International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Hybrid Feature Selection and Classification using RF-DNN for Anomaly Detection in IoT-WSN

[1*]Dabbara Jayanayudu, [2]A. Ch. Sudhir

Submitted: 03/11/2023 **Revised**: 24/12/2023 **Accepted**: 04/01/2024

Abstract: There are a number of factors that may impact how an attack detection system can identify a threat. It is clear that current Intrusion Detection System (IDS) approaches for the Internet of Things (IoT) are still in their youth. There are just a few ways to categorise attack types. However, only conventional networks have applied and assessed such techniques. Due to this, the IoT-specific needs and computational capabilities of these approaches were not taken into account while developing these methods. In this paper, hybrid feature selection and classification using Random Forest-Deep Neural Networks (RF-DNN) for anomaly detection technique in Internet of Things (IoT) Wireless Sensor Networks (WSN) is proposed. In this technique, a filtering method of Fisher' score and correlation coefficient is applied to select the candidate feature set. Then the combination of RF and DNN is used as the classifier for feature selection. The static properties are divided into five primary categories. Similarly, it categorise the dynamic features into the classes of location, network, protocol, registry and Internet Protocol (IP) address. Experimental results show that the proposed RF-DNN algorithm achieves higher detection accuracy, higher throughput, lesser computational cost and higher residual energy, when compared to the existing techniques.

Keywords*: Anomaly detection, Feature selection, Classification, Random Forest-Deep Neural Networks (RF-DNN), Wireless Sensor Networks (WSN)*

1. Introduction

People and other equipment are being connected together through IoT, a worldwide technology that delivers a variety of Internet-based services. WSN is a distributed network of sensors that gather environmental data and transmit it to a preset point for further processing. An IoT application cannot function without a WSN. [1]

Perimeter defences degrade as the number of IoT devices grows dramatically, increasing the amount of unknown vulnerabilities and threats. When applied to IoT ecosystems, traditional anomaly detection methods are rendered useless by IoT devices' broader and more dynamic set of probable typical behaviour patterns. An Intrusion Detection System (IDS) monitors all network activity and user behaviour to determine whether there are any suspicious activities or breaches of the given policy [2.].

Artificial intelligence (AI) refers to computer systems that mimic human intelligence in decision-making processes. Machine learning (ML), Deep Learning (DL), and robotic process automation are all part of this system. ML has been hailed as a milestone in AI research. Unsupervised, semi-supervised, and supervised approaches are the types of ML techniques.

An IoT-based WSN's IDS might use ML techniques to identify anomalous user behaviour and traffic patterns.

1.1 Problem Identification and Objectives

There are a number of factors that may impact how an attack detection system can identify a threat [3]. It is clear that current IDS approaches for the IoT ecosystem are still in their youth. There are several ways that concentrate on identifying individual cyber-attacks, rather than categorizing the sort of assault. The ability to apply particular countermeasures for certain attack types is a key feature of an IDS.

There are just a few ways to categorise attack types at this time. However, only conventional networks have applied and assessed such techniques. Due to this, the IoT-specific needs and computational capabilities of these approaches were not taken into account while developing these methods.

1.2 Objectives

Hence the objectives of the work is to

Design optimal feature selection method such that the outliers and redundant data are removed to give enhanced classification accuracy with reduced false positives.

[1]*Research Scholar, Dept. of EECE, GST GITAM (DEEMED TO BE UNIVERSITY), Visakhapatnam*
*Corresponding Author: *d.jayanaidu@gmail.com@gmail.com*
[2]*Associate Professor, Dept. of EECE, GST*
[2]*GITAM (DEEMED TO BE UNIVERSITY), Visakhapatnam*
camanapu@gitam.edu

Develop a ML algorithm to classify the selected features and detect the malicious behaviour of users .

1.3 Proposed Contributions

In order to meet the above said objectives, Hybrid Feature Selection and Classification using RF-DNN for anomaly detection technique is proposed.

The major novelty and contributions of this paper are summarized as follows:

The Fisher' score and correlation coefficient based filtering technique is applied to select the candidate feature set.

