Data and Metadata. 2022; 1:33
doi: 10.56294/dm202271

REVIEW

Enhancing Intrusion Detection Systems using Ensemble Machine Learning Techniques

Mejora de los sistemas de detección de intrusos mediante técnicas de aprendizaje automático conjunto

Ibraheem Khalil Ibraheem[1]

[1]Midland Oil Company of The Iraqi Minstry of Oil. Iraq.

Cite as: Khalil Ibraheem I. Enhancing Intrusion Detection Systems using Ensemble Machine Learning Techniques. Data and Metadata. 2022; 1:33. https://doi.org/10.56294/dm202271

Submitted: 13-09-2022 **Revised:** 06-11-2022 **Accepted:** 20-12-2022 **Published:** 22-12-2022

Editor: Prof. Dr. Javier González Argote

ABSTRACT

The increasing usage of the Internet has also brought about the risk of network attacks, leading to the need for effective intrusion detection systems. This chapter aims to fill the gap in literature by conducting a comprehensive review of 55 relevant studies conducted from 2000 to 2007, focusing on the use of machine learning techniques for intrusion detection. The reviewed studies are compared based on the design of their classifiers, the datasets used in their experiments, and other experimental setups. Single, hybrid, and ensemble classifiers are examined, and their achievements and limitations are discussed. The chapter provides a thorough evaluation of the strengths and weaknesses of using machine learning for intrusion detection and suggests future research directions in this field. In conclusion, this chapter addresses the need for a comprehensive review of machine learning techniques in intrusion detection. It provides insights into classifier design, dataset selection Other experimental details an assessment of the use of machine learning for intrusion detection is presented, and recommendations for future studies are suggested.

Keywords: PSNR; LSB; Watermarking; Legendre Moment; DCT.

RESUMEN

El creciente uso de Internet también ha traído consigo el riesgo de ataques a la red, lo que ha llevado a la necesidad de sistemas eficaces de detección de intrusiones. Este capítulo pretende llenar el vacío existente en la literatura realizando una revisión exhaustiva de 55 estudios relevantes llevados a cabo entre 2000 y 2007, centrados en el uso de técnicas de aprendizaje automático para la detección de intrusiones. Los estudios revisados se comparan en función del diseño de sus clasificadores, los conjuntos de datos utilizados en sus experimentos y otras configuraciones experimentales. Se examinan clasificadores simples, híbridos y de conjunto, y se discuten sus logros y limitaciones. El capítulo ofrece una evaluación exhaustiva de los puntos fuertes y débiles del uso del aprendizaje automático para la detección de intrusiones y sugiere futuras líneas de investigación en este campo. En conclusión, este capítulo responde a la necesidad de una revisión exhaustiva de las técnicas de aprendizaje automático en la detección de intrusos. Proporciona información sobre el diseño de clasificadores, la selección de conjuntos de datos y otros detalles experimentales. Se presenta una evaluación del uso del aprendizaje automático para la detección de intrusos y se sugieren recomendaciones para futuros estudios.

Palabras clave: PSNR; LSB; Marca de Agua; Momento de Legendre; DCT.

INTRODUCTION

The Internet has become an indispensable tool in our daily lives, permeating various aspects such as trad, recreation, and knowledge. Particularly in the realm of business, the Internet is utilized extensively, with both businesses and customers relying on it for activities such as websites and email. However, the security of information transmitted over the Internet must be carefully considered. Intrusion detection has emerged as a critical research problem for both personal and business networks.[1]

Given the multitude of risks associated with cyber-attacks in the online realm, numerous mechanisms have been devised to combat these hazards. In particular, intrusion detection systems (IDSs) serve as a protective barrier against external attacks on computer systems. The primary goal of IDSs is to defend against attacks by allowing the detection of malevolent networking communications and unapproved usage of computer systems, a duty that traditional firewalls find difficult to achieve. Intrusion detection is based on the belief that the conduct of invaders varies from that of authorized users.[2]

Broadly speaking, (IDSs) can be categorized into two groups depending on their detection methods: anomaly-based and signature-based. (signature-based) detection. Anomaly-based detection involves identifying intrusions by detecting deviations from established normal usage patterns, while misuse-based detection relies on known attack patterns or vulnerabilities within the system to identify intrusions.[3]

Several anomaly detection systems have been created, utilizing diverse machine learning techniques. For instance, some researches have utilized individual learning methods like artificial neural networks, evolutionary algorithms, and kernel machines.

