ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 10, Issue 3, March 2025

Machine Learning-Based Intelligent Intrusion Detection System Design

[1] Prakash M, [2] Kavietha K, [3] Dinesh Kumaar R
[1] [2] [3] Assistant Professor, Department Computer Science and Engineering Er. Perumal Manimekalai College of Engineering, Hosur, India.
prakeshkgiri86@gmail.com, kaviethak@gmail.com, dineshkumaarbe@gmail.com

Abstract: ***Any concerted effort to breach a network's security measures in order to compromise its availability, privacy, or both is considered an incursion. Any attempt to violate the rules, whether they be those for acceptable use or for fundamental security, is considered an intrusion. We say that there has been a breach when an attacker uses a trustworthy person or the interconnected web to get access to a system, elevate their privileges, and then do something malicious with them. The term "intrusion detection" refers to the steps taken to identify potential breaches in privacy, security, or resource availability. The article details the process of creating a machine learning-based intelligent intrusion detection system. Feature optimization is done using ACO, whereas classification is done using Support Vector Machine, K-NN, and the Naïve Bayes algorithm.***

Keywords- Intrusion Detection, Feature Optimization, ACO, Machine Learning, SVM, KNN, Accuracy, Specificity, Sensitivity

I. INTRODUCTION

Although the security standards make an attempt to find a balance between restricting and allowing access to systems and data, they are unable to provide a guarantee that the data and systems will be completely secure. As a secondary kind of protection, the use of intrusion detection systems in systems is a standard and accepted practise. An intrusion is any intentional effort to penetrate a network's defences and, as a result, compromise its availability, confidentiality, or all three. An incursion is the term used to describe what happens when a person or group that is not authorised to access a protected network or system does so [1].

The process of determining whether there has been an attempt to violate security, privacy, or the availability of resources is known as intrusion detection [2]. The end aim is to identify organisations that are actively seeking to get around security measures, and this will be accomplished via the use of this tool. This category of intrusions encompasses anything from threats to breach acceptable usage requirements to threats to violate fundamental security measures. These breaches take place when an attacker gains access to a system by way of a trusted human or the internet of systems with the intention of elevating their privileges and then abusing them in some way. Some individuals are under the false impression that firewalls can identify and prevent the actions of malevolent actors [3].

The efficacy of the Internet may minimise the influence that is known as intrusions, despite the fact that the Internet is essential for continuous contact in the modern day. An activity that has an adverse effect on the system for which it was designed is referred to as an incursion [4]. An invasion might potentially compromise not just the availability but also the security and privacy of the system that is being attacked. When malevolent users gain access to a computer system, the system's defences are immediately compromised. There are invasions that target hosts as well as networks. Examples of host intrusions include unauthorised access to, alteration or deletion of, or destruction of data altogether, as well as attempts to render the system unreliable or unworkable. System call manipulation, file system tampering, escalation of privileges, unauthorised access to sensitive data, and malware, such as trojan horses, viruses, and worms, all work together to undermine the security of a system. System security may be compromised in a number of different ways [5].

A network intrusion occurs when hostile packets gain access to a network and try to undertake acts such as Denial of Service (DOS) attacks or break into systems. Other examples of these types of actions include hacking into computers. An effort to prohibit authorised users from accessing their computers is an example of a denial-of-service attack, abbreviated as DoS. Land, Ping of Death, and flood assaults are three examples of these types of attacks (POD) [6]. Sluggish network performance, user commands, sudden system crashes, modifications to kernel data structures, and access to websites or opened files that is unusually slow are all signs that an intrusion has occurred. Computer systems have been safeguarded

utilising intrusion prevention measures such as user authentication (passwords or biometrics), code error avoidance, and data protection as a first line of defence against potential security breaches (encryption) [7].

This article presents development of intelligent intrusion detection system using machine learning. ACO is used for feature optimization and Support Vector machine, K-NN , and Naïve Bayes algorithm is used for classification.

