Anomaly-Based Intrusion Detection for Software-Defined Networks Through an Ensemble Learning Approach

Muhammad Usman Younus[1,2], Muhammad Faisal[3], Kalsoom Safdar[3,4], Muhammad Shoaib[3]

[1]Department of Computer Science & IT, Baba Guru Nanak University, Nankana Sahib, Pakistan.

[2]Ecole Doctorale Matheematiques, Informatique, Telecommunication de Toulouse, University of Toulouse (III), Paul Sabatier, Toulouse, France.

[3]Department of Computer Science & IT, University of Jhang, Jhang Pakistan.

[4]Faculty of Electronic Engineering and Technology, Universiti Malaysia Perlis, 02600 Arau, Perlis, Malaysia.

***Correspondence**: usman1644@gmail.com, kalsoomsafdar11@gmail.com

Citation | Younus. M. U, Faisal. M, Safdar. K, Shoaib. M, "Anomaly-Based Intrusion Detection for Software-Defined Networks Through an Ensemble Learning Approach", IJIST, Vol. 8 Issue. 1 pp 89-101, January 2026.

Received | December 04, 2025 **Revised |** December 31, 2025 **Accepted |** January 04, 2026 **Published |** January 09, 2026.

Software Defined Network (SDN) is a paradigm shift in the wired network; the approach separates the control plane from the data plane. However, such architectural development becomes problematic for IDS due to the inherent limitations of the signature-based models, which cannot keep pace with the increasingly emerging threats. In this paper, the performance of the ensemble learning models is analyzed for anomaly-based intrusion detection of SDNs. we use general and IoT network intrusion datasets to compare different machine learning methodologies for constructing IDS., with the test of the model on SDN network intrusion datasets. We also investigate three Support Vector Machines (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN) algorithms in their base formats and improved versions using ensemble methods, showcasing that the combination significantly enhances detection accuracy, precision, and recall essential for achieving robust SDN security. The weighted-average precision and recall are used to report the performance metrics in order to consider the imbalance across attack categories. In particular, the ensemble model achieved an accuracy of 0.99 on general and IoT datasets, and 0.97 on SDN datasets, with weighted precision of 1.00 and weighted recall of 0.99 and 0.97, respectively. The proposed approach of the ensemble method has the advantage of learning a lot of types of traffic patterns, and the false positive rate as well as the false negative rate are comparatively low, however, this comes at the cost of increased training time and computational complexity. In general, our study reveals the effectiveness and applicability of ensemble learning toward critical security issues related to SDNs.

Keywords: SDN, Random Forest, SVM, Machine Learning, KNN.

Introduction:

Software-defined Networking (SDN) is the most important network innovation that has come up in the market, and it has redefined how network architectures are developed. Typical broadcast networks employ extensive static and distributed hardware, making them highly inflexible and slow. On the other hand, SDNs put the network control plane in a separate entity distinct from the data plane to enhance network management and provide proactive and efficient control features for the network elements [1][2]. This separation proves useful in increasing flexibility for managing a network since administrators can more easily fine-tune parameters related to available network resources, the configuration of those resources, and even shifting priorities in harnessing those resources for a network [3]. The centralized control model is useful for management since it provides control over all activities, but it is a focal point that can be targeted by an intruder. Hence, security solutions to prevent threats affecting SDNs should be put in place [4].

A promising solution in that direction is provided by Anomaly-based IDS. For example, in the case of signature-based IDS, the primary goal of anomaly-based IDS is the discovery of certain patterns of activity on a network, but in this case, the patterns are compared to the normal patterns expected of an ideal network [5]. This approach is particularly beneficial when dealing with SDNs because the network is frequently changing, and there might be new and unknown attack types originating from such a network. As IDS focuses on analyzing network traffic, anomaly-based IDS can discover new threats after defining normal traffic patterns and then detecting deviations from them. [6]. There are additional improvements when ensemble learning techniques are incorporated into the process of anomaly detection [3]. Ensemble learning, which involves the use of multiple learning models to come up with better predictions and hence better detection, takes advantage of all the positive attributes of all the different algorithms to arrive at a precise one[7][8]. The rationale for this research is as follows: the existing anomaly detection methods have certain limitations, but SDNs' potential for security enhancement can be unlocked through the ensemble learning approach [9].

Software Defined Network (SDN) signifies a revolution in the architecture and control protocols of the network because of the requirement for dynamic, adaptive, and elastic conducting structures. More precisely, SDN decouples the control plane from the data/forwarding plane. of the network and makes it more centralized. Conventional network devices like routers and switches contain both the control and the data planes and hence have complicated and at times very rigid network structures. On the other hand, the concept of SDN has an added control layer called the SDN controller that is separate from the actual Layer 3/4 devices that are involved in forwarding data packets. It also allows the different departments, such as the network administrator, to program the behavior of the Network and organize the utilization of the different resources in the Network by just implementing a mechanism of policies across the different areas of the Network through one point of reference. Therefore, SDN enhances the ability to control the network proactively in the sense that real-time changes can be made, and tasks related to managing the congestion and distribution of traffic within a network can be handled more efficiently [10].

Security in Software Defined Networks (SDNs) is a critical consideration since the architecture and operation of SDNs present new opportunities as well as threats to the network structures. As shall be postulated in this work, security on traditional networks is realized on every device through security features, while the same on an SDN is centralized through the SDN controller. Although this centralization helps in effective control of the network and management of policies, it comes with a lot of risks, given the fact that there is a major central point that hackers may find easy to attack. Any compromise or malfunction of the SDN controller will have ramifications on the entire network; thus, it is not an exaggeration to say that it can result in catastrophes such as data leakage, service interruption, or intrusion on restricted resources. Hence, the protection of the SDN controller and its connections is highly sensitive to the general security posture of the resulting network.

