*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
***Article Received:*** *22 August 2023* ***Revised:*** *10 October 2023* ***Accepted:*** *24 October 2023*

# Network Based Intrusion Detection System Using Weighted Product Model (WPM)

**K. Swapna Rani**, Assistant professor Department of CSE KG Reddy College of Engineering and Technology chilukuru village Hyderabad. Mail: r.swapna.k@gmail.com
**Gayatri Parasa**, Assistant Professor Department of CSE KoneruLakshmaiah Education Foundation Vaddeswaram, Andhra Pradesh, India – 522502 Mail: gayathriparasa20@gmail.com
**Dr. D. Hemanand**, Professor, Department of Computer Science and Engineering, S.A. Engineering College Poonamallee-AvadiRoad, Thiruverkadu, Chennai-600077 Tamil Nadu, India
Mail: hemanand1982@gmail.com
**Dr. Devika SV**, Professor, Department of ECE Hyderabad Institute of Technology and Management, Hyderabad
Mail: devikasv.ece@hitam.org
**Dr. S. Balambigai**, Associate professor Dept of ECE Kongu Engineering College
Mail sbalambigai@gmail.com

## Abstract
A security technology called a network-based intrusion detection system (NIDS) was created to safeguard computer networks against unauthorised access and criminal activity. This technology works by analysing network traffic, spotting potential risks, and informing administrators of any possible incursions or attacks. NIDS research ensures that intrusion detection systems are built to minimise the gathering and storage of sensitive data by taking into account the value of privacy and data protection .In general, network-based intrusion detection system research has a major impact on how well these security measures operate, how efficiently they perform, and how adaptable they are.By addressing the evolving challenges posed by cyber threats, NIDS research helps organizations enhance their network security posture, protect sensitive information, and defend against potential intrusions and attacks." The weighted product model (WPM), a multi-criteria decision-making (MCDM) technique, is used to evaluate and rank solutions based on a variety of distinct criteria. It provides a methodical approach to decision-making by considering the relative importance of each attribute and the performance of other solutions in relation to those criteria. The WPM normalises the data, weights the criteria, and gives a weighted score for each alternative. The option with the greatest score is regarded as the ideal option. The weighted product model offers a structured framework for making decisions by taking into account many factors and their varying degrees of importance. It enables decision-makers to assess and contrast options using a wide range of criteria, resulting in more informed and unbiased choices. It's crucial to check nonetheless that the model's weights and normalisation techniques appropriately capture the decision-maker's preferences as well as the features of the choice problem.J48, Random Forest, JRIP, RIDOR, PART. The definition of true positive, false positive, true negative and false negative rates has already been established. These metrics for measuring the effectiveness of classification algorithms, anomaly detection systems, and binary decision-making processes are accurately presented. As can be seen from the results, J48 received the highest rank, while PART received the lowest .In order to increase the security of computer networks, network-based intrusion detection systems (NIDS) are essential. They provide real-time monitoring and analysis of network traffic to identify suspected breaches and malicious activities, enabling appropriate action to be taken. However, it is important to recognize that NIDS can have limitations and are not infallible.

**Keywords:** network intrusions, technology, analysis, network based, NIDS

## 1. INTRODUCTION

The Internet of Things (IoT), a recent advancement in information technology and communications, has surpassed traditional methods of environmental sensing. It has enabled the development of technologies that significantly improve the quality of life. By 2020, it is estimated that there will be 50 billion IoT devices, making it one of the fastest-growing fields in computing. It is anticipated that by 2025, the IoT and its associated applications could have an annual economic impact ranging from $3.9 trillion to $11.1 trillion. Devices could become smart objects through the integration of the key IoT technologies, such as communication