Others have utilized hybrid or ensemble techniques, combining different learning methods. These techniques typically function as classifiers that distinguish between normal and potentially malicious incoming Internet access. However, there has been no comprehensive review of the different machine learning techniques in the field of intrusion detection.

Therefore, the aim of this paper is to analyze a study and related systems that have been published and this analysis will delve into the methods used, experimental setups, and potential directions for future research from a machine learning perspective.

MACHINE LEARNING TECHNIQUES

Pattern classification

The ability to select the formula is the procedure of examining unprocessed information and arranging it into distinct groups or categories. Different pattern recognition issues can be solved by using supervised and unsupervised learning techniques. In supervised learning, a function is constructed based on training data, where each training data includes an input vector and its corresponding output (i.e., class label). The goal of the learning (training) task is to be able to detect the expected space between the input and output data in order to establish a classifier or model. Once the model is created, real classification can be optimized.[4,5]

Single classifiers

The problem of knowing the secret hack can be addressed with a machine learning architecture. Machine learning techniques such as k-nearest, artificial neural networks, decision trees, self-organizing maps, and others have been used to address these problems.

K-nearest neighbor

K-nearest neighbor (k-NN) is a straightforward and conventional nonparametric method for classifying samples. It computes the approximate distances between various points in the input vectors and assigns the unlabeled point to the category of its k-closest neighbors. The selection of the parameter k is crucial in creating a k-NN classifier, as different k values yield different performance outcomes. A significantly large value of k may result in longer classification time and can impact prediction accuracy. k-NN belongs to the instance-based learning approach and differs from the inductive learning approach. It does not involve a model training stage but instead searches the examples of input vectors and classifies new instances. Thus, k-NN “on-line” trains the examples and identifies the k-nearest neighbors of the new instance.[6,7]

Support vector machines

Support vector machines (SVM) were initially introduced by Vapnik in 1998. SVM starts by transforming the input vector into a feature space with a higher dimension and subsequently identifies the most effective hyperplane for separation. hyperplane within that space. The decision boundary, or the separating hyperplane, is determined based on support vectors rather than the entire training sample set, making it highly robust against outliers. Specifically designed for binary classification, an SVM classifier separates a set of training vectors belonging to two different classes. It is important to note that the support vectors are the training

https://doi.org/10.56294/dm202271

samples close to a decision boundary. Furthermore, SVM allows the user to specify a penalty factor parameter, which enables balancing between the number of misclassified samples and the width of the decision boundary.

Artificial neural networks

Artificial neural networks are designed to mimic the way the human brain processes information. One popular type of neural network architecture is called the multilayer perceptron (MLP), which is often used in pattern recognition tasks. The MLP consists of different layers, including an input layer, one or more hidden layers, and an output layer.

The input layer contains sensory nodes that receive input data, while the hidden layers contain computation nodes that process this data. The output layer also consists of computation nodes, which produce the desired output based on the input data.

Each connection between nodes in the MLP is associated with a weight scalar. During the training phase, these weights are adjusted to optimize the network's performance. The backpropagation learning algorithm is commonly used to train MLPs, as it iteratively adjusts the weights based on the error between the predicted output and the desired output., also known as backpropagation neural networks. The training process begins with random weights assigned, followed by weight tuning using the algorithm to determine the most effective representation of hidden units that minimizes the misclassification error.

Self-organizing maps

Self-organizing maps (SOMs) are a type of machine learning algorithm that use unsupervised learning to create a low-dimensional representation of high-dimensional data. They were developed by Finnish professor Teuvo Kohonen in the 1980s.

SOMs are often used for clustering and visualization purposes, as they can help in identifying patterns and relationships between data points. The algorithm works by creating a grid of neurons, each with its own weight vector. These weight vectors are initially randomly assigned to the data points in the input space.

During training, the SOM adjusts its neurons' weights based on the similarities between the input data and the weights. The most similar neurons to a particular input are identified using a distance metric, such as Euclidean distance. The weights of the winner neuron and its neighboring neurons are then updated to move closer to the input data point.

This process is repeated iteratively, with the neighborhood of the winner neurons shrinking over time, until the SOM converges to a stable configuration. The final grid of neurons represents a compressed, low-dimensional representation of the high-dimensional input space.