Software Defined Networks (SDNs) are one of the most promising technologies in current networking infrastructure since they provide a flexible approach to managing the networks, predominantly because of their ability to separate the control layer from the forwarding layer. This separation means incredible amounts of work in practice, especially when it comes to security issues and specifically, intrusion detection and prevention. Conventional IDS fails to operate effectively in a programmable and fluid networking environment, as it is epitomized by SDNs, because the messy structure of networks and high levels of traffic make it difficult to distinguish between typical patterns and signs of an intrusion. This research, therefore, seeks to come up with an anomaly-based IDS for SDNs through the limitations of conventional IDS [11][12]. Here, there are three models, which include Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Random Forest models, employed, where the idea is to take advantage of the best attributes of each for improving detection precision while at the same time reducing the frequency of false results. Preliminary empirical findings show that baseline models, including SVM and Random Forest, have high levels of accuracy, precision, and recall, as well as an F1 score, which is also nearly optimal for the combined model. This study will endeavor to address the existing gap in the currently existing SDN security solutions by presenting an IDS architecture that possesses high levels of accuracy, scalability, and adaptability so that it might be able to adequately address several different types of network intrusions in real-time.

This work focuses on improving anomaly-based IDS for SDNs using ensemble learning models. The control-data plane architecture of SDNs that breaks up the old models of control and data planes poses more security threats than the traditional-signature based IDS are poorly equipped to identify for their incapability to identify emerging threats. It also analyzes three algorithms: Support Vector Machine (SVM), Random Forest (RF), and k-Nearest Neighbors (KNN) in their vanilla and ensemble forms. The results show that the ensemble approach significantly improves detection accuracy, precision, recall, and F1 scores across three datasets: general network intrusion, IoT, and SDN-specific datasets. Thus, the proposed ensemble model demonstrated 0.99 accuracy in general and IoT datasets, along with 0.97 accuracy for the individual SDN datasets, outperforming individual models. However, since the computational steps and training time were higher with the ensemble model, the approach worked well in lowering false positives and negatives. In its final analysis, the study confirms ensemble learning's viability as a strong approach to improving the security of dynamic SDN environments, and the results imply that ensemble learning can provide superior model adaptability and scalability compared to single models. Future work will further investigate these results under realistic SDN conditions and to the consideration of the potential of the deep learning methods. The contribution of this paper is listed below:

To evaluate the effectiveness of ensemble learning techniques for anomaly-based intrusion detection in Software Defined Networks (SDNs).

To compare the performance of SVM, RF, and KNN individual classifiers with their ensemble variants.

To evaluate the model generalization across IoT, general network, and SDN-specific intrusion datasets.

The rest of the paper is organized as follows: Section 2 covers the related work, discussing traditional anomaly detection methods and machine learning-based approaches. In Section 3 methodology is discussed, where we apply machine learning models such as SVM, K-NN, Random Forest, and ensemble methods to detect anomalies. Section 4 showcases the results, evaluating the models based on accuracy, precision, recall, and F1 scores, with a focus on how the ensemble model performed across three different datasets. Finally, Section 5 wraps up the paper by summarizing the main findings and suggesting future research paths, particularly in integrating deep learning to enhance anomaly detection for SDNs.

Related Work:

Intrusion Detection Systems (IDS), on the other hand, are very important tools in the current network security architecture since they can be seen as guards who keenly observe the traffic within a network for virtually any anomalous activities or security threats. IDS can be described as a software application or a piece of hardware that runs in a computer system or a network analyzing events, that occur in the said system or network for any sign of an incident or a violation of an incident which is generally classified as a violation or a probable violation of computer security policies, user acceptable policies or standard security practices [13]. The primary objective of an IDS is to identify unauthorized access and misuse, where any violation will be quickly detected to limit the rate of possible problems. IDSs over the years have changed from merely being rule-based to sophisticated systems whereby machine learning and artificial intelligence improve their detection features.

Traditional IDSs are generally categorized into two types: Two categories of Intrusion Detection Systems. They are network-based Intrusion Detection Systems (NIDS) and Host-based Intrusion Detection Systems (HIDS). NIDS is responsible for monitoring the network traffic of several devices; it records the streams of data packets in the network and looks for signs of intrusion. They work based on inspecting these packets as traffic, as well as their payloads, and investigating whether they match the threats or behaviors that are considered to be risky [14]. On the other hand, HIDS is normally implemented in single hosts or devices where its main role is to oversee and report on the health of the internal operating system and the applications running in the host. They concern themselves with the alterations that occur on vital system files, unauthorized access, and applications' logs to capture threats. Both have their peculiarities and drawbacks; for example, NIDS can be more general in monitoring general network activities while HIDS can be more detail-oriented in terms of hosts' security state.