II. RELATED WORK

Zhao et al. [8] have created a one-of-a-kind method for intrusion detection that is based on augmented K-means. This was done in order to accommodate the distinct features of cloud computing as well as its stringent security requirements. This technique may be split down into two primary components, namely, a clustering algorithm and a distributed intrusion detection strategy. Both of these components are equally important. This system is able to recognise standard as well as atypical threats that occur in cloud computing settings. The findings of the simulation indicate that using this strategy might hasten the process of intrusion detection while simultaneously reducing the rates of both false positives and false negatives.
The technique of Chakir et al. [9], which prioritises alerts based on risk assessments, was applied. Indicators of priority, reliability, and asset worth are used as decision factors in this technique to quantify alert risk. This improves snort's ability to detect intrusions by limiting the administrator's attention to only those alerts that represent a genuine security risk; and, in turn, reduces the time and effort spent analysing false positives. The objective is to determine the significance of individual IDS alerts in terms of the overall security of an information system. The model is evaluated via a pattern-matching approach, with the KDD Cup 99 Dataset serving as a testbed for the technique.

Santoso et al [10] conducted an investigation of the efficiency of NIDS in the context of an OpenStack private cloud system. The primary objective of this investigation is to determine how well NIDS performs its responsibilities and how correctly it classifies threats. The results provide evidence of the dependability and accuracy of the model's predictions. In conclusion, the real-time alert that NIDS provides has the ability to effectively detect network intrusions based on their categorization.
An Improved Back Propagation (BP) Algorithm-based cloud-based intrusion detection model was used by Sun et al. [12] in their study. This model combines the capabilities of both the gradient descent technique, which is utilised by the BP algorithm, and the PSO algorithm, which is used to do global optimization, so that it can perform both local search and global optimization. In this work, we provide a PSO technique for optimising the initial weight and threshold of BP inside the momentum and adaptive learning rate approach. This method is designed in such a way that the BP demonstrates rapid network convergence and successfully avoids falling into a local optimum. On the basis of the results of the experiments, it is possible to state that the model that was proposed has a better average detection rate, and it may be used to the provision of cloud-based intrusion detection.

Seth et al. [13] have developed a method for the network intrusion detection system that makes use of key feature selection based on binary Grey Wolf Optimization (GWO) and a neural network classifier. The installation of an intrusion detection system, often known as an IDS, does not need to take place at a particular node in the network. It is possible to use GWO to filter out the properties of the dataset that are less significant in order to cut down on the amount of time needed to train the classifier and the amount of storage space required. The simulation experiments conducted on the NSL-KDD dataset reveal that the accuracy of the intrusion detection approach may be improved by making use of a more condensed feature collection.
Principal component analysis (PCA) was used by Ahmad et al. [6] to select feature subsets based on the variances (also known as eigen values) of the features. Despite the fact that the features with the highest eigen values might not be the ones that are the most sensitive for the classifier, this was done. Because choosing the most distinctive subset of converted features is really an optimization problem, the tried-and-true technique of picking features with the greatest eigen values must be replaced with an optimization approach. An evolutionary optimization approach known as GA has been utilised by researchers to identify the characteristics that may be altered to become the most distinguishing over time. Particle swarm optimization (PSO), which is based on research into the behaviour of animals and birds, is an additional approach of optimization that, in certain circumstances, performs more well than GA.

Methodology is presented in figure 1. ACO optimization is used for feature selection. For classification, SVM, Naïve bayes and KNN algorithm are used.