SOMs are known for their ability to preserve the topological properties of the input data, meaning that spatially close data points in the input space are also represented as neighbors in the SOM grid. This makes SOMs useful for visualizing complex data in a simpler and more interpretable form.

Overall, self-organizing maps provide a powerful tool for analyzing and understanding high-dimensional data by organizing it into a lower-dimensional representation while preserving its inherent structure and relationships.[8]

Decision trees

Decision trees are a popular machine learning algorithm used for classification tasks. They consist of a tree-like structure where each internal node represents a feature or attribute, and each leaf node represents a class label or decision.

The tree is built by recursively splitting the data based on the values of different features, such that the resulting splits maximize the information gain or decrease the impurity in the dataset. This process continues until a stopping criterion is met, such as reaching a maximum tree depth or when all the instances belong to the same class.

To classify a new instance, it traverses the decision tree from the root node to a leaf node based on the feature values of that instance. The leaf node reached represents the predicted class label for that instance.

Decision trees have several advantages, including their interpretability, versatility, and ability to handle both categorical and numerical features. They can also handle missing values and outliers. However, decision trees are prone to overfitting, especially when the tree becomes too complex. To overcome this limitation, techniques like pruning or using ensemble methods like random forests or gradient boosting can be applied.

decision trees are a powerful and widely used classification algorithm that constructs a tree-like model to make predictions based on feature values. They are easy to understand and interpret, making them a popular choice in various fields such as finance, healthcare, and customer segmentation.[9]

Naïve bayes networks

Naïve Bayes networks (NB) are probabilistic graph models used to represent the structural and causal

dependencies between random variables. They are particularly useful when the exact probabilistic relationships among variables are difficult to express. NB answers questions about the probability of a certain event we can analyze the network's configuration using conditional probability equations. In this configuration, a directed acyclic graph (DAG) is commonly used to depict the relationships between variables in the system. Each node in the DAG represents a variable, while the connections between nodes represent the impact or influence between these variables.[10]

Genetic algorithms

Genetic algorithms (GAs) are inspired by natural selection and evolution. They start with a large amount of data from the expected answers and use appropriate tools in order to generate excellent ratings for each solution. Through iterations, weaker solutions are omitted and stronger combinations are put in their place. The algorithm mimics adaptive survival by eliminating programs with low fitness and allowing those with high fitness to survive and reproduce.[11]

Fuzzy logic

Fuzzy set theory, deals with imprecise and uncertain information commonly encountered in the real world. It uses membership values ranging from 0 to 1 to represent the degree of membership in a set. Fuzzy logic allows for reasoning under uncertainty and is particularly useful in situations where precise boundaries are difficult to define The truth value of a statement can span between 0 and 1, and it is not limited to the binary values of true or false. For instance, the statement that "rain" is a commonly observed natural phenomenon acknowledges its potential for significant variability. Rainfall can range from light drizzles to intense downpours, exhibiting a wide spectrum of intensity.[12]

Hybrid classifiers

In the context of developing an Intrusion Detection System (IDS), the main objective is to achieve the highest possible accuracy for the given task. Consequently, the design of hybrid approaches emerges as a natural strategy to enhance system performance. A hybrid classifier blends multiple machines learning techniques, leveraging their collective strengths to achieve significant improvements. Typically, a hybrid approach A combination of two functional components is involved in this process: the first component processes the raw data and generates intermediate results, while the second component takes these intermediate results as input and produces the final results.[13]

Hybrid classifiers can be created by combining different classifiers, such as using neural obfuscation techniques. Moreover, they can integrate clustering-based methods for preprocessing input samples and eliminate irrelevant training examples from each category. The results obtained from the compilation can be used as training examples for subsequent classifier design. Thus, the initial primitive level of merged taxonomies can be generated via supervised flagging or untracked flagging.

Ensemble classifiers

Datasets have been introduced so as to better perform individual classifiers. The group concept refers to the combination of multiple weak learning algorithms, also known as weak learners. These weak learners are trained on various training samples, allowing them to make more effective predictions.

There are different strategies for combining poor learners, but one of the most common methods is the "majority vote" method. In this method, the bad learner is given a prediction, and the final answer is taken into the classification process based on the majority vote combination of all the poor learners.

In addition to majority voting, there are other fusion techniques such as reinforcement and mobilization. Both methods involve re-sampling the training data and then pooling the predictions generated by the weak learners. This assembly is usually done using majority vote.

