for extracting insights from complex datasets. Table II lists the prominent features.

TABLE II
Significant Feature List

Features	
Total Length of Fwd Packets	Flow Packets/s
Total Length of Bwd Packets	Fwd IAT Mean
Fwd Packet Length Mean	Max Packet Length
Bwd Packet Length Min	Packet Length Variance
Bwd Packet Length Std	Avg Fwd Segment Size
Fwd IAT Total	Subflow Fwd Bytes
Fwd Header Length	$\text{Init}_W in_bytes_backward$
Packet Length Std	Flow IAT Max
Average Packet Size	Fwd IAT Std
Fwd Header Length.1	Packet Length Mean
$\text{Init}_W in_bytes_forward$	PSH Flag Count
Fwd Packet Length Max	Avg Bwd Segment Size
Bwd Packet Length Max	Subflow Bwd Bytes
Bwd Packet Length Mean	Fwd IAT Max

C. Data Prepossessing

We selected the desired features, which would be impactful for the model to learn the pattern for accurate predictions. The data is then cleaned by using a function, which removes rows containing missing or infinite values to ensure data quality. It is used to convert the data to only numeric values, except for the Label column. It is vital for accurate analysis and modeling processes, resulting in enhanced data reliability. The column names had blank characters, which were cleaned. We separated the data classes into 3 sub-groups, depending on their data samples. Benign, DoS Hulk, PortScan, and DDoS were separated in the High Samples class. Which had more than 80000 rows per class. We then reduced the Benign sample from 1432050 to 150000 samples, as the extra data was not significantly contributing to model performance. The Mid-Samples class consisted of DoS GoldenEye, FTP-Patator, SSH-Patator, DoD Slowloris, DoS Slowhttptest classes, where each of them had at least 3000 samples. The high and mid-sample classes were less challenging to classify. The other classes consisted of less than 1000 samples, in fact, three of them had less than 50 samples. We then augmented five of these classes using the synthetic minority over-sampling technique (SMOTE) [6] approach. It works by generating synthetic samples for the minority class by interpolating between existing instances and their nearest neighbors. This helps create a more balanced class distribution, preventing the model from being biased towards the majority class and improving its ability to accurately classify the minority class. Table III illustrates the number of samples for each of these classes before and after the augmentation.

TABLE III
LOW-SAMPLE OF CLASSES DISTRIBUTION BEFORE AND AFTER AUGMENTATION

Attack Type	Original Count	After SMOTE
Web Attack – Brute Force	949	949
Web Attack – XSS	410	1210
Infiltration	22	272
Web Attack – Sql Injection	12	262
Heartbleed	6	256

We then concatenated three subsets of data and consolidated it into a single training file with a relatively similar number of data samples among three subsets of data. Then the data preprocessing is done by scaling the feature values to have zero mean and unit variance. The labels are prepared to convert textual class labels into numerical values. The training dataset is then split into a smaller training set and a validation set, where 10% of the data is allocated for validation while ensuring that the class distribution is preserved in both sets. These preprocessing steps are essential to ensure consistent and well-prepared data for training, validation, and testing, ultimately leading to accurate and reliable model performance evaluation.

D. Model Architecture

In the model construction phase, a sophisticated ensemble approach is employed. It involves combining multiple individual models to create a more accurate and robust predictive model [7]. This approach enhances accuracy by leveraging diverse models that capture different data patterns, reduces overfitting by promoting generalization, and handles complex relationships in the data effectively [8]. In ensemble learning, ”training” involves developing individual models, each trained on distinct subsets of data or employing varied algorithms to capture diverse data patterns. These models collectively form the ensemble. The ”meta model,” also called the ensemble model, integrates the predictions from the individual models to make a final prediction. This amalgamation harnesses the strengths of each model, yielding enhanced accuracy and robustness.

Ensemble methods that are implemented in our experiment is Random Forest, AdaBoost, and XGBoost. Multiple bagging and boosting algorithms can be combined to create a generic heterogeneous model architecture. In our study, each classifier is meticulously instantiated with algorithm-specific parameters, such as random state and objective, which shape the way each classifier constructs its decision boundaries and responds to optimization goals. During the training of base models, the essence of supervised learning takes place. The training set is partitioned into smaller segments of training subsets. These subsets, encompassing features and corresponding labels facilitate the iterative fitting process. The base models learn from samples in the training data. They adjust their weights to make accurate prediction about labels for new validation data. These predictions are combined with the original features to create a new set of data. The meta-model then learns how to best use these combined predictions and features from previously used three base models, to make a final prediction for unseen data. This improves prediction accuracy by bringing together different models' insights and making a more informed decision. In our experiment, the Random Forest classifier was used as a meta-model.