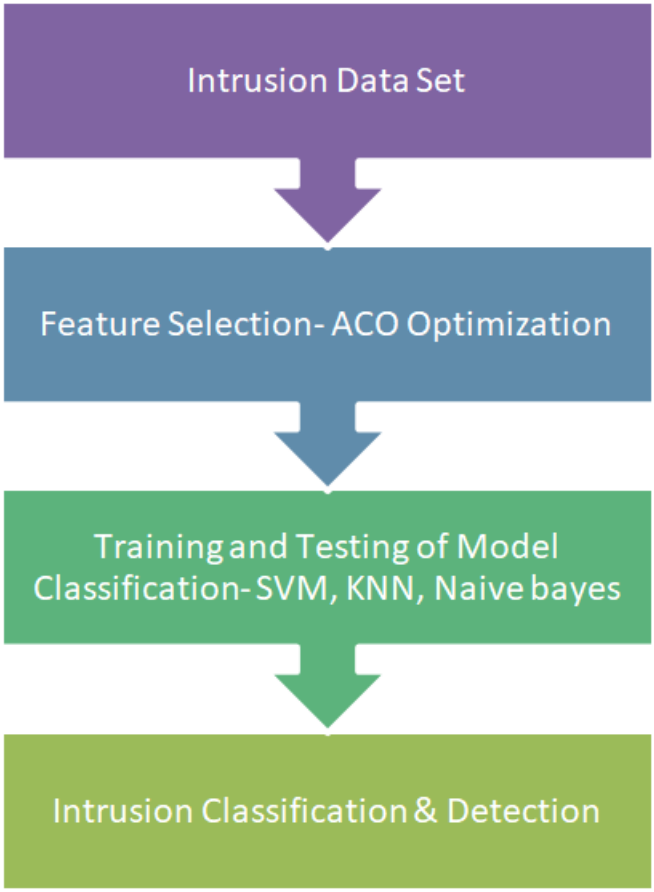

Figure 1: Intelligent Intrusion Detection System Using Machine Learning

Ant colonies rely on pheromone, an odorous chemical, as a type of nonverbal communication rather than verbal exchange. Once they have discovered a source of food, ants will next emit a pheromone in order to indicate its track. The effectiveness of a pheromone that is used to indicate ovulation is dependent on many factors, including the quantity, closeness, and quality of the food supply. Since the fragrances of ants and their trails are so similar, a solitary ant will quickly discover the path of another ant that it has been following even if it is wandering about aimlessly. This particular ant leaves behind pheromones and, in doing so, creates a pheromone trail along the paths it has already established. As a direct consequence of this, the path taken by big ants is more appealing. As a result, the chance that an ant will pick its route increases in proportion to the number of ants that have already chosen their path in the past. Therefore, the decision to do this action has been made via the positive feedback loop. ACO is a metaheuristic that was developed in order to solve difficult combinatorial progress issues. When applied to the problem of the travelling salesman, ACO demonstrated its potential as a strategy for finding the best possible answers. The employment of this technology has been beneficial to the optimization of a wide variety of problems, including data mining, telecommunications networks, vehicle routing, and a great many others. In order to create solutions to the optimization issue in an iterative manner, many synthetic ants are used. After each iteration, an ant will leave a specific amount of pheromone behind, the quantity of which will vary depending on the quality of the solutions. Each ant, as part of the iteration process, analyses a variety of alternative extensions to its current partial solution and chooses one based on a number of criteria including local heuristics and previous knowledge. This process is repeated until the problem has been fully solved [14].

Support Vector Machines (SVMs) are supervised learning models with associated learning algorithms that examine data and identify patterns [15]. These models are discarded for use in regression and classification. SVM is a non-probabilistic binary linear classifier since it takes a set of input data and identifies, for the given input, two possible classes to generate output. This means that SVM is not a probabilistic classification method. Let there be a collection of training examples, each of

which is labelled with one of the two categories; the SVM training process will then create a copy that allocates newly encountered instances to one of the two categories. SVMs search for a hyper plane that accurately classifies the data in the training set according on the features that are provided. In order to maximise the usefulness of the nearest training example, margin distance optimization of the hyper plane is performed. The collection of data points that reaches all the way to the edge on both sides is referred to as support vectors. If there was some kind of mapping or objective function between the data and the labels, the computer could be able to learn how to label the data. The dot product is the kernel function, and it is used to remap the feature vectors that are input and to locate the hyper plane. It will be possible to sort the unlabeled instances of the test set after the hyper plane has been constructed. This sorting takes place in the support vectors that were found. The resultant segmentation of the space into discrete groups is fairly precise because the SVM model treats sample specifications as if they were spatial coordinates. Because there is a gap on the side of this space, it is possible to map new samples to the same space and determine whether or not they belong to certain categories. With the right tools, support vector machines (SVMs) may easily do non-linear classification. This is made possible by the kernel technique, which completely remaps inputs into high-dimensional feature spaces to enable linear classification.