In general, group classifiers have been shown to be more successful in improving classification performance than using individual classifiers alone. Combining various weak learners with integration techniques such as majority voting can lead to more accurate and robust classifications.[14]

COMPARISONS OF RELATED WORK

The mechanisms used to find out about breaches can be categorized into three groups: single classifiers, hybrid classifiers, and batch classifiers. To gain insight into the different types of classifier designs, Table 1 provides an overview of articles that use single, group, and mixed classifiers. In addition, Figure 1 shows the distribution of these articles over the years based on their classifier design.

Examination of Table 1 reveals that single classifiers were the most prevalent in the literature. Conversely, only a limited number of studies looked at group classifiers, although they could outperform single-data data through robust classification performance.

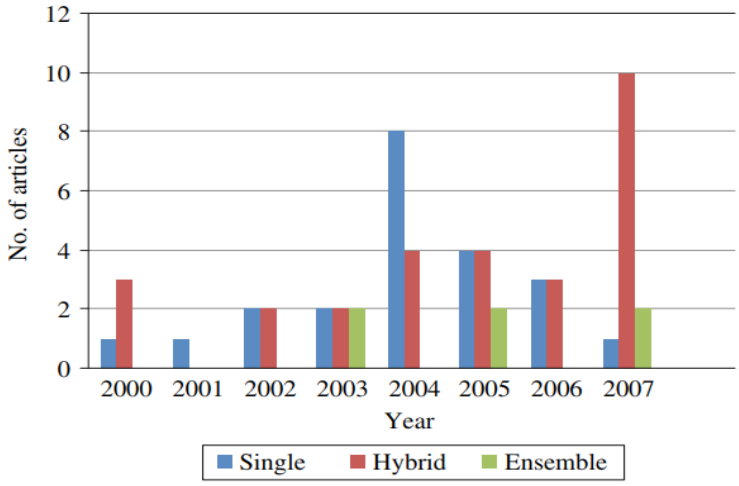

Figure 1. Yearwise distribution of articles for the types of classifier design

Figure 1 presents the number of research studies per year. It was known that the use of individual methods reached its peak in 2004, followed by a gradual decline thereafter. Due to advances in intrusion detection, it has become increasingly difficult to design a single approach that goes beyond existing approaches. However, hybrid approaches have gone from being marginal to mainstream in recent years. This transition is supported by ten research publications focusing on mixed approaches.

In 2007, there was only one research paper using a single intrusion detection method. However, it is noteworthy that during the same year, ten publications focused on mixed approaches. Undoubtedly, hybrid methods provide greater flexibility and thus have gained popularity in recent times.

For studies centered around the design of single classifiers, Table 2 presents the total number of articles implementing different classification techniques such as Support Vector Machines "SVM", Multilayer Perceptron "MLP", etc. Furthermore, Figure 2 shows the distribution of these articles over years, categorized by advanced classifiers.

Among the techniques used for individual intrusion detection, K-Nearest Neighbors "K-NN" and "SVM" stand out as the most widely used. This observation indicates that 'SVM' is increasingly being considered in single classifier design, although the number of samples compared in this context is limited. On the other hand, probabilistic rules "fuzzy logic" and self-organizing maps "SOM" have not been extensively researched in the field of intrusion detection.

In terms of hybrid workbooks, since there are three distinct strategies for their design, Table 3 quantifies the total number of articles published for each type. Furthermore, Figure 3 provides an annual distribution of these materials, based on the design of multiple 'hybrid' classifiers.