Among the significant developments in the field of IDS, it is possible to note the possibility of the creation of anomaly-based detection systems. As distinct from the signature-based IDS, which presupposes the use of specific rules and known attack patterns, the anomaly-based IDS constructs a pattern of normal behavior, and any deviations from the latter signal a suspicious activity [15][16]. This approach would enable one to scan for new and hitherto unknown intrusive activities that are becoming more and more rampant in today's complex threat environment [17]. This method is especially useful in situations where new forms of threats appear quite frequently; therefore, anomaly-based detection is an essential component of contemporary IDS solutions. However, this method is not without problems; there might be lots of false positives and false negatives, as the identification of the anomalies that may be threats is not easy. To try to solve these difficulties, six advanced machine learning algorithms and advanced statistical methods have been introduced into IDS to enhance and expand their capacities.

Machine learning techniques have been implemented in IDSs to improve their effectiveness and decrease the role of human involvement. Support vector machines and neural networks are some of the supervised learning techniques used to build IDSs so that they can detect and differentiate different types of intrusions based on labeled datasets. Other techniques like clustering and anomalous data detection do not need training on the database to find novelties in attacks, which is an added advantage of IDS. These artificial intelligence and machine learning methodologies facilitate IDSs to learn and adapt new knowledge from the updated data stream and thus become more efficient. Furthermore, deep learning in IDS has enhanced the creation of models that are capable of analyzing other patterns in high dimensions; this has led to better detection methods as compared to the traditional methods.

Software Defined Networks (SDNs) can be described as a novel paradigm shift in the acquisition and management of network infrastructure since they change the conventional ways in which networks are developed. Data control: SDNs have a two-layer architecture, the control layer and the forwarding layer, where the control layer is centralized, and the forwarding layer is distributed. This separation is crucial in offering the elasticity and the scripts that characterize SDN. The decisions of where given traffic needs to be forwarded are made on the control plane, independent of the data plane that plays the role of forwarding the traffic to the intended destination. This decoupling makes network management easy, and the possibility of managing the network resources optimally is as easy as well.

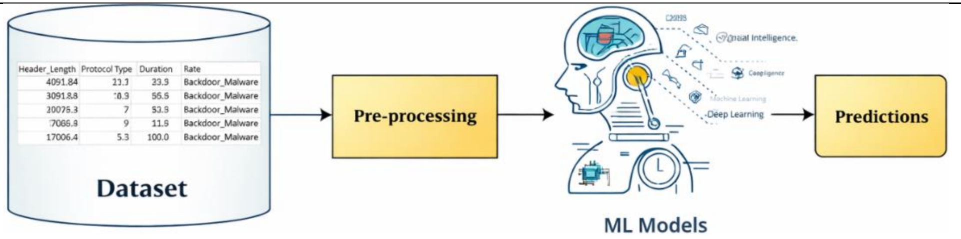
The concept of Software Defined Networks (SDNs) has several benefits as compared to conventional networking methodologies, including flexibility, programmability, and control architecture. However, with this, they bring out new security threats that need to be addressed to make sure the SDN systems are secure and reliable. Another main concern that has been identified in SDNs is the fact that the control plane is centralized. That scenario helps to control the network and avoid any problems being concealed, but at the same time, it creates a weak link. The control plane is centralized in the SDN controller; hence, if the attacker gets control of this plane, he or she will control all the network. For this reason, the SDN controller has become a very attractive target for these cyber threats, and hence, proper security measures should be employed to protect the controller.

Anomaly-based IDS is a kind of IDS that is very useful in today's world for detecting new and complex attacks. However, these systems have many problems and constraints that can affect their function and application. The ability to appreciate these challenges is fundamental to the emergence of IDS solutions that are stronger and more reliable.

The major drawback of Anomaly-based IDS is that it includes a rather large number of false positives. Since these systems regard any variation from the set pattern as a danger, they result in a high number of alerts. This is especially detrimental to local areas when the baseline of normalcy is highly fluctuating, as can commonly be seen with complex networking scenarios such as SDNs. False positives can result in the dilution of alertness: due to the large number of false alarms detected, the security personnel may ignore real threats. In addition, they also spend additional time and resources on nothing because certain activities are deemed threatening and are blocked or reported.

Methodology:

Three datasets will be used to train our models, as shown in Figure 1; each dataset will undergo basic preprocessing before the models are trained individually. Once the best results are obtained, those models are used for future ensemble learning models. This hybrid ensemble learning model will be based on the models that are doing the best in terms of subsequent correctness of this ensembled hybrid model on three distinct datasets. Finally, to confirm the accuracy and outcomes, we validate the ensemble model by comparing it with results from two recent studies (2024).



Header_Length	Protocol Type	Duration	Rate
4091.84	21.1	23.9	Backdoor_Malware
3091.88	18.3	56.6	Backdoor_Malware
20076.3	7	53.9	Backdoor_Malware
7066.8	9	11.6	Backdoor_Malware
17006.4	5.3	100.0	Backdoor_Malware

Figure 1. Intrusion Detection Using ML

Following the analysis of the data preprocessing part of the study, the research embraces several machine learning methodologies pertinent to anomaly-based Intrusion Detection Systems (IDS). These models include methods of statistical analysis, decision trees, classification, and clustering, together with supervised and unsupervised machine learning, and and hybrid approaches combining multiple models. accuracy, precision, recall, and F1 score metrics are used to evaluate model performance to determine how each of the models performs in correctly identifying the anomalies while minimizing false positives, as shown in Figures 2 and 3.