The RF-DNN model is applied as the classifier for the feature selection.

The static and dynamic features are categorized into distinct classes.

This paper is organized as follows: Section 2 contains the related works, section 3 contains the proposed solution, section 4 contains the experimental results and section 5 concludes the paper.

2. Related Works

Simone Facchini et al [4] have suggested a new multi-level Distributed IDS in a Smart Home atmosphere. The suggested method aims to perceive unpredicted actions of a network module by using the association amid the diverse IoT manoeuvres. The solution has been provided by implementing a distributed hash table (DHT)-based design that enables network and system information sharing among the nodes. A disseminated IDS situated in every node of the network, signifies the essential module to sense malevolent conduct. The suggested system outfits a dualistic classifier centred on a machine learning mechanism. It analyses the accumulation of structures mined from data approaching. . But it failed to distinguish between the attack types.

Yulong Fu et al [5] have suggested a constant intrusion recognition technique for the massive varied IoT networks. Their technique utilises an allowance of branded transition models to suggest a constant depiction of IoT systems and can notice the interferences by associating the preoccupied movement's flows. They intended the IDS method, constructed the tables, and applied the analyzer to attain the IDS methods. They also intended attesting atmosphere to authenticate the suggested IDS technique and observe the attack of RADIUS application. But it requires specific hardware and software features to detect the anomalies.

An example of an intellectual faith calculation based on ML has been made commercially available. Multi-class Support Vector Machine (SVM) is used to classify reliable and malicious communications in this software K-Means grouping is used to organise the communications. It also involves a great degree of computational complexity since it incorporates both unverified and validated techniques of learning how to recognise trustworthy transactions.

Distributed Denial of Service (DDoS) attacks have been industrialised with the help of an ML framework.[7]. The congestion detention procedure gathers numerous topographies of congestion. The gathered topographies are clustered and have been removed based on the IoT performances. Lastly, certain dualistic cataloguing methods were used precisely to differentiate normal congestion from DDoS about congestion.

In anomaly detection technique for IoT sensors [8], Logistic Regression, SVM , ANN, Decision Tree and RF algorithms are used. To determine which method was more accurate in training as well as testing, cross-validation was used. However it did not include any feature selection methods

DL algorithms used in intelligent IDS [9] have been developed to identify malicious traffic on IoT networks. However, there is a significant amount of communication overhead associated with the connection probe module.

The Passive Infrared Sensor (PIR) was replaced with IoT based IDS [10], since the PIR sensor is difficult to trace. In the event of an intrusion, the IDS will notify the next customers. However it is used in real-time applications to notify the occupant of the house about intrusion of an unknown person.

Data collection, feature extraction, and binary classification are all performed using a ML approach designed [11] for detecting DDoS attacks on IoT traffic. based on network and flow patterns, the features are chosen. They have applied this technique over a consumer IoT device network. But it involves huge processing overhead.

The filter method's output is utilised as an input to the wrapper method in [12] to improve the classifier's performance in a hybrid approach to feature selection. The filter model is used to pick the candidate subset in the Filter method, whereas the classifier model is used in the Wrapper method to assess the Filter model. RF and K-nearest neighbour (k-NN) classifiers were utilised in this study.

3. Proposed Methodology

3.1 Overview

In this paper, Hybrid Feature Selection and Classification using RF-DNN for Anomaly Detection technique is

proposed. In our proposed method, first Filter method using Fisher' score and correlation coefficient is applied to select the candidate feature set. Then we use the combination of random forest and deep learning (RF-DL) model [13] as the classifier for the feature selection. It categorises the both the static and dynamic features into five main classes, separately.

Static Features

The static attributes are divided into five categories, as shown in Table 1: size, count (entropy), IAT information, and entropy. There are a total of 79 static features in these five feature types. There are also five feature classes for dynamic features (see Table 2). Table 1 shows a summary of our features.