Table 1. Total numbers of articles for the types of classifier design

	Single	Hybrid	Ensemble
No. of articles	26 Balajinath and Raghavan (2000), Bouzida et al. (2004), Chen et al. (2005), Chimphlee et al. (2006), Depren et al. (2005), Eskin et al. (2002), Fan et al. (2004), Heller et al. (2003), Li and Guo (2007), Liao and Vemuri (2002), Mukkamala et al. (2004), Peddabachigari et al. (2004), Ramos and Abraham (2005), Schultz et al. (2001), Scott (2004), Shyu et al. (2003), Tian et al. (2004), Wang and Stolfo (2004), Wang and Battiti (2006), Wang et al. (2004), Wang et al. (2006), Zhang and Shen (2005)	23 Abadeh et al. (2007), Bridges and Vaughn (2000), Chavan et al. (2004), Chen et al. (2007), Depren et al. (2005), Eskin et al. (2002), Florez et al. (2002), Giacinto and Roli (2003), Jiang et al. (2006), Joo et al. (2003), Kayacik et al. (2007), Khan et al. (2007), Lee and Stolfo (1998, 2000), Liu and Yi (2006); Liu et al. (2007, 2004), Luo and Bridgest (2000), Moradi and Zulkernine (2004), Ozyer et al. (2007), Peddabachigari et al. (2007), Shon et al. (2006), Shon and Moon (2007); Stein et al. (2005), Toosi and Kahani (2007), Tsang et al. (2007), Xiang and Lim (2005), Zhang et al. (2005), Zhang et al. (2004)	6 Abadeh et al. (2007), Giacinto et al. (2006, 2008), Giacinto and Roli (2003), Han and Cho (2003), Kang et al. (2005), Mukkamala et al. (2005); Peddabachigari et al. (2007)

The results indicate that "integrated-based hybrid classifiers" were the most popular approach for intrusion detection, especially in 2007. Cascaded hybrid classifiers also received significant attention in the literature. However, because there is so little research on "group workbooks" that they are not discussed with this study.

https://doi.org/10.56294/dm202271

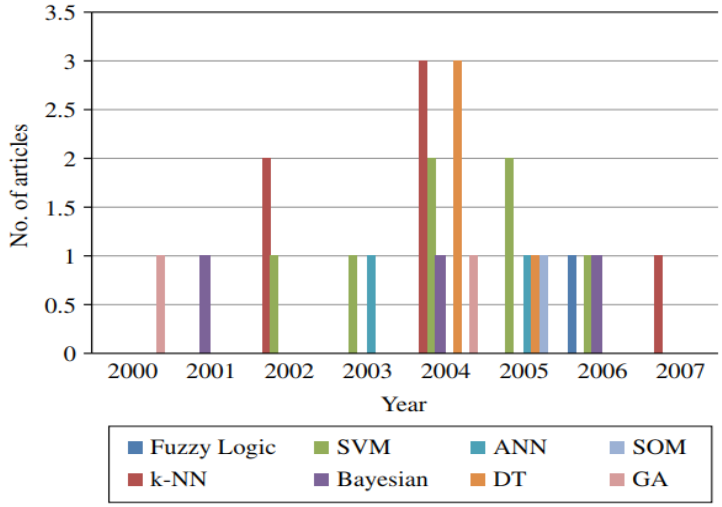

Figure 2. Articles according to annual publication and distribution of individual works

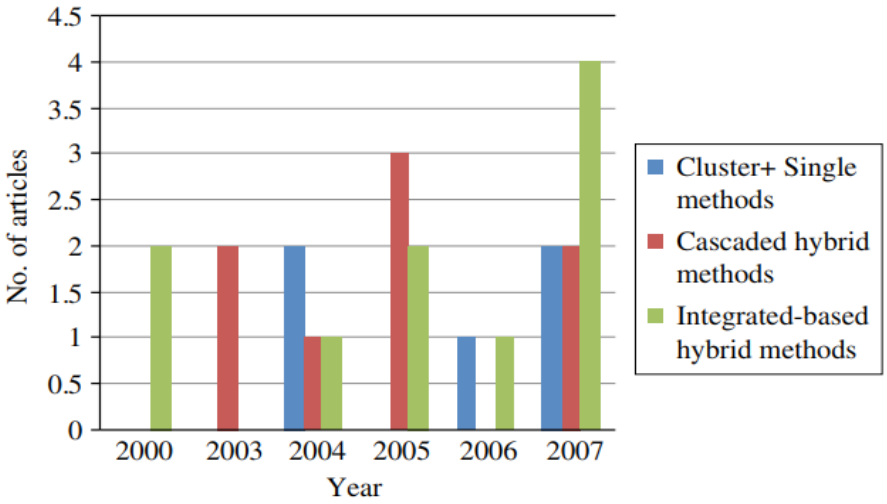

Figure 3. Rotation of articles based on year for hybrid classes

The basics classifiers

The basic classifiers differ between different studies, with each work selecting different classifiers to validate intrusion detection systems. SVM is the most widely used baseline technology and has also been considered in recent model comparisons. And in the case of aggregate classifiers, many of the core classifiers depend on the individual classifiers mentioned above.