It is possible to use the non-parametric method known as the k-NN algorithm to problems involving classification and regression. The voting strategy is used to categorise the item; nevertheless, the classification, much like that of any other supervised algorithm, is reliant on the properties of the object in question as well as labelled training examples. In order to simplify the process of developing an algorithm, k-NN takes into account the distance that exists between the newly created instance and its immediate neighbours. The new item is added to the domain of the k-NN algorithm in a manner that is determined by the votes of each neighbour as well as the category that obtains the most votes overall. It is essential to choose the correct number for k, since if k equals 1, the instance will be immediately accepted by the class that is the closest match. If k is too low, a new instance will be simply assigned to the class that is the closest match, and if k is too high, it will boost the computational cost and complexity, which is counter to the core idea of the k-NN method. The k-NN algorithm often employs Shephard's method as a means of determining the proper weights to apply. Because of k-shown NN's efficacy in text categorization, its use in the field of intrusion detection has become widespread. Since k-NN does not require rebuilding the profile or checking each sequence in the middle of the new program's execution, it significantly reduces the computational burden associated with the prediction of new instances while categorising the new invader. This is because rebuilding the profile and checking each sequence are both required by other methods. Each intrusion is compared to a word, and each process is compared to a document, in a manner that is analogous to how they would be categorised in a text-based organisation system [16].

To put it more simply, the principle upon which the Naive Bayes classifier is built is that the effect of the feature value on the class supplied is independent of the values of other features in decision making [17]. The classification is strengthened by the use of a probabilistic technique, in which the likelihood of each attribute-value combination is taken into consideration. It is possible to teach it in SL scheme, which might be of tremendous use in a variety of intricate real-world crises, particularly ones that are susceptible to computer-based diagnostic procedures. As a consequence of the conditional independence of attributes, we are able to compute the variances of attributes for each class rather than the whole covariance matrix. This is because the conditional independence of attributes. Classifiers based on the Naive Bayes algorithm have a denominator that does not depend on the class variable (C') and a probability (P) that takes into account both the class variable (C') and the feature variables (F).

III. RESULT AND DISCUSSION

NSL- KDD data set consists of 41 attributes. ACO is used to select features from data set. ACO selected 18 attributes from 41 attributes. Then refined data set is classified by Support Vector Machine, K-NN algorithm and Naïve Bayes classification algorithm. Results are shown in figure 2, figure 3 and figure 4.
Accuracy of SVM is 98 percent, which is higher than NB and KNN algorithm. Sensitivity of KNN is 99 percent, which is higher than the sensitivity of SVM 98 percent. Specificity of SVM is 98 percent, which is higher than NB and KNN algorithm