Random Forest is a robust meta learning model due to its ability to combine the predictive power of multiple decision trees while addressing their limitations. It constructs an ensemble by training numerous decision trees on different data subsets and features, resulting in diverse models. Through majority voting, it aggregates these trees' predictions, reducing overfitting and enhancing generalization. Moreover, the random feature selection for each tree adds an element of variability, mitigating the risk of a single tree dominating the ensemble. This approach leads to robust and accurate predictions, making Random Forest an effective choice for complex datasets like our dataset.

IV. RESULT ANALYSIS

In our study, we thoroughly examined how well our ensemble model performed in detecting different types of network traffic and security threats. We tested the model on two different sets of data: one to check its accuracy during training,

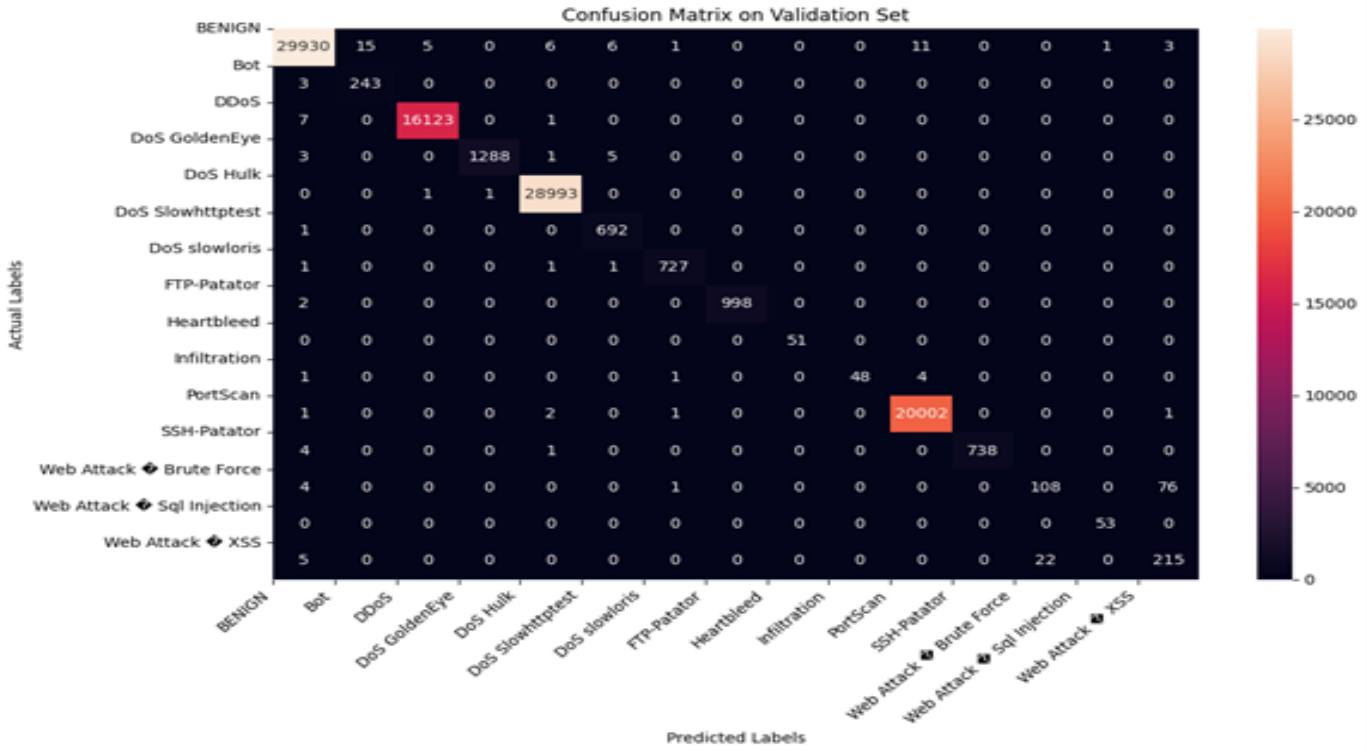

Fig. 1. Validtion Set Result

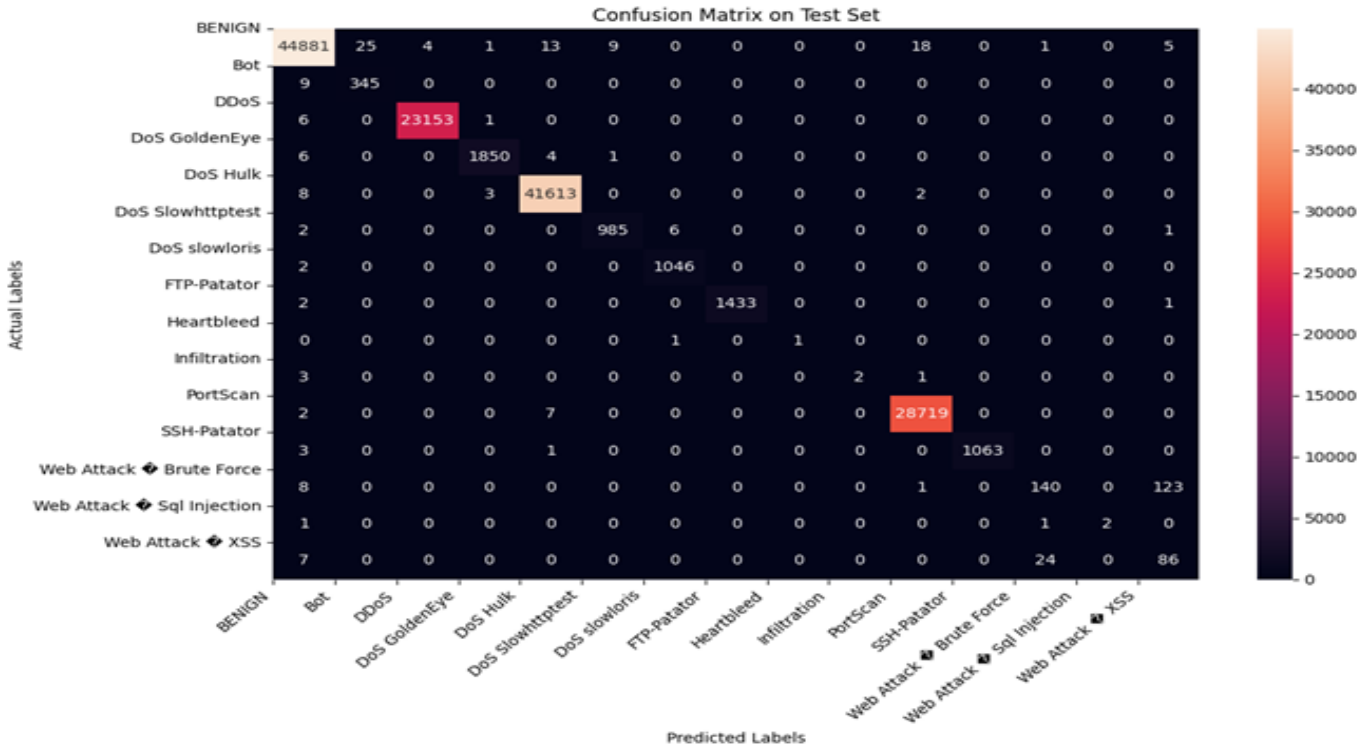

Fig. 2. Test Set Result

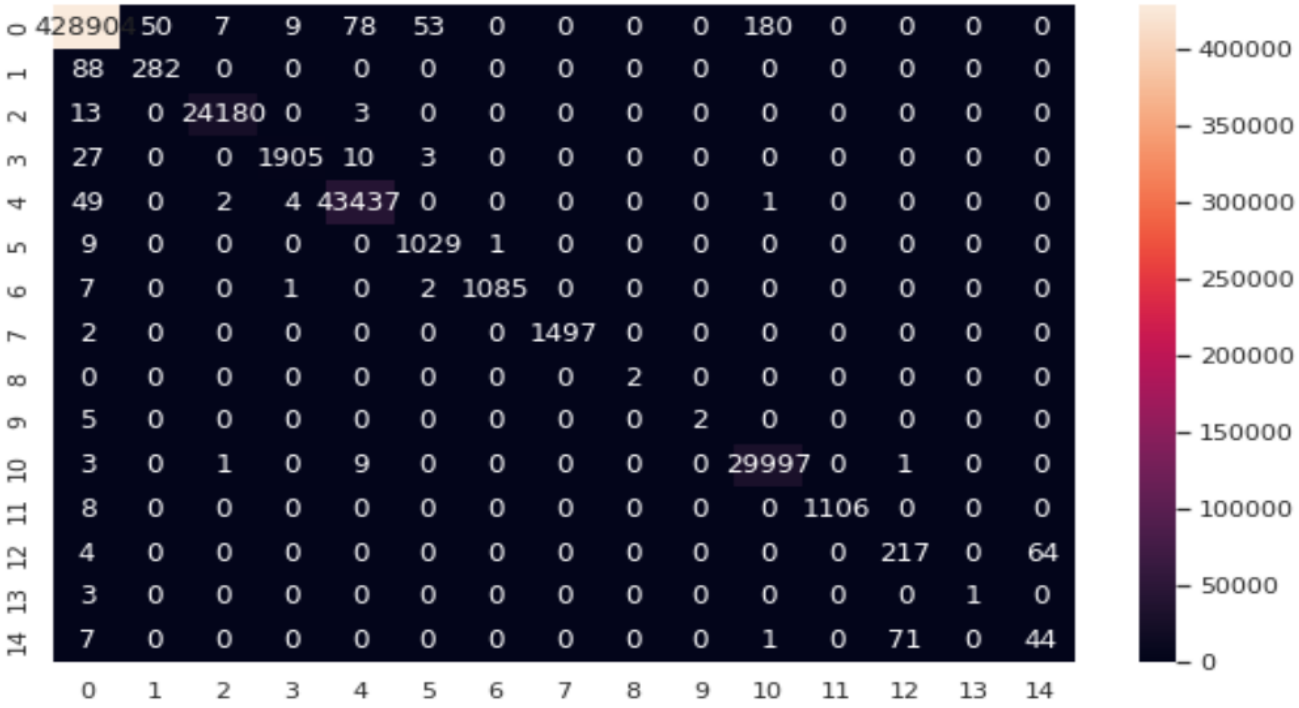

Fig. 3. Confusion Matrix of Hybird Approach [8]