Datasets

Public datasets such as KDD'99, DARPA 1998, and DARPA 1999 are commonly used in intrusion detection experiments, while only a handful of studies make use of proprietary or customized datasets. This demonstrates the acknowledged importance of these publicly available datasets as normative benchmarks within the field. [15,16]

Pick the best feature

Discriminatory profiling is not carried out consistently in the pre-experienced stage of the classifier. While 26 trials touched on picking and taking the best advantage, 30 trials did not. Nine studies in 2007 used a diverse method to take best-selection of features, suggesting that their-selection can yield somewhat better classification results in detecting parasitism.

CONCLUSION

In conclusion, the reviewed studies examined machine learning techniques for intrusion detection, with a particular focus on papers published between 2000 and 2007. The study looked at a wide range of "combined, combined, and one-sided" workbooks in the field of intrusion detection. However, more research is needed to

develop intrusion detection systems using machine learning techniques. Possible future research areas include conducting comparisons between overlapping classifiers and equidistant groups with a common factor is correct future detection, exploring the structure of multiple classifiers by combining clustered and mixed classifiers, and investigating the performance of different feature selection methods according to “techniques”. Different classifications in order to know and detect security breaches.

REFERENCES

1. Abadeh MS, Habibi J, Barzegar Z, Sergi M. A parallel genetic local search algorithm for intrusion detection in computer networks. Eng Appl Artif Intell. 2007;20:1058–69.

2. Agarwal R, Joshi MV. A new framework for learning classifier models in data mining. Department of Computer Science, University of Minnesota; 2000.

3. Anderson J. An introduction to neural networks. Cambridge: MIT Press; 1995.

4. Balajinath B, Raghavan SV. Intrusion detection through behavior model. Comput Commun. 2000;24:1202–12.

5. Bishop CM. Neural networks for pattern recognition. Oxford: Oxford University Press; 1995.

6. Bouzida Y, Cuppens F, Cuppens-Boulahia N, Gombault S. Efficient intrusion detection using principal component analysis. In: Proceedings of the 3eme conference sur la securite et architectures reseaux (SAR). Orlando, FL, USA; 2004.

7. Bridges SM, Vaughn RB. Intrusion detection via fuzzy data mining. In: Proceedings of the twelfth annual Canadian information technology security symposium. Ottawa, USA; 2000.

8. Chavan S, Shah KDN, Mukherjee S. Adaptive neuro-fuzzy intrusion detection systems. In: Proceedings of the international conference on information technology: Coding and computing (ITCC’04); 2004.

9. Chen Y, Abraham A, Yang B. Hybrid flexible neural-tree-based intrusion detection systems. Int J Intell Syst. 2007;22:337–52.

10. Chen WH, Hsu SH, Shen HP. Application of SVM and ANN for intrusion detection. Comput Oper Res. 2005;32:2617–34.

11. Chimphlee W, Addullah AH, Sap MNM, Srinoy S, Chimphlee S. Anomaly-based intrusion detection using fuzzy rough clustering. In: Proceedings of the international conference on hybrid information technology (ICHIT’06); 2006.

12. Depren O, Topallar M, Anarim E, Ciliz MK. An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. Expert Syst Appl. 2005;29:713–22.

13. Ertoz L, Eilertson E, Lazarevic A, Tan PN, Dokas P, Kumar V, et al. Detection and Summarization of Novel Network Attacks Using Data Mining.

14. Eskin E, Arnold A, Prerau M, Portnoy L, Stolfo S. A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data. Kluwer; 2002.

15. Rincon Soto IB, Sanchez Leon NS. How artificial intelligence will shape the future of metaverse. A qualitative perspective. Metaverse Basic and Applied Research. 2022. 27];1:12. https://doi.org/10.56294/mr202212.

16. Fan W, Lee W, Miller M, Stolfo SJ, Chan PK. Using artificial anomalies to detect unknown and known network intrusions. Knowl Inf Syst. 2004;507–27.

FUNDING

No financing.

CONFLICT OF INTEREST

None.

AUTHORSHIP CONTRIBUTION

Conceptualization: Ibraheem Khalil Ibraheem.
Research: Ibraheem Khalil Ibraheem.
Methodology: Ibraheem Khalil Ibraheem.
Writing - original draft: Ibraheem Khalil Ibraheem.
Writing - revision and editing: Ibraheem Khalil Ibraheem.

https://doi.org/10.56294/dm202271