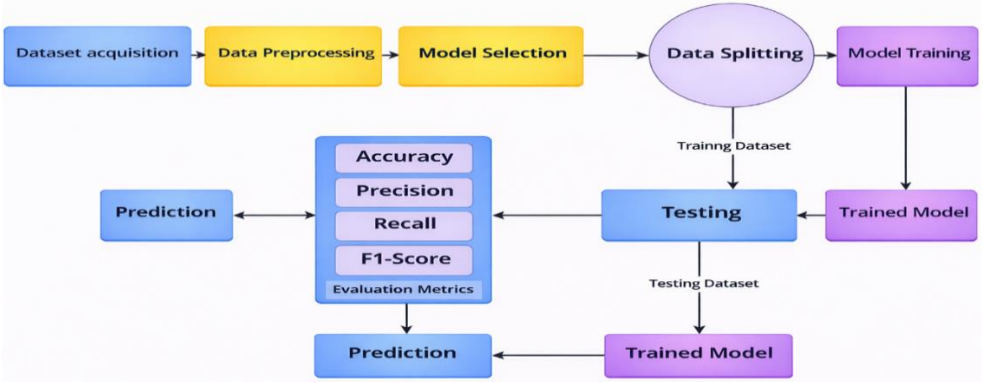


Figure 2. Flow of Work

As for the subsequent steps of the research activities of the study, the most suitable ML model is selected based on its performance on the dataset. This entails comparing the various models to the dataset and identifying the model that offers the highest efficiency and effectiveness in the identification of an anomaly. Finally, emphasis is placed on refining the selected model., which includes activities such as hyperparameter tuning, feature engineering, and improvement of the ensemble method for estimating the precision and accuracy of an effective anomaly detection system for application-level cybersecurity settings. The ensemble framework follows a stacking strategy, where SVM, RF, and KNN were used as base learners. Their output probabilities were fed into a meta-classifier to generate the final prediction. The meta-learner was trained using cross-validated predictions from the base models to prevent overfitting.

For this study, three datasets were used. We use the Network Intrusion Detection dataset on the Kaggle platform; there are numerous network traffic records, including flow duration, protocols, flag count, and classes of attacks such as DoS and DDoS. The IoT Intrusion Detection dataset is based on traffic flow and can be used to detect threats targeting IoT devices; the metrics included are those based on flow duration and types of protocols. Finally, the SDN-specific dataset includes information on network topology, number of switches, and connection details, the number of switches, and connection specifics with a view to simulating other applications within academies and the corporate world. The selected datasets vary in terms of traffic composition, feature distribution, diversity of protocols, and attack taxonomy. The general dataset represents the performance of the legacy network, the IoT dataset shows the reflection of heterogeneous and resource-constrained device traffic., and the SDN dataset depicts flow-based forwarding and controller-oriented communication patterns, specific to SDN architectures. This diversity ensures comprehensive performance validation.

The basics of the IDS model in ASD include clustering technologies as well as statistical methods in the determination of an abnormally functioning network. Clustering, like K-means, identifying outliers without supervision while also incorporating statistical modeling., which involves the use of distribution techniques to detect the anomaly, since they define normal behavior as anything that falls outside the distribution as a threat. Improvements such as the ensemble methods involve the use of various models to arrive at a better decision while reducing false alarms. Bagging, boosting, and stacking are methods that operate on many models to get a boost in productivity. However, cross-validation and real-world testing procedures are also important to oversee the performance of the model against the newly emerged cybersecurity threats.

Results Evaluation Metrics:

Evaluation metrics are used in measuring IDS performance with a special focus on Anomaly-based IDS. Some of the important metrics used are Precision, Recall: accuracy, and recognition power, respectively, as well as F1, which defines the balance between precision and accuracy of recognition. Because of the inherent class imbalance in the network intrusion data, weighted averaging was used to compute precision, recall, and F1-score. This means that the contribution made by individual classes will correspond to their support. Additionally, stratified sampling was used during dataset splitting to preserve class distribution.