Entropy information, for instance, can be used to improve detection rates by changing the counts as well as size, as shown in Table 1. Using these traits, we were able to discover other features in malware that were previously unseen. When it comes to classification models, these characteristics may provide even better results (see Table 3). Static characteristics include the size of file sections and particular file parts like data, text, bss, or header, as well as features from prior malware. Size is the most common atomic unit used to choose features. The API and DLL amounts are also included to enhance the static features. Entropy, entry point, and the amount of IAT mentioned are also included. Specific API calls are triggered by malware, and the IAT information they left over may be utilised to tell suspicious program apart from benign ones. In spite of this, malware is often misclassified due to attacks that resemble innocuous files in order to pass for the real thing. To illustrate this point, malicious files that have been obfuscated using packers appear to be similar to one other when it comes to the length of each part of the file, the number of file sections, or the headers. Many other types of malware have these characteristics, so you're likely to run against them elsewhere. As a result, we've included methods that are more dynamic. Such features may be detected by examining the API methods that are used to perform the suspicious actions (such as creating files, accessing networks, or editing registry keys). Detailed information is provided in the next sections.

Table 1: Static Features

Class	Features	Explanation
Size (byte)	File	File size
	Headers	Header Size
	InitData	Initialized data size (.data section)
	UninitData	Uninitialized data size (.bss section)
	Text	Text section size
	Debug	Debug section size
	Rsrc section	Size of resource folder
Count	API	Doubtful APIs count
	DLL	Accessed DLL counts
	FSection	File sections count
	RVA Sizes	Data-folder entries' count
	Language	Languages utilized in resource section
Entropy	Entr_Data	Data section's entropy
	Entr_rData	rdata section's entropy
	Entr_reloc	reloc section's entropy
	Entr_text	Text section's entropy
	Entr_rsrc	rsrc section's entropy
Entry Point	En_point	Entry point gathered
IAT	API_Func	Predefined API functions

Dynamic Features

There are limitations with static features that make it impossible to see malicious behaviour. Dynamic analysis is a superior method for using behavioural data, but it is also more time demanding. Such situations need the use of dynamic feature extraction, rather than the use of static features.

Table 2: Dynamic Features

Feature Class	Features	Explanation
API	File system varies	Predefined file system changes (copy, rename, delete)
	DLL loaded info	# suspicious file change; create, write, rename, delete
	API Call	Suspicious API call
Location	File varies in suspicious position	(Loaded) DLL position, in which paths are defined already
	Suspicious directory access	Categorized by symbolized mutex
	Suspicious registry access	# suspicious directory access
	Suspicious DLL location	Registry routes utilized by malware
Registry	Command running	Registry varies (create, read, modify, delete)
	Suspicious registry	Predefined file system varies (copy, rename, delete)
Network	Network (by winsock DLL)	Persistency (exact command running while rebooting)
Mutex	Mutex based features	N/W open, outbound access and malicious IP spaces

We used a variety of to differentiate between malicious and normal features.. When it comes to determining how important a feature is, Table 2 is a good starting point. File system modifications indicate changes in the amount of files that we have defined, which means that files are either created, copied, or deleted.

3.4 Filter based Feature selection method

Filter and wrapper techniques were often used to narrow down the list of potential features. The biased power of each feature is assessed using statistical criteria in the filter approach. When it comes to evaluating numerical characteristics, Fisher's score is the best option, whereas correlation coefficient evaluates the linear connection between two variables. Features are selected depending on their output. In the wrapper method, machine learning is utilised to decide which feature sets picked by a particular feature search algorithm are the most relevant.

In order to develop a decision tree model, the best characteristics are selected using filtering methods. It is used in this study to pick a candidate feature set using Fisher's score and correlation coefficients.

Initially, the proportion of the average separation between groups to the one within classes (Fisher's score) is calculated for each numeric characteristic [11]. Among the different features of IoT traffic, the 20 best host-centric characteristics (e.g., Host-IP and Host-MAC&IP categories) are chosen by F. In the case of DDoS assaults that create a lot of traffic, host-centric data would show the disparities between different classes of attacks.

The correlation coefficient (CC), which comes after the first factor (F1), is a metric for