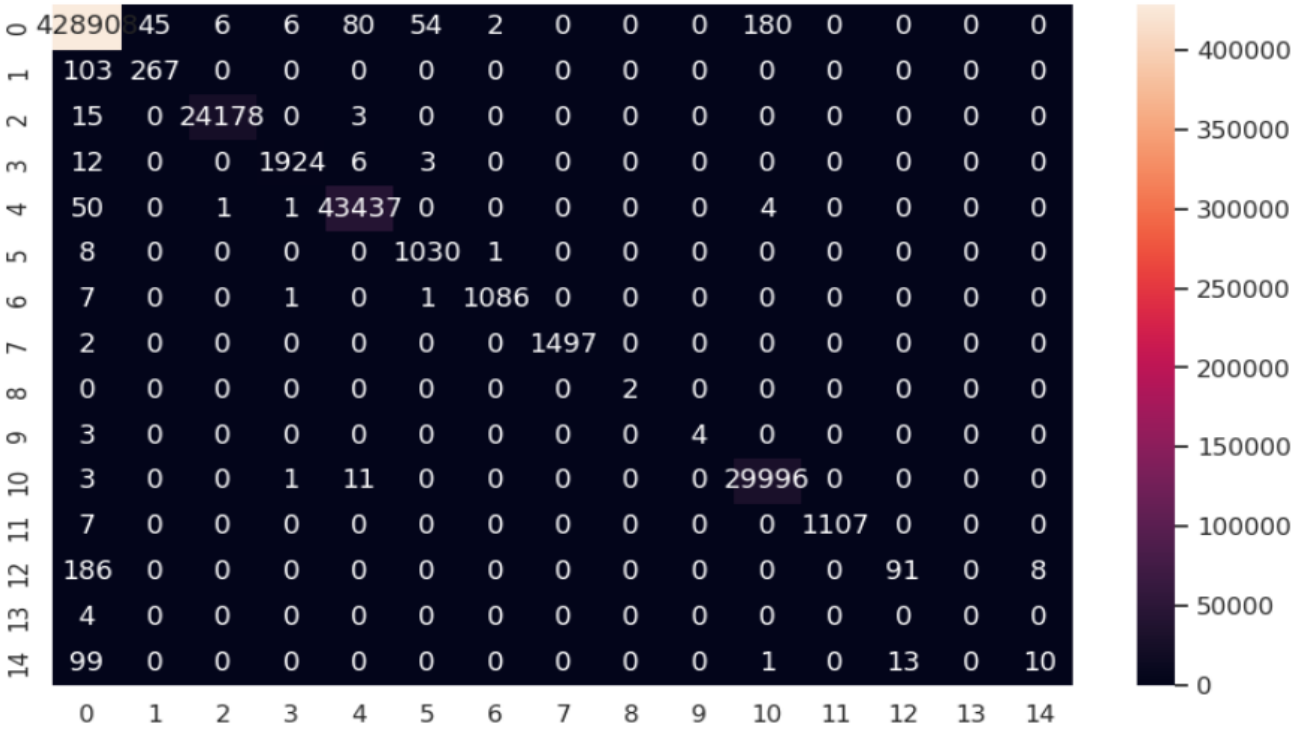

Fig. 4. Confusion Matrix of RF

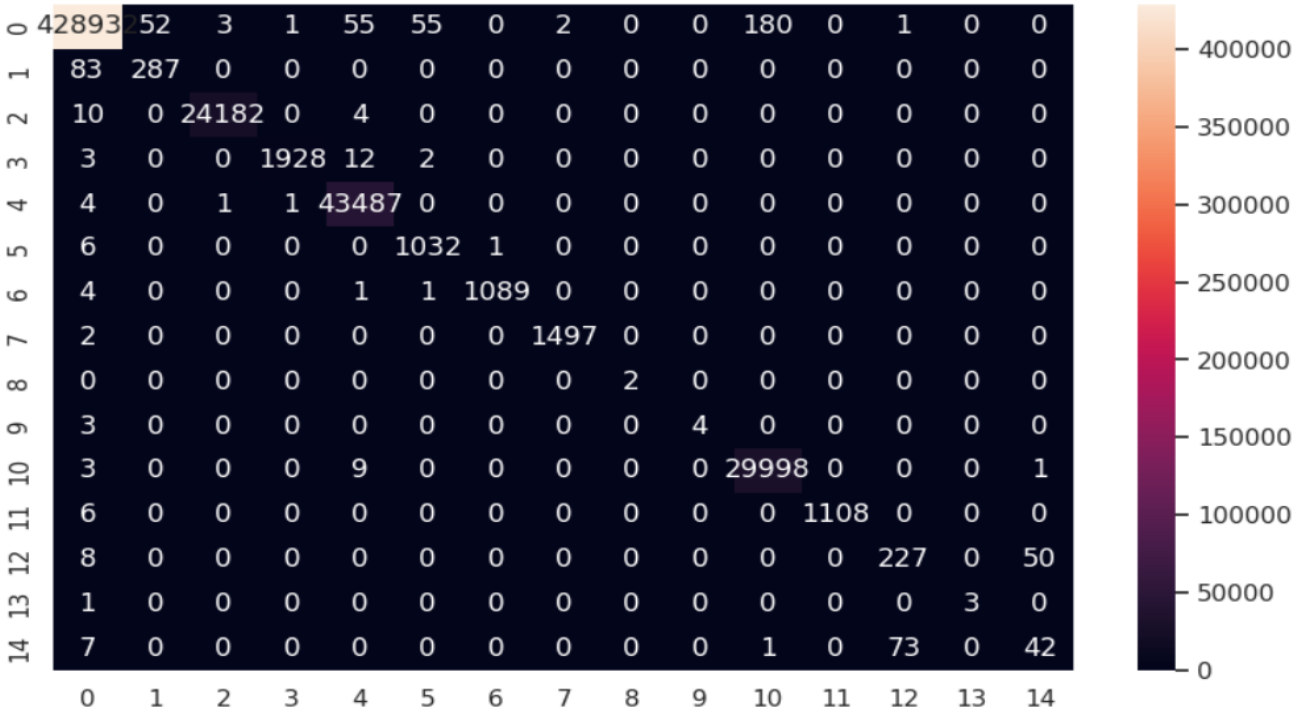

Fig. 5. Confusion Matrix of XGBoost

and the other to see how well it works on new, unseen data. The results were quite promising. The average F1-score for 5-fold cross validation on the validation set was 97.77%. Our model was very good at correctly identifying common types of network traffic and well-known attacks. It also did a good job of maintaining its accuracy when dealing with new data, which is an important sign of a robust model.

However, we did notice some challenges. When it came to detecting rare security threats or specific types of attacks, the model struggled. This tells us that we might need more data or different techniques to handle these rare cases effectively. We solved the class imbalance issue by augmenting the minority class data, but due to a very small number of original samples in the minor classes, the model was not able to robustly classify them in test data. We also found that the model had a bit of trouble when dealing with complex attack patterns. For instance, it was not as accurate in identifying certain web-based attacks.