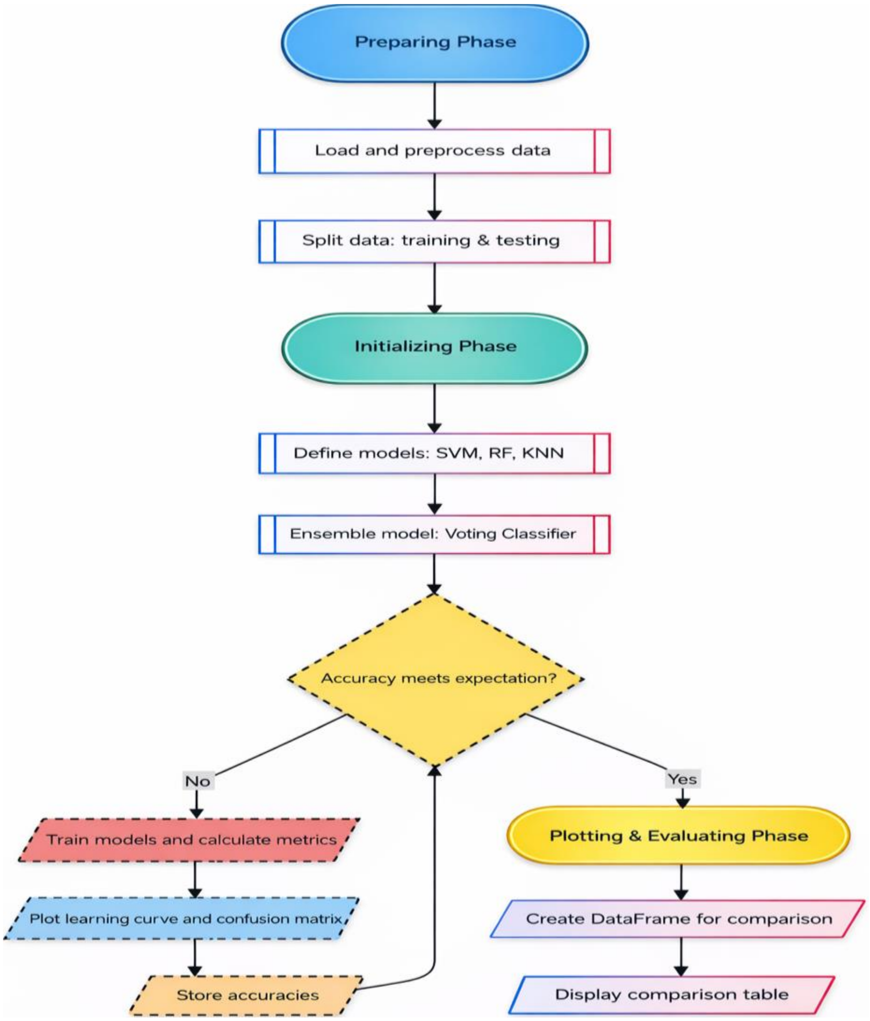

Preparing Phase
Load and preprocess data
Split data: training & testing
Initializing Phase
Define models: SVM, RF, KNN
Ensemble model: Voting Classifier
Accuracy meets expectation?
No
Yes
Train models and calculate metrics
Plot learning curve and confusion matrix
Store accuracies
Plotting & Evaluating Phase
Create DataFrame for comparison
Display comparison table

Figure 3. Flow Chart of Pseudo Code

Four ML algorithms: SVM, K-NN, RF, and Ensemble model are compared and contrasted based on three datasets. The proposed Ensemble model achieves higher accuracy and better generalization compared to the individual models. In Dataset 1, In Dataset 1, Random Forest achieves the highest accuracy of 0.999735, compared with 0.993384 for SVM and 0.994972 for the Ensemble model. In Dataset 2, the Ensemble model achieves 0.99 accuracy, which helps overcome the limitations of individual models. The results of the experiment show that the integration of models gives the best outcomes in terms of accuracy, precision, recall, and F1 scores. The Confusion matrices of SVN and KNN are presented in Figures 4 and 5, respectively.

Table 1. Comparison of Different Data Set Accuracy

Model	Accuracy Dataset 1	Accuracy Dataset 2	Accuracy Dataset 3
SVM	0.993384	0.96	0.997
K-Nearest Neighbors	0.981477	0.94	0.95
Random Forest	0.999735	0.95	0.97
Ensemble (SVM + RF + KNN)	0.994972	0.99	0.97

A comparative study of the four models: Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Random Forest, and an ensemble of these models has been presented on three datasets based on accuracy, precision, recall, and F1-score as shown in Table 1. The perfect scores are seen for all four metrics: accuracy, precision, recall, and F1 score at 1.00 for Dataset 1 and the Random Forest algorithm in particular, thereby proving its proficiency in learning and generalizing from the data. SVM and the Ensemble model also have high performance, with an accuracy of 0.99 and perfect precision at 1.00. Both models achieve recall of 0.99 and F1 score of 1.00., with no significant variations between the two models. KNN has a slightly lower result, namely 0.98 accuracy and 0.98 recall, but it is still a stable classifier in this dataset and compares favorably with SVM and Random Forest.

In the case of Dataset 2, it is possible to observe more variance depending on the selected model. The Ensemble model tops the chart once more, sustaining an accuracy of 0.99; it has a very high precision of 1 and nearly excellent recall and F1 score of 0.99, which reflects its ability to leverage the advantages of several models. SVM also boasts high results: accuracy of 0.96 and F1 score of 0.95. Still, it resembled the results obtained in Dataset 1 only in the high level of accuracy. Random Forest gives an accuracy of 0.95 in the new data set and has a high precision score of 0.97 and a high recall score of 0.96 for the new data set; thus, making the Random Forest model more reliable. Overall, we have a set of results showing that using KNN not only benefits the analysis but also, compared to other algorithms, with an accuracy of 0.94, indicating slightly lower generalization performance across datasets.

For Dataset 3 (SDN), accuracy is 0.997 with near-perfect precision of 0.999., a recall of 0.97, and an F1 score of 0.98. This goes on to prove that SVM is applicable for this dataset since it performs better than the other algorithms in this case, from the aspect of accuracy. However, the Ensemble model has a good balance with an accuracy of 0.97, a precision of 0.98, and an F1 score of 0.98, meaning that this model can analyze almost any data characteristic. Random forest is also slightly slower than the previous two models, yet achieving high accuracy of 0.97., and high precision, recall, and F1 score, each at 0.96, which demonstrates its ability to work stably with numerous datasets. KNN performs slightly less than it does, with an accuracy of 0.95, a precision of 0.95, and an F1 score equal to 0.93. Whereas it is also efficient, it is less flexible and performs slightly worse with larger datasets, such as SDN.