technologies, pervasive and ubiquitous computing, embedded devices, Internet protocols, sensor networks, and AI-based applications..Intrusions into networks are a prevalent and complex issue today. An attacker has benefits when assaulting a system remotely that do not exist while targeting a host. Network attacks, for example, may be completely unnoticed from the audit trail that is left by the targeted host and normally don't need any prior connection to the attacked system. Open, effective, and risky internal networks have grown as a result of the deployment of firewalls to shield business networks from the unsecured Internet. Intrusions by insiders could happen on these networks.Attackers have an edge when using networks, however "intrusion detection systems (IDSs)" can also benefit from some of these features. Without regard to the installed OSes or the monitoring tools accessible on the hosts as well as networks, for example, can provide comprehensive information on system activity. The operation of the observed hosts or the entire network as a whole won't be harmed by network auditing, and network audit stream generation can't be disabled. Finally, timing data from network traffic is more accurate and accurate than the audit trails created by conventional OS auditing instruments. Distributed IDSs and network-based IDSs are the two primary categories of network-oriented intrusion detection systems.. The initial single-host intrusion detection technique is expanded to several hosts by distributed IDSs. By performing intrusion detection analysis on audit streams obtained from many sources, distributed IDSs enable detection of attacks that target several systems. These systems include IDES and ISOA, to name just two. Instead of concentrating on the communication infrastructure—the network and its protocols—network-based IDSs take a different approach and concentrate on the computing infrastructure—the hosts and their operating systems. These systems collect information on security using the network. EMERALD, DIDS, and NSM are just a few examples of these systems. A "network-based intrusion detection system (NIDS)" monitors and analyses network traffic to protect a system from network-based attacks. A NIDS reads all incoming data packets and scans them for any odd patterns. Depending on the severity of the threat is, the computer can take the proper action, such as informing administrators or preventing network access through the IP address that originated the attack. For NIDS and prevention systems (NIPS), Snort is an open-source technology. Snort can operate in three different modes: sniffer, packet logger, and network intrusion detection; these modes encompass live packet tracking, log compilation, and preventive system functions. Since insider and intrusion threat identification requires extensive study, IT security is a crucial concern. Numerous contributions for processing security-related dataidentifying botnets port scans brute force assaults and other tasks have been published. These projects all share the need for representative network-based data sets. Additionally, benchmark data sets provide a solid foundation for assessing and contrasting the effectiveness of various network intrusion detection systems (NIDS). Given a labelled data collection in which each data point is assigned to the categories of normal or attack, the number of detected attacks or the number of false alarms may be utilised as assessment criteria. The study of invasion and detection of insider threats has garnered a lot of interest because IT security is a significant concern. Many articles have been released on maintaining "security-related information through, detecting botnets through, port scans through, brute force assaults through, and other issues". All of these programmes share the need for relevant network-based data sets. Furthermore, benchmark data sets can be used to compare and evaluate the performance of various "network intrusion detection systems (NIDS)". The number of identified attacks or the number of erroneous alarms may be used as assessment criteria when given a tagged collection of data in which every point of information is allocated to the class of regular or attack.By examining the data records that programmes associated with the same network have accessed, intrusion detection searches for computer dangers. Host-based intrusions and network-based intrusions are the two main divisions of these attacks. System call information gathered through an audit procedure, which logs all system calls made on a machine's account by each user, is frequently used by host-cantered attack detection techniques. Every system being watched typically has these auditing procedures running. Network traffic information using a network packet sniffer, like tcpdump, is typically used in network-based detection of threats approaches. A shared medium is utilised by many networks of computers, especially the widely utilised Ethernet (IEEE 802.3) network. Therefore, all that has to be used is the network that is shared between the monitored workstations and the packet sniffer.

## 2. MATERIALS AND METHODS

The Weighted Product Model (WPM), a multi-criteria decision-making (MCDM) technique, is used to assess and rank options. It enables decision-makers to consider a variety of factors and tally them up to generate an overall score for each possibility.

**The WPM follows a straightforward procedure:**Specify the criteria for decision-making Determine the factors that will affect your choice. These standards ought to be quantifiable and should correspond to the decision-maker's goals or preferences.

**Normalise the Criteria:** To ensure that the values are similar, normalise each criterion to a shared scale. If the criteria are measured using multiple scales or units, this step is required. Techniques like min-max normalisation and z-score normalisation can be used for normalisation.

**Assign Weights:** Give each criterion a weight to represent its relative importance. The weights should all equal one, or 100%, to represent the overall importance given to each criterion.

**Establish the Decision Matrix:** Construct a decision matrix in which each row denotes a potential course of action and each column a criterion. For each option and criterion, enter their normalised values in the matrix.

Calculate the weighted product in step 5: Add the corresponding weight for each criterion to the normalised values of each choice. Each alternative and criterion receive a weighted score as a result of this calculation.

**Add the Weighted Scores:** Add the weighted scores for each possibility to obtain the overall score. The choice considered to be the greatest is the one with the highest final score.

**Sensitivity Analysis (Optional):** To assess the influence on the ranking, perform sensitivity analysis by altering the weights. This process enables the exploration of various possibilities and aids in understanding how resilient the choice is.

A flexible and simple strategy that can handle both qualitative and quantitative criteria is the weighted product model. It does, however, presuppose that the criteria are distinct and equally significant within the bounds of their respective weights. Sensitivity analysis can aid in overcoming some of these restrictions and offer perceptions into the choice-making procedure.

**Weighted Product Model:**

In many situations, the Weighted Product Model (WPM) is a crucial decision-making tool. The WPM is important for the following main reasons:

Using multiple criteria to make decisions: Decision-makers can assess alternatives simultaneously based on a variety of factors thanks to the WPM. This is especially helpful when making decisions in difficult situations with many different variables to take into account. The WPM enables decision-makers to explicitly state their preferences and priorities by giving weights to each criterion.

Flexibility and Applicability: The WPM can be applied to a variety of decision problems in a range of fields.. It is adaptable for decision-making in a variety of sectors, including business, engineering, finance, project management, and more. It can handle both quantitative and qualitative criteria. The model can handle a variety of data, including expert judgements and subjective evaluations.

Openness and Reproducibility: The WPM offers a systematic, open method of decision-making. The procedure becomes more methodical and repeatable by clearly specifying the criteria, normalising the data, and allocating weights. Due to the transparency, debates and the development of consensus among decision-makers are made easier for stakeholders to grasp.

Taking Stakeholder Preferences into Account: The WPM enables decision-makers to take into account the priorities and preferences of various stakeholders. Decision-makers can represent the relative importance that diverse stakeholders place on various aspects by allocating weights to criteria. This makes decision-making more inclusive and aids in balancing conflicting interests.Overall, the Weighted Product Model is useful because it gives multi-criteria decision-making a systematic approach, takes stakeholder preferences into account, encourages transparency, and permits sensitivity analysis. It is a helpful tool in a variety of decision-making circumstances due to its adaptability and extensive applicability.J48 is generally used to create decision tree models using labelled training data in machine learning and data mining applications.