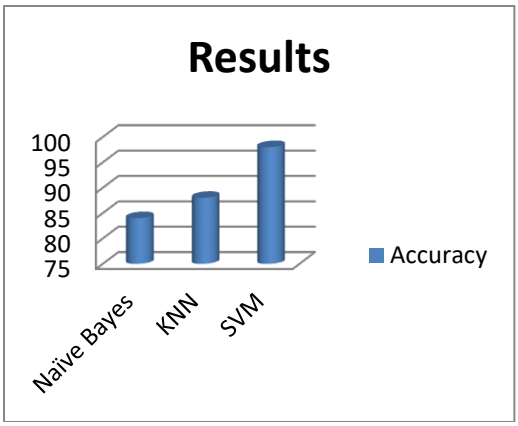

Fig.2 Accuracy comparison for IDS

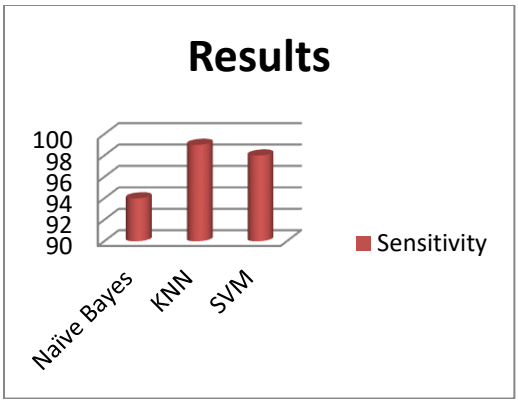

Fig.3 Sensitivity comparison for IDS

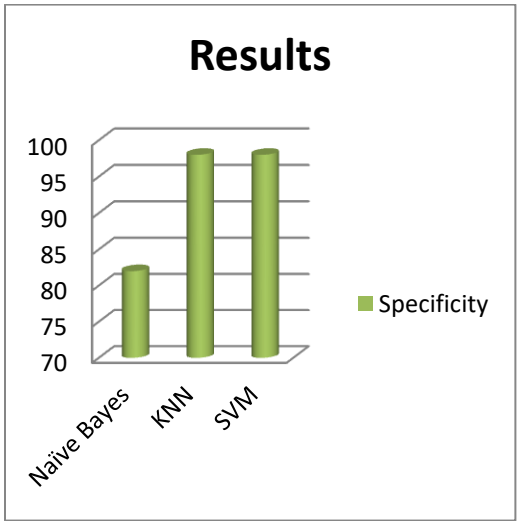

Fig.4 Specificity comparison for IDS

Conclusion

Any intentional effort to get past the protections of a network in order to make it less accessible, less private, or all three is considered an intrusion. The process of determining whether or not someone has attempted to breach a system's security, privacy, or availability of resources is referred to as intrusion detection. The construction of an intelligent intrusion detection system using machine learning is presented in this paper. ACO is used in the process of feature optimization, whereas Support Vector machine, K-NN, and the Naive Bayes algorithm are utilised in the classification process. Accuracy of SVM is 98 percent, which is higher than NB and KNN algorithm. Sensitivity of KNN is 99 percent, which is higher than the sensitivity of SVM 98 percent. Specificity of SVM is 98 percent, which is higher than NB and KNN algorithm

REFERENCES

[1] Raghuvanshi, A., Singh, U., Sajja, G., Pallathadka, H., Asenso, E., & Kamal, M. et al. (2022). Intrusion Detection Using Machine Learning for Risk Mitigation in IoT-Enabled Smart Irrigation in Smart Farming. Journal Of Food Quality, 2022, 1-8. doi: 10.1155/2022/3955514
[2] Raghavendra, S., Dhabliya, D., Mondal, D., Omarov, B., Sankaran, K. S., Dhablia, A., ... & Shabaz, M. (2022). Development of intrusion detection system using machine learning for the analytics of Internet of Things enabled enterprises. IET Communications.
[3] UmaMaheswaran, S. K., Prasad, G., Omarov, B., Abdul-Zahra, D. S., Vashistha, P., Pant, B., & Kaliyaperumal, K. (2022). Major Challenges and Future Approaches in the Employment of Blockchain and Machine Learning Techniques in the Health and Medicine. Security and Communication Networks, 2022.
[4] Raghuvanshi, A., Singh, U., & Joshi, C. (2022). A Review of Various Security and Privacy Innovations for IoT Applications in Healthcare. Advanced Healthcare Systems, 43-58.doi: 10.1002/9781119769293.ch4
[5] Jaiswal, S, Saxena, K, Mishra, A &Sahu, SK 2016, 'A KNN-ACO Approach for Intrusion Detection using KDDCUP'99 dataset', IEEE International Conference on Computing for Sustainable Global Development (INDIACom), pp. 628-633.
[6] Ahmad, I & Amin, F 2014, 'Towards feature subset selection in intrusion detection', IEEE Joint International Information Technology and Artificial Intelligence Conference, pp. 68-73.
[7] A. Raghuvanshi, U. Singh, T. Kassanuk and K. Phasinam, "Internet of Things: Security Vulnerabilities and Countermeasures", ECS Transactions, vol. 107, no. 1, pp. 15043-15052, 2022. Available: 10.1149/10701.15043ecst.
[8] Zhao, X & Zhang, W, 2016, 'An Anomaly Intrusion Detection Method Based on Improved K-Means of Cloud Computing", IEEE International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC), pp. 284-288.
[9] Chakir, EM, Moughit, M & Khamlichi, YI 2017, 'An Efficient Method for Evaluating Alerts of Intrusion Detection Systems', IEEE International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), pp. 1-6.
[10] Santoso, BI, Idrus, MRS & Gunawan, IP, 2016, 'Designing Network Intrusion and Detection System using Signature-Based Method for Protecting OpenStack Private Cloud', IEEE International Annual Engineering Seminar (InAES), pp. 61-66.
[11] Bombatkar, AG & Parvat, TJ 2015, 'Efficient Method for Intrusion Detection and Classification and Compression of Data', IEEE International Conference on Computational Intelligence and Communication Networks (CICN), pp. 1085-1090.
[12] Sun, H, 2016, 'Improved BP Algorithm Intrusion Detection Model Based on KVM', IEEE International Conference on Software Engineering and Service Science (ICSESS), pp. 442-445.
[13] Seth, JK & Chandra, S, 2016, 'Intrusion Detection Based on key Feature Selection using Binary GWO', IEEE International Conference on Computing for Sustainable Global Development (INDIACom), pp. 3735-3740.