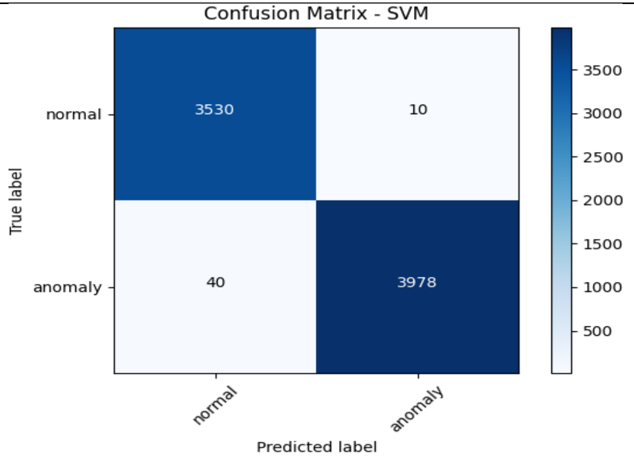


Figure 4. Confusion Matrix of SVM

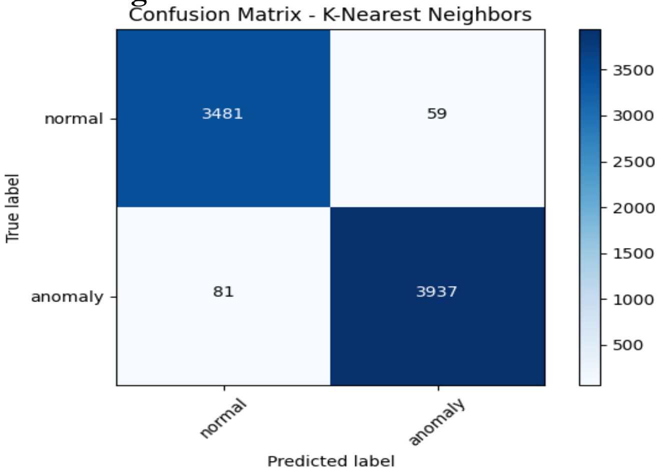


Figure 5. Confusion Metrix KNN

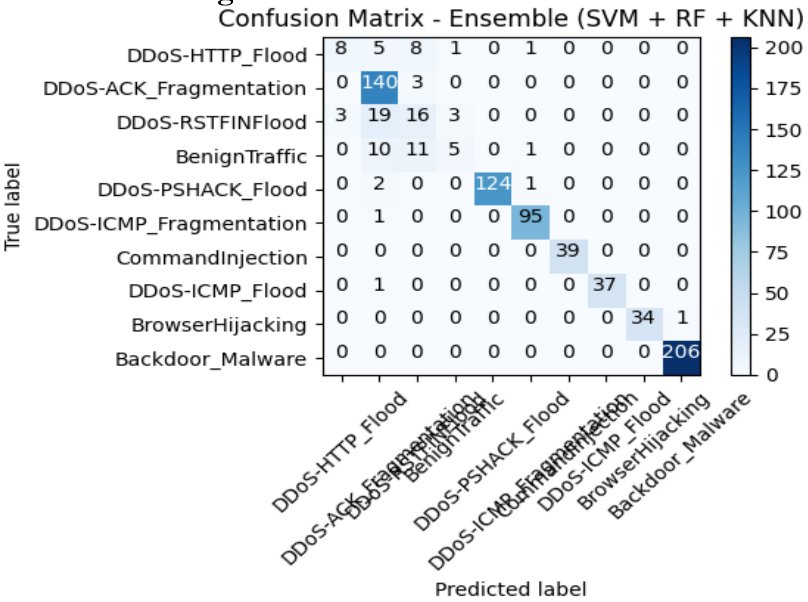


Figure 6. Confusion Matrix - Ensemble

Yet all four of the models, namely SVM, KNN, Random Forest, and Ensemble, show good results in all three sets, yet the Ensemble model always holds the best combination of accuracy, precision, recall, and F1 score, especially in Dataset 2 and Dataset 3, as shown in Figure 8. the ensemble approach mitigates the individual weaknesses of SVM, KNN, and Random Forest., resulting in a model that generalizes well across diverse datasets., making it the most generalized. This is why the Ensemble model is the best to use for work with varying data since it guarantees high accuracy and stable performance in terms of various metrics, as shown in Table 2. The experimental comparison shows that individual classifiers are sensitive to specific traffic patterns and data characteristics., which leads to a decrease in performance when tested on SDN traffic. The ensemble model minimizes this sensitivity by combining different decision boundaries, hence reducing variance and enhancing minority class detection.