***Here are some key features and characteristics of the J48 algorithm:***

Handling Categorical and Numerical Attributes: J48 can handle both categorical and numerical attributes. It can discretize continuous attributes to create categorical partitions during the tree construction process.Handling Missing Values: J48 has built-in mechanisms to handle missing attribute values. It can use surrogate splits to estimate missing attribute values and determine the appropriate branch to follow in the decision tree.Pruning: J48 incorporates a pruning mechanism to avoid overfitting the training data. It uses a separate validation dataset or cross-validation to assess the performance of the decision tree and selectively prune Handling Multi-Class Classification: J48 supports multi-class classification tasks by creating decision trees with multiple branches and leaf nodes. It can assign class labels to instances based on the majority class in each leaf node.Interpretability: J48 produces extremely comprehensible decision trees. The final model takes the shape of a tree structure, with each leaf node standing in for a class label assignment and each inside node denoting a choice based on an attribute.Feature Importance: J48 can provide an indication of the importance of different attributes in the classification task. By examining the attribute selection frequency during the tree construction process, one can identify the most influential attributes for the classification task. J48 is a machine learning library that can be found in many machine learning software packages, including Weka (Waikato Environment for Knowledge Analysis), a well-liked open-source machine learning and data mining tool. It is popular because of how easy it is to understand, how well it can handle categorical and numerical data, and how easily it can manage both. The decision tree extension algorithm, Random Forest, has a number of benefits.

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
***Article Received:** 22 August 2023 **Revised:** 10 October 2023 **Accepted:** 24 October 2023*

***The main traits and qualities of Random Forest are listed below:***

Ensemble Learning: Random Forest makes the final forecast by combining the predictions of various distinct decision trees. Each decision tree is independently constructed using various bootstrap samples, or the combined forecasts of all the trees and subsets of the training data are used to get the final prediction. Random Feature Selection: Random Forest randomly selects a subset of features from the available features to use in building each decision tree. As a result of this process, there is less correlation between the trees and more variation among them. It shields the model from overfitting and strengthens its capacity for generalisation. Bagging: By randomly sampling the training data with replacement and bagging, it is possible for some cases to show up more than once in the bootstrap sample. This approach introduces randomness and reduces the variance of the model.Out-of-Bag Analysis: Because each decision tree is trained using a separate bootstrap sample, certain occurrences are excluded from the training set for each tree. Without the need for a separate validation set, these out-of-bag (OOB) instances can be used for assessment. An objective evaluation of the model's performance and accuracy is provided via OOB evaluation.Generalisation and Robustness: Random Forest is renowned for its resistance to noisy data and outliers. It can handle datasets with many features that are high in dimension. The combination of multiple trees helps to capture complex relationships and reduces the risk of overfitting, leading to better generalization to unseen data.Random Forest has applications in various domains, including classification tasks like image recognition, text classification, and fraud detection, as well as regression tasks such as predicting housing prices or stock market trends. Its ability to handle complex datasets, robustness against noise, and feature importance analysis make it a popular and effective algorithm in machine learning.A rule-based classification system called JRip (Repeated Incremental Pruning to Produce Error Reduction) combines features of rule induction and decision tree techniques. It was developed as part of the Weka machine learning software and is particularly useful for datasets with discrete attributes.

***the key characteristics and features of the RIDOR algorithm:***

Rule-Based Classification: RIDOR, like JRip, generates a set of rules to classify instances in a dataset. Each rule consists of a conjunction of attribute-value conditions that determine the class label assignment.Incremental Rule Construction: Similar to JRip, RIDOR builds the rule set incrementally by adding rules one at a time. It starts with an empty rule set and iteratively adds rules to cover instances not covered by existing rules.Rule Pruning for Overfitting Reduction: RIDOR includes additional pruning steps specifically aimed at reducing overfitting. It evaluates the potential reduction in errors and the complexity of the rule set to decide whether to prune a rule. This pruning process helps simplify the model and improve its generalization ability.Rule Ordering: RIDOR considers the order in which rules are added to the rule set. It uses a heuristic based on the expected reduction in errors and the complexity of the rule to determine the best order of rule addition. This ordering helps in prioritizing rules that contribute the most to the overall accuracy while keeping the model compact.When working with datasets that are very susceptible to overfitting or when the interpretability and simplicity of the final model are essential, RIDOR is especially helpful. By incorporating additional pruning steps, RIDOR aims to strike a balance between accuracy and model complexity, leading to improved generalization and better performance on unseen data.PART (Partial Decision Trees) is a rule-based classification algorithm that constructs decision trees using a division-and-conquer tactic. It is an extension of the C4.5 algorithm and is designed to handle datasets with missing attribute values efficiently.

***Here are the key characteristics and features of the PART algorithm:***

Rule-Based Classification: PART generates a set of rules to classify instances, similar to other rule-based algorithms. Each rule in the rule set represents a specific class label assignment based on attribute-value conditions.Divide-and-Conquer Strategy: PART builds decision trees using a divide-and-conquer strategy. Continuously creating decision trees on these smaller portions of the data, it divides the data into fewer subsets depending on attribute requirements.Partial Trees: PART constructs partial decision trees that do not cover the entire attribute space. Unlike traditional decision trees that aim for complete coverage, PART focuses on creating more compact and interpretable trees that still provide accurate predictions. Handling Missing Attribute Values: PART has built-in mechanisms to handle missing attribute values effectively. It can handle both discrete and continuous attributes with missing values by creating separate branches or rules to account for these missing values. This capability allows PART to handle datasets with missing data without requiring explicit imputation or preprocessing.Rule Pruning: PART includes a pruning step to simplify the generated rule set. Pruning is performed based on statistical significance tests to determine if removing a rule from the set leads to a significant decrease in accuracy. This pruning helps to reduce model complexity and improve generalization to unseen data.PART is particularly suitable for datasets with missing attribute values and when interpretability is important. Its focus on constructing partial decision trees and handling missing data allows it to handle real-world datasets more effectively. By creating compact and interpretable models, PART provides a balance between model complexity and accuracy. The True Positive Rate (TPR), also known as sensitivity, recall, or hit rate, is a performance indicator for assessing how well a binary classification model performs. It measures the percentage of positive instances that the model accurately detected out of all the actual positive

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
***Article Received:*** *22 August 2023* ***Revised:*** *10 October 2023* ***Accepted:*** *24 October 2023*

instances in the dataset. A binary classification model's performance is measured using a metric known as the False Positive Rate (FPR). It determines the proportion of negative occurrences that the model incorrectly interprets as positive out of all the actual negative examples in the dataset.a measure of performance known as True Negative Rate (TNR) .It is used to evaluate how effectively a binary classification model performs and is frequently referred to as specificity or selectivity. It determines the proportion of negative cases that the model correctly recognises as negative out of all the actual negative instances in the dataset. A binary classification model's efficacy is measured using a performance indicator known as the false negative rate (FNR), also known as the miss rate or Type II error. It calculates the proportion of positive instances that the model misclassifies as negative out of all the real positive cases in the dataset.

## 3. RESULT AND DISCUSSION

**TABLE 1.**Network Based Intrusion Detection System

| | TPR | FPR | TNR | FNR | Accuracy |
|---|---|---|---|---|---|
| J48 | 0.993 | 0.064 | 0.936 | 0.007 | 0.984 |
| RandomForest | 0.996 | 0.018 | 0.982 | 0.004 | 0.994 |
| JRIP | 0.993 | 0.05 | 0.95 | 0.007 | 0.986 |
| RIDOR | 0.993 | 0.064 | 0.936 | 0.007 | 0.97 |
| PART | 0.991 | 0.03 | 0.97 | 0.009 | 0.987 |

The performance characteristics of various Network Based Intrusion Detection System (NIDS) algorithms are displayed in Table 1. The metrics are broken down as follows:

1. TPR (True Positive Rate): Also referred to as sensitivity or recall, this metric represents the percentage of real positive events that the model accurately detected.
2. FPR (False Positive Rate): This statistic measures the percentage of negative occurrences that the model misclassifies as positive.
3. TNR (True Negative Rate), also known as specificity, estimates the percentage of real negative occurrences that the model accurately identified.
4. FNR (False Negative Rate): It determines the percentage of positive cases that the model misclassifies as negative.
5. By assessing the percentage of examples that were properly classified out of all the instances, accuracy reflects how accurately the model's predictions were made overall.

Now, let's interpret the results for each algorithm:

1. J48:
    - TPR: 0.993
    - FPR: 0.064
    - TNR: 0.936
    - FNR: 0.007
    - Accuracy: 0.984
2. Random Forest:
    - TPR: 0.996
    - FPR: 0.018
    - TNR: 0.982
    - FNR: 0.004
    - Accuracy: 0.994
3. JRIP:
    - TPR: 0.993
    - FPR: 0.050
    - TNR: 0.950
    - FNR: 0.007
    - Accuracy: 0.986

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
***Article Received:*** *22 August 2023* ***Revised:*** *10 October 2023* ***Accepted:*** *24 October 2023*

4. RIDOR:
    - TPR: 0.993
    - FPR: 0.064
    - TNR: 0.936
    - FNR: 0.007
    - Accuracy: 0.984
5. PART:
    - TPR: 0.991
    - FPR: 0.030
    - TNR: 0.970
    - FNR: 0.009
    - Accuracy: 0.987

These metrics provide insights into the performance of each algorithm. Generally, higher TPR and TNR values indicate better performance, while lower FPR and FNR values are desirable. Additionally, higher accuracy values indicate more accurate predictions overall. Based on the given metrics, Random Forest has the highest TPR (0.996) and TNR (0.982), along with the highest accuracy (0.994), suggesting that it may be the most effective algorithm for the NIDS task among the listed options.