A common theme throughout our analysis was finding the right balance between precision and having a good recall score. While the model did well in most of the cases, there's still room for improvement in this balance. Figure 1 illustrates the confusion matrix of the ensemble model on validation data and Figure. 2 depicts the result for test data. In the training, we used 5-fold stratified cross-validation to ensure model robustness.

To sum up, our ensemble model showed promising results in detecting network traffic and security threats. By focusing on its strengths, addressing challenges with rare cases and complex attacks, and fine-tuning some aspects, we can make it even better at its job. This analysis gives us important insights to improve the model and make it more effective in real-world situations.

As per the requirement of the competition, we need to compare the performance of our approach with three existing methods. We found the approach mentioned in [8] interesting as the authors propose a hybrid models that combines CNN and Random forest for intrusion detection. We implemented the approach mentioned in [?]. The customized regularized function from [8] is also implemented. The result of this approach on validation dataset are shown in Figure 3. The other two baselines are supervised Random Forest and Xgboost models. We select these two models as these two models are commonly used as comparison in the literature. The confusion matrix of these two models are shown in Figure 4 and 5, respectively.

V. Conclusion

In terms of future work, several approaches for enhancing the performance and robustness of our ensemble model can be applied. First, the handling of rare classes, such as "Heartbleed," "Infiltration," and "Web Attack – Sql Injection," demands specialized meticulous attention. Second, an exploration of alternate feature engineering can be investigated. Investigating them by identifying and incorporating more salient aspects of network traffic data, we can enhance the model's overall performance. For imbalanced data, rectifying

the skewed class distributions can empower the model to learn more effectively from underrepresented classes, thus allowing a more equitable learning process. Third, the optimization of threshold values and hyperparameters stands as a key consideration. Future research could delve into comprehensive experiments to fine-tune threshold settings, achieving an optimal balance between precision and recall across various classes. Moreover, further refinement of the ensemble model architecture by using different classifiers and incorporating neural networks can be explored. Techniques like class-specific weighting or the exploration of diverse ensemble configurations could be used to address performance disparities among specific classes. Class distribution-wise separate training models can be implemented in future to find out the performance based on different subsets of data. For instance, larger sample data can be trained by neural networks, and smaller data can be trained by tree-based models. Such architectures and voting mechanisms can be further investigated.

REFERENCES

[1] Maxime Labonne. Anomaly-based network intrusion detection using machine learning. Cryptography and Security [cs.CR]. Institut Polytechnique de Paris, 2020. English. NNT : 2020IPPAS011. Tel 02988296.

[2] Junhong Li. DETECTION OF DDOS ATTACKS BASED ON DENSE NEURALNETWORKS, AUTOENCODERS AND PEARSON CORRELATION COEFFICIENT. Dalhousie University. April 2020.

[3] Naveen BINDRA, Manu SOOD. Evaluating the Impact of Feature Selection Methods on the Performance of the Machine Learning Models in Detecting DDoS Attacks. ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY. Volume 23, Number 3, 2020, 250–261.

[4] K.Muthamil Sudar, P.Deepalakshmi. Flow Based Intrusion Detection System for Software Defined Networking using Hybrid Machine Learning Technique. International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-9 Issue-2S2, December 2019.

[5] Zhen Yang, Xiaodong Liu, Tong Li, Di Wu, Jinjiang Wang, Yunwei Zhao, Han Han, A systematic literature review of methods and datasets for anomaly-based network intrusion detection, Computers & Security, Volume 116, 2022.

[6] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," Journal of artificial intelligence research, vol. 16, pp. 321–357, 2002.

[7] B. N. Narayanan and V. S. P. Davuluru, "Ensemble malware classification system using deep neural networks," Electronics, vol. 9, no. 5, p. 721, 2020.

[8] K. Abbas, M. Afaq, T. Ahmed Khan, A. Rafiq, and W.-C. Song, "Slicing the core network and radio access network domains through intent-based networking for 5G networks," Electronics, vol. 9, no. 10, p. 1710, 2020.

[9] M. S. ElSayed, N.-A. Le-Khac, M. A. Albahar, and A. Jurcut, 'A novel hybrid model for intrusion detection systems in SDNs based on CNN and a new regularization technique', Journal of Network and Computer Applications, vol. 191, p. 103160, 2021