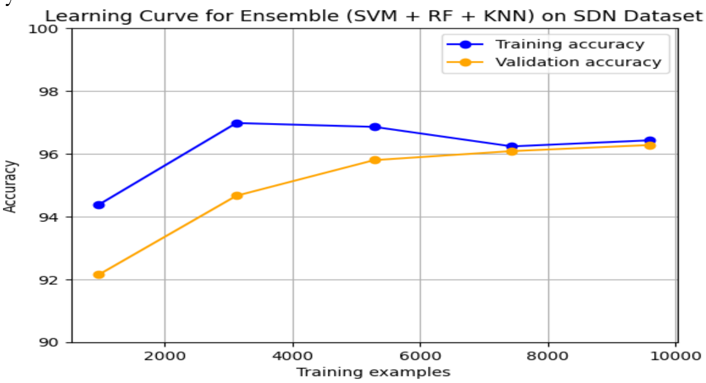


Figure 7. Learning Curve Ensembled Model

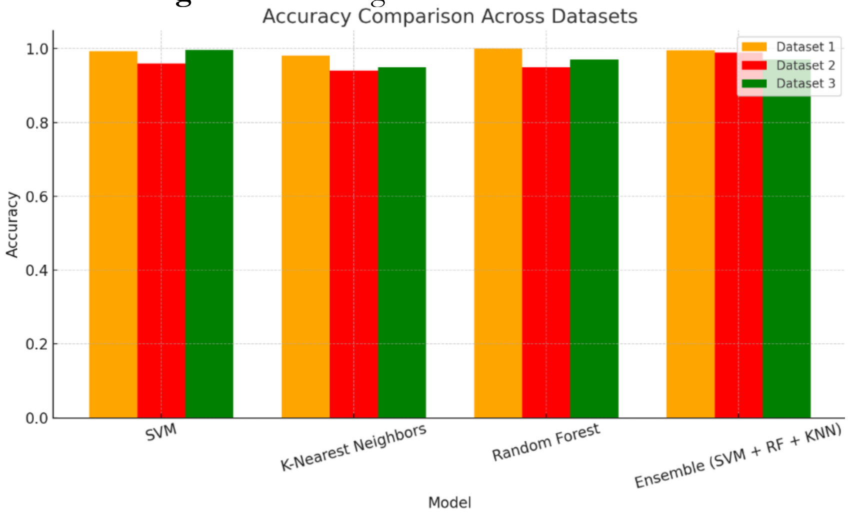


Figure 8. Accuracy Comparison of Models

Even though the proposed ensemble learning model has a slightly lower accuracy compared to some of the individual models, they are optimal because the model is a composition of the best of the three, namely SVM, Random Forest, and KNN. This widens generality and stability as every model's disadvantage is corrected by the other, thus mitigating bias or variance when faced with complex data. The study establishes that, in terms of anomaly detection in Software-Defined Networks, ensemble learning is effective. It achieves high real-time intrusion detection accuracy of 0.99, demonstrating improved performance., which makes this model applicable for SDNs datasets and scalable.

Table 2. Performance Analysis

Model	**Accuracy DS1**	**Precision DS1**	**Recall DS1**	**F1 Score DS1**
SVM	0.99	1	0.99	0.99
K-Nearest Neighbors	0.98	0.99	0.98	0.98
Random Forest	0.99	1		1
Ensemble (SVM + RF + KNN)	0.99	1	0.99	1

Model	Accuracy DS2	Precision DS2	Recall DS2	F1 Score DS2
SVM	0.96	0.95	0.94	0.95
K-Nearest Neighbors	0.94	0.93	0.94	0.95
Random Forest	0.95	0.97	0.96	0.94
Ensemble (SVM + RF + KNN)	0.99	1	0.99	0.99
Model	**Accuracy DS3**	**Precision DS3**	**Recall DS3**	
SVM	0.997	0.999	0.97	
K-Nearest Neighbors	0.95	0.95	0.96	
Random Forest	0.97	0.97	0.96	
Ensemble (SVM + RF + KNN)	0.97	0.98	0.97	

Comparison with the Latest Models:

In comparison to recent publications in the literature, our model demonstrates superior performance. For example, the most recent paper, NAD-IFA, from 2024, has an accuracy of 98%. In contrast, our model shows an accuracy of 99%. Another recent study, CSBM for Detecting Anomalies in Software-Defined Networks," again demonstrates an accuracy of about 98%. Again, our model outperforms the compared study. The accuracy comparison of the model is shown in Figure 7.

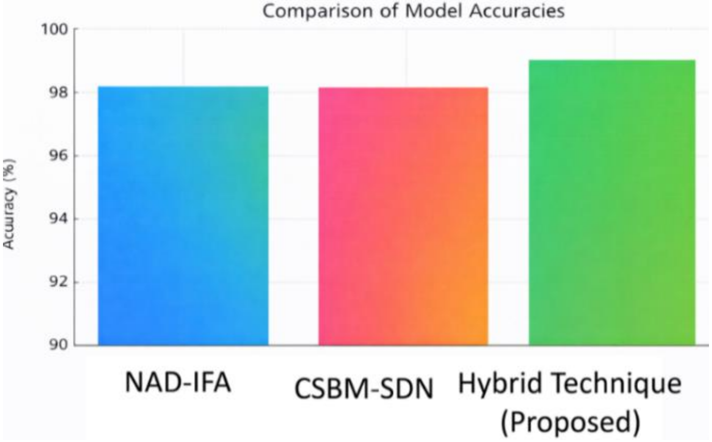


Figure 7. Comparison of Model Accuracies

Conclusion:

SDN offers flexibility and centralized control, compared to traditional networking, which requires security approaches beyond traditional signature-based intrusion detection. This paper focuses on the problem of enhancing anomaly detection in SDNs by employing the ensemble learning methods, namely, SVM, RF, and KNN. We validate three datasets: general network intrusion, IoT, and an SDN dataset used in the study to demonstrate that ensemble learning outperforms individual models in terms of accuracy, precision, recall, and F1 score, precision, recall, and F1 score. The ensemble model achieved 99% overall accuracy, with the SDN-specific dataset showing 0.997 accuracy and 0.999 precision. This work effectively demonstrates the importance of integrating multiple algorithms to improve threat detection without compromising on new threat situations in dynamic SDN settings. Furthermore, we intend to implement deep learning and an ensemble with improved features for mainstream applications. We have also planned to use large scale network in the future.

Acknowledgement: We extend our heartfelt gratitude to Prof. Dr. Muhammad Aslam, who played an important role in improving the research article. Also, the manuscript has not been published or submitted to other journals previously.