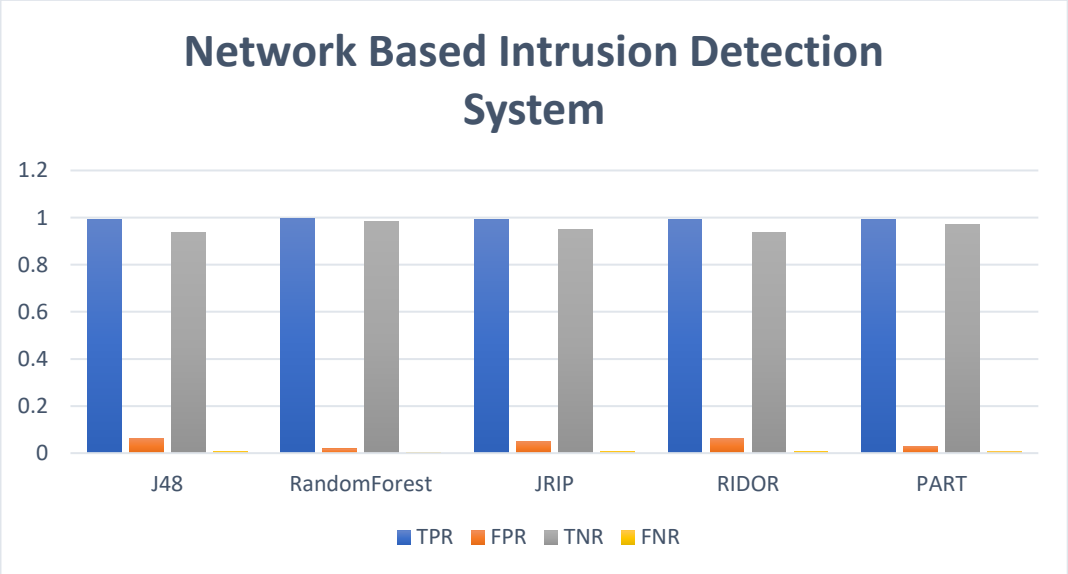


**FIGURE 1.**Network Based Intrusion Detection System

Figure 1 displays the TPR graphically, with Random Forest Private Limited displaying the highest value and PART displaying the lowest value. FPR As can be observed, PART is displaying the lowest value while RIDOR is displaying the greatest value .In TNR, Random Forest exhibits the highest value while J48 exhibits the lowest value.. FNR It can be noticed that whereas Random Forest displays the lowest value, PART displays the highest value .In terms of accuracy, Random Forest exhibits the highest value while RIDOR exhibits the lowest value.

**TABLE 2.** Performance value

| Performance value | | | | |
|---|---|---|---|---|
| 0.996988 | 1 | 1 | 0.571429 | 1 |
| 1 | 0.28125 | 0.953157 | 1 | 0.98994 |
| 0.996988 | 0.78125 | 0.985263 | 0.571429 | 0.997972 |
| 0.996988 | 1 | 1 | 0.571430 | 1 |

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
***Article Received:*** *22 August 2023* ***Revised:*** *10 October 2023* ***Accepted:*** *24 October 2023*

| | | | | |
|---|---|---|---|---|
| 0.99498 | 0.46875 | 0.964948 | 0.444444 | 0.99696 |

The performance value for the following alternative parameters is shown in Table 2: J48, Random Forest, JRIP, RIDOR, and PART. TPR (True Positive Rate), FPR (False Positive Rate), TNR (True Negative Rate), FNR (False Negative Rate), Accuracy are the parameters that should be evaluated.

T

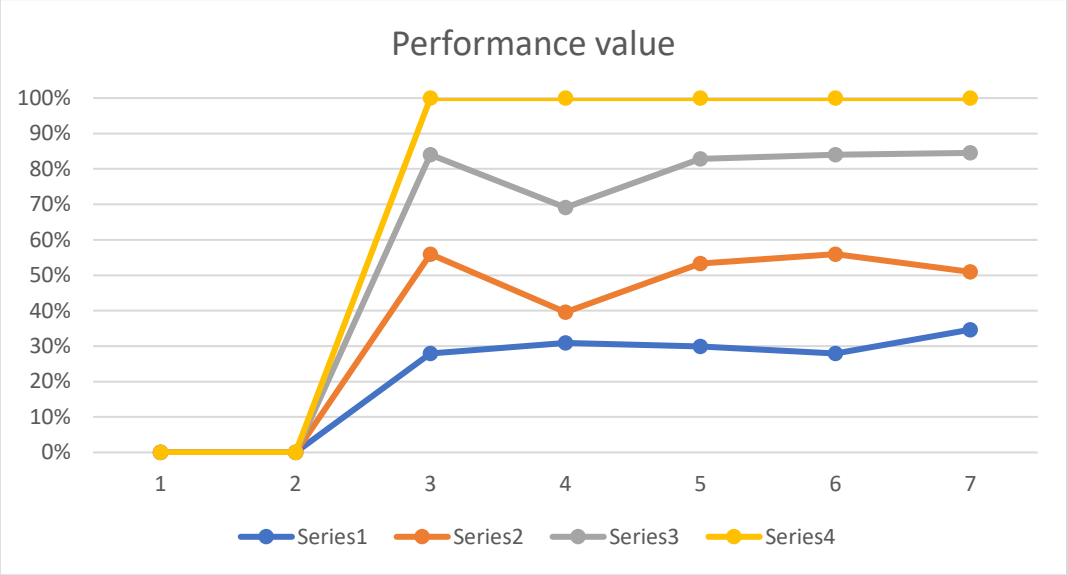


**FIGURE 2.** Performance value

The alternative parameters J48, Random Forest, JRIP, RIDOR, and PART are displayed in Figure 2's performance value. TPR (True Positive Rate), FPR (False Positive Rate), TNR (True Negative Rate), FNR (False Negative Rate), Accuracy are the parameters that should be evaluated.

**TABLE 3.**Weight

| Weight | | | | |
|---|---|---|---|---|
| 0.25 | 0.25 | 0.25 | 0.25 | 0.25 |
| 0.25 | 0.25 | 0.25 | 0.25 | 0.25 |
| 0.25 | 0.25 | 0.25 | 0.25 | 0.25 |
| 0.25 | 0.25 | 0.25 | 0.25 | 0.25 |
| 0.25 | 0.25 | 0.25 | 0.25 | 0.25 |

The Weight ages used for the analysis are displayed in Table 3. For the analysis, we use the same weights for all the parameters.

**TABLE 4.** Weighted Normalized Decision Matrix

| Weighted Normalized Decision Matrix | | | | |
|---|---|---|---|---|
| 0.999246 | 1 | 1 | 0.869442 | 1 |
| 1 | 0.728238 | 0.988078 | 1 | 0.997475 |
| 0.999246 | 0.940151 | 0.996295 | 0.869442 | 0.999493 |
| 0.999246 | 1 | 1 | 0.869441 | 1 |
| 0.998743 | 0.827438 | 0.99112 | 0.816497 | 0.999239 |

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
***Article Received:*** *22 August 2023* ***Revised:*** *10 October 2023* ***Accepted:*** *24 October 2023*

The Weighted Normalised Decision Matrix with Alternative Parameters (J48, Random Forest, JRIP, RIDOR, and PART) is displayed in Table 4. TPR (True Positive Rate), FPR (False Positive Rate), TNR (True Negative Rate), FNR (False Negative Rate), Accuracy are the parameters that should be evaluated.

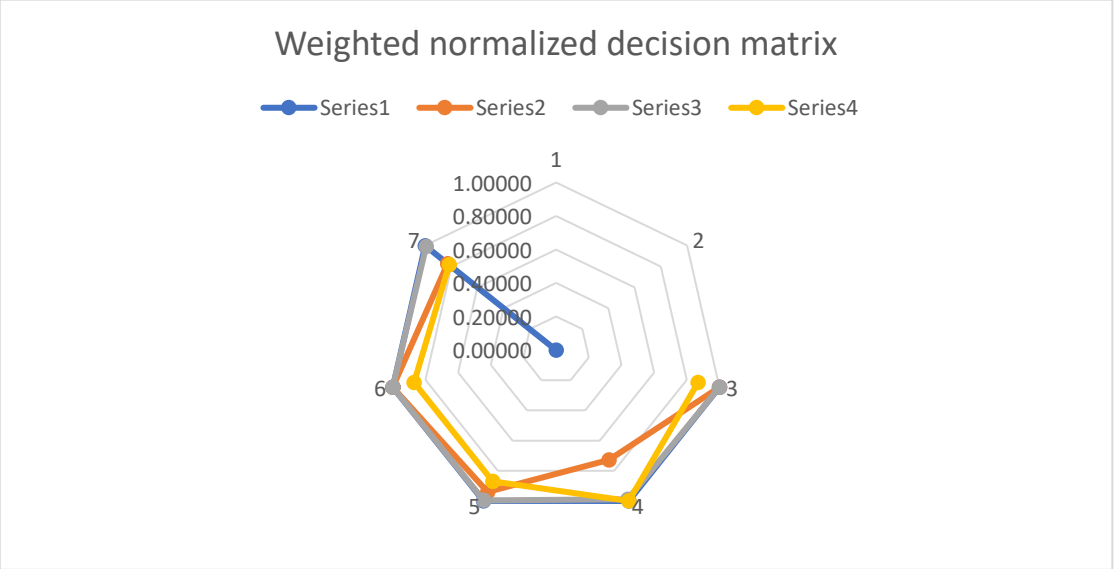


**FIGURE 3.** Weighted normalized decision matrix

**TABLE 5.** Preference Score&Rank

| | Preference Score | Rank |
|---|---|---|
| J48 | 0.868786 | 1 |
| RandomForest | 0.719555 | 4 |
| JRIP | 0.813764 | 3 |
| RIDOR | 0.868785 | 2 |
| PART | 0.668758 | 5 |

Table 5.shows the Preference Score value J480.722838, RandomForest0.719555, JRIP0.813764, RIDOR0.868786, PART0.668758.the final result of this paper the J48is in 1st rank,RandomForest is in 4thrank, JRIP is in 3rd rank,RIDOR is in 2ndrank, PART is in 5th rank.

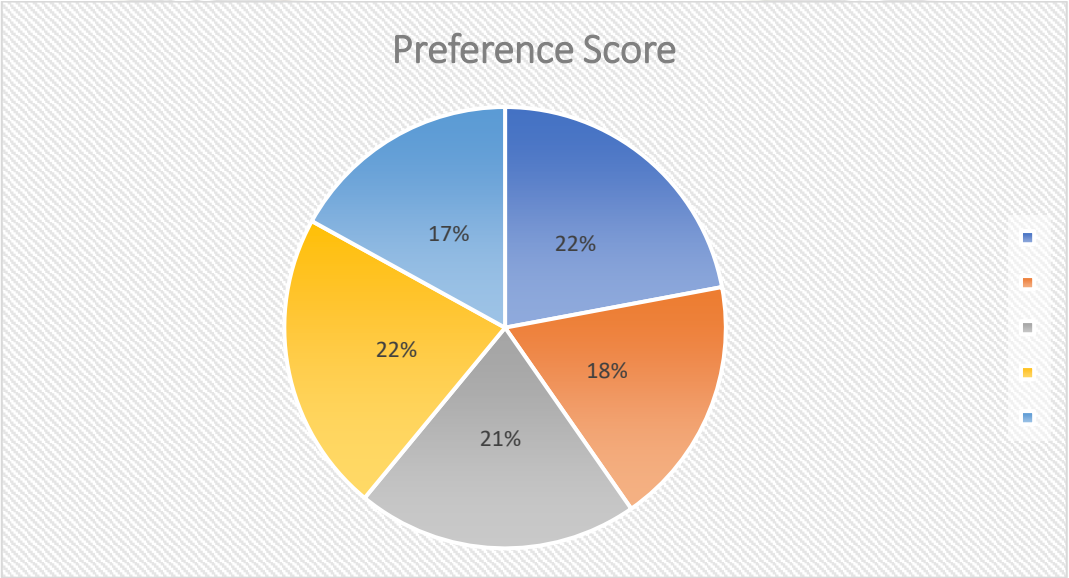


**FIGURE 4.** Preference Score

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 10*
***Article Received:*** *22 August 2023* ***Revised:*** *10 October 2023* ***Accepted:*** *24 October 2023*

Figure 4.Preference Score shows the RIDOR: With a preference score of 0.868786, RIDOR achieves the highest ranking. JRIP: JRIP secures the second rank with a preference score of 0.813764. J48: J48 algorithm obtains the third rank with a preference score of 0.722838. Random Forest: Random Forest captures the fourth rank with a preference score of 0.719555. PART: PART algorithm receives the fifth and final rank with a preference score of 0.668758. To summarize, RIDOR performs the best based on the preference scores, followed by JRIP, J48, Random Forest, and PART in descending order.

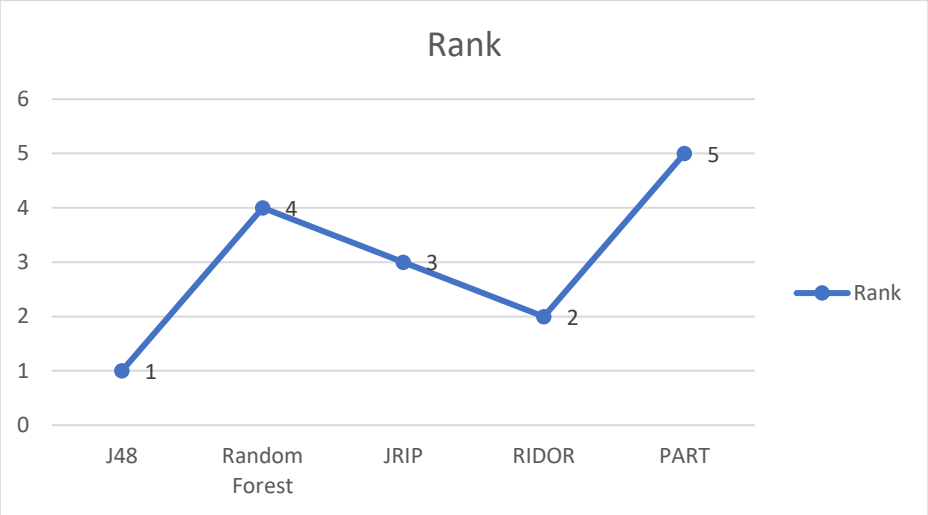


**TABLE 5.** Rank

Table 5. Rankshows the graphical representation in the final evaluation of this paper, the J48 algorithm emerges as the top performer, securing the first rank. Following closely behind is the RIDOR algorithm, securing the second rank. The JRIP algorithm captures the third rank, while Random Forest falls behind in the fourth rank. Lastly, the PART algorithm takes the fifth rank in the overall ranking of the evaluated algorithms.

## CONCLUSION

Modern cyber security infrastructure must include a network-based intrusion detection system (NIDS), which is essential for protecting networks. It accomplishes this by monitoring and analyzing network traffic, enabling the identification of potential threats and unauthorized activities.The primary objective of a NIDS is to detect and respond to various types of intrusions, such as malicious attacks, unauthorized access attempts, and suspicious network behaviour. Through the analysis of network traffic patterns, signatures, and anomalies, a NIDS can accurately identify and promptly alert system administrators or security personnel regarding potential security breaches.NIDS systems employ a range of techniques and technologies, recognising known attack patterns and unusual network behaviour, such as signature-based identification, anomaly detection, and behavioural analysis. They can be placed in a network architecture at a variety of locations, including network boundaries, routers, switches, and even individual hosts.. Implementing a NIDS brings several benefits, including improved network security, early threat detection, faster response times, and enhanced incident response capabilities. By continuously monitoring and analyzing network traffic, a NIDS helps organizations minimize the impact of security breaches and mitigate the risks associated with cyber attacks. It is crucial to remember that an NIDS should be included as part of a thorough cyber security strategy and is not a stand-alone solution. To create a multi-layered defence system, it should be reinforced by other security measures like firewalls, antivirus software, and user education .In conclusion, in the current environment of cyber danger, a network-based intrusion detection system is a crucial tool. By offering real-time monitoring, detection, and response capabilities, it significantly enhances the overall security posture of organizations, safeguarding sensitive data and critical systems against unauthorized access and malicious activities.

## REFERENCES

1. Vigna, Giovanni, and Richard A. Kemmerer. "NetSTAT: A network-based intrusion detection system." Journal of computer security 7, no. 1 (1999): 37-71.
2. Singh, Amrit Pal, and Manik Deep Singh. "Analysis of host-based and network-based intrusion detection system." International Journal of Computer Network and Information Security 6, no. 8 (2014): 41-47.

3. Ring, Markus, Sarah Wunderlich, Deniz Scheuring, Dieter Landes, and Andreas Hotho. "A survey of network-based intrusion detection data sets." Computers & Security 86 (2019): 147-167.
4. Linda, Ondrej, Todd Vollmer, and Milos Manic. "Neural network based intrusion detection system for critical infrastructures." In 2009 international joint conference on neural networks, pp. 1827-1834. IEEE, 2009.
5. Bivens, Alan, Chandrika Palagiri, Rasheda Smith, Boleslaw Szymanski, and Mark Embrechts. "Network-based intrusion detection using neural networks." Intelligent Engineering Systems through Artificial Neural Networks 12, no. 1 (2002): 579-584.
6. Zarpelão, Bruno Bogaz, Rodrigo SanchesMiani, Cláudio Toshio Kawakani, and Sean Carlisto de Alvarenga. "A survey of intrusion detection in Internet of Things." Journal of Network and Computer Applications 84 (2017): 25-37.
7. Asharf, Javed, Nour Moustafa, Hasnat Khurshid, Essam Debie, Waqas Haider, and Abdul Wahab. "A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions." Electronics 9, no. 7 (2020): 1177.
8. Shah, Bhavin, and Bhushan H. Trivedi. "Artificial neural network based intrusion detection system: A survey." International Journal of Computer Applications 39, no. 6 (2012): 13-18.
9. Karatas, Gozde, and Ozgur KoraySahingoz. "Neural network based intrusion detection systems with different training functions." In 2018 6th International Symposium on Digital Forensic and Security (ISDFS), pp. 1-6. IEEE, 2018.
10. Dias, Leonardo P., Jés de Jesus FiaisCerqueira, Karcius DR Assis, and Raul C. Almeida. "Using artificial neural network in intrusion detection systems to computer networks." In 2017 9th Computer Science and Electronic Engineering (CEEC), pp. 145-150. IEEE, 2017.
11. Das, Niva, and Tanmoy Sarkar. "Survey on host and network based intrusion detection system." International Journal of Advanced Networking and Applications 6, no. 2 (2014): 2266.
12. Lo, Wai Weng, SiamakLayeghy, MohanadSarhan, Marcus Gallagher, and Marius Portmann. "E-graphsage: A graph neural network based intrusion detection system for iot." In NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium, pp. 1-9. IEEE, 2022.
13. Chua, Tuan-Hong, and Iftekhar Salam. "Evaluation of machine learning algorithms in network-based intrusion detection system." arXiv preprint arXiv:2203.05232 (2022).
14. KS, Devikrishna, and B. Ramakrishna. "An artificial neural network based intrusion detection system and classification of attacks." International Journal of Engineering Research and Applications 3 (2013): 1959-1964.
15. Al-Janabi, Sufyan T. Faraj, and Hadeel Amjed Saeed. "A neural network based anomaly intrusion detection system." In 2011 Developments in E-systems Engineering, pp. 221-226. IEEE, 2011.
16. Tunas BangsaPematangsiantar, S. T. I. K. O. M. "Comparison of weighted sum model and multi attribute decision making weighted product methods in selecting the best elementary school in Indonesia." International Journal of Software Engineering and Its Applications 11, no. 4 (2017): 69-90.
17. Goswami, Shankha Shubhra, Dhiren Kumar Behera, and Soupayan Mitra. "A comprehensive study of weighted product model for selecting the best product in our daily life." Brazilian Journal of Operations & Production Management 17, no. 2 (2020): 1-18.
18. Aminudin, Nur, Eni Sundari, K. Shankar, P. Deepalakshmi, Rita IrvianiFauzi, and AndinoMaseleno. "Weighted product and its application to measure employee performance." International Journal of Engineering & Technology 7, no. 2.26 (2018): 102-108.
19. Susanto, R., and A. D. Andriana. "Employee recruitment analysis using computer based weighted product model." In IOP Conference Series: Materials Science and Engineering, vol. 662, no. 2, p. 022049. IOP Publishing, 2019.
20. Das, Bijoy, Suman Sankar Bhunia, Sarbani Roy, and Nandini Mukherjee. "Multi criteria routing in wireless sensor network using weighted product model and relative rating." In 2015 Applications and Innovations in Mobile Computing (AIMoC), pp. 132-136. IEEE, 2015.
21. Setyawan, Agus, Florentina YuniArini, and Isa Akhlis. "Comparative analysis of Simple Additive Weighting method and weighted product method to new employee recruitment Decision Support System (DSS) at PT. Warta Media Nusantara." Scientific Journal of Informatics 4, no. 1 (2017): 34-42.
22. Levine, Seth E., and Linda J. Broadbelt. "Detailed mechanistic modeling of high-density polyethylene pyrolysis: Low molecular weight product evolution." Polymer Degradation and Stability 94, no. 5 (2009): 810-822.
23. Fitriasari, Novi Sofia, SyifaAfifahFitriani, and Rosa ArianiSukamto. "Comparison of weighted product method and technique for order preference by similarity to ideal solution method: Complexity and accuracy." In 2017 3rd International Conference on Science in Information Technology (ICSITech), pp. 453-458. IEEE, 2017.
24. San Cristóbal Mateo, José Ramón, and José Ramón San Cristóbal Mateo. "Weighted sum method and weighted product method." Multi criteria analysis in the renewable energy industry (2012): 19-22.
25. Divayana, D. G. H., A. Adiarta, and I. B. G. S. Abadi. "Initial draft of CSE-UCLA evaluation model based on weighted product in order to optimize digital library services in computer college in Bali." In IOP Conference Series: Materials Science and Engineering, vol. 296, no. 1, p. 012003. IOP Publishing, 2018.
26. Putra, SyahrizalDwi, Rohmat Indra Borman, and Gina Hapsari Arifin. "Assessment of Teacher Performance in SMK Informatika Bina Generasi using Electronic-Based Rating Scale and Weighted Product Methods to Determine the Best Teacher Performance." International Journal of Informatics, Economics, Management and Science (IJIEMS) 1, no. 1 (2022): 55-62.