Author's Contribution: M.U. Younus and M. Fasial proposed the Anomaly-Based Intrusion Detection, performed the experimental work, and wrote the manuscript. K. Safdar provided the guidance during experiments, verified the results, and proofread the manuscript. M. Shoaib. helps in manuscript formatting and refines the manuscript.

Conflict of interest: The authors declared no conflict of interest.

References:

[1] SDN-Enabled IoT Anomaly Detection Using Ensemble Learning, "SDN-Enabled IoT Anomaly Detection Using Ensemble Learning," *Artif. Intell. Appl. Innov.*, 2020, [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC7256579/

[2] M. M. Mathabathe and T. E. Mathonsi, "A Hybrid Security Algorithm for Enhanced Man-in-the-Middle Attacks Detection in Public Wireless Networks," pp. 349–362, 2026, doi: 10.1007/978-981-95-2820-2_27.

[3] S. Bharath, D. Dhanashree, M. Nandika, and A. S. Nandhini, "Intrusion Detection in SDN Using Ensemble Learning Technique," *Int. Conf. Smart Syst. Electr. Electron. Commun. Comput. Eng. ICSSEEC 2024 - Proc.*, pp. 191–196, 2024, doi: 10.1109/ICSSEECC61126.2024.10649533.

[4] Ngamba Thockchom, Moirangthem Marjit Singh, "A novel ensemble learning-based model for network intrusion detection," *Complex Intell. Syst.*, vol. 9, 2023, [Online]. Available: https://link.springer.com/article/10.1007/s40747-023-01013-7

[5] M. Samadzadeh, M. H. Zahedi, and E. Farahani, "Using Ensemble Learning, A Cosine Similarity-Based Model for Detecting Security Anomalies in Software-Defined Networks," *2024 20th CSI Int. Symp. Artif. Intell. Signal Process. AISP 2024*, 2024, doi: 10.1109/AISP61396.2024.10475278.

[6] Abdinasir Hirsi, Lukman Audah, "Enhancing SDN security using ensemble-based machine learning approach for DDoS attack detection," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 38, no. 2, pp. 1073–1085, 2025, [Online]. Available: https://www.researchgate.net/publication/389635759_Enhancing_SDN_security_using_ensemble-based_machine_learning_approach_for_DDoS_attack_detection

[7] Salma A. Walli, "Deep Learning-Based Intrusion Detection Systems for IoT Networks: A Systematic Literature Review and Comparative Analysis," *Int. J. Comput. Informatics (Zagazig Univ.*, vol. 10, 2016, [Online]. Available: https://www.ijci.zu.edu.eg/index.php/ijci/article/view/162

[8] A. Mishra, "Intrusion Detection Through Deep Learning: Emerging Trends and Challenges," *Deep Learn. Intrusion Detect.*, pp. 107–123, Jan. 2026, doi: 10.1002/9781394285198.ch5.

[9] W. Krzemień, K. Jędrasiak, and A. Nawrat, "Anomaly Detection in Software Defined Networks Using Ensemble Learning," *Lect. Notes Networks Syst.*, vol. 439 LNNS, pp. 629–643, 2022, doi: 10.1007/978-3-030-98015-3_44.

[10] Ruyue Xin, Hongyun Liu, Peng Chen, "Robust and accurate performance anomaly detection and prediction for cloud applications: a novel ensemble learning-based framework," *J. Cloud Comput.*, vol. 12, no. 7, 2023.

[11] K. A. Fatmah Alanazi, Kamal Jambi, Fathy Eassa, Maher Khemakhem, Abdullah Basuhail, "Ensemble Deep Learning Models for Mitigating DDoS Attack in Software-Defined Network," *Intell. Autom. Soft Comput.*, 2021, [Online]. Available: https://www.techscience.com/iasc/v33n2/46781/html

[12] M. U. Younus, Y. Li, M. Shahbaz, R. Shafi, and H. He, "Robust security system for intruder detection and its weight estimation in controlled environment using Wi-Fi,"

2016 2nd IEEE Int. Conf. Comput. Commun. ICCC 2016 - Proc., pp. 985–990, May 2017, doi: 10.1109/CompComm.2016.7924852.

[13] Ibrahim M. Elezmazy, Walid Abdullah, "Advanced Intrusion Detection in Software-Defined Networks through Ensemble Modeling," *Inf. Sci. with Appl.*, vol. 4, pp. 1–11, 2024, [Online]. Available:https://www.researchgate.net/publication/384570722_Advanced_Intrusion_Detection_in_Software-Defined_Networks_through_Ensemble_Modeling

[14] J. Miller, R. Thomas, "SDN Security: Anomaly Detection Using Ensemble Learning Techniques," *IEEE Access*, vol. 12, pp. 30345–30356, 2024.

[15] Chaofei Tang, Nurbol Luktarhan, "SAAE-DNN: Deep Learning Method on Intrusion Detection," *Symmetry (Basel).*, vol. 12, no. 10, p. 1695, 2020, [Online]. Available: https://www.mdpi.com/2073-8994/12/10/1695

[16] M. U. Younus, M. K. Khan, and A. R. Bhatti, "Improving the Software-Defined Wireless Sensor Networks Routing Performance Using Reinforcement Learning," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3495–3508, Mar. 2022, doi: 10.1109/JIOT.2021.3102130.

[17] Swechchha Gupta, Buddha Singh, "Lightweight ensemble learning based intrusion detection framework with explainable artificial intelligence," *Eng. Appl. Artif. Intell.*, vol. 163, no. 2, p. 112936, 2026, [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0952197625029677?dgcid=rss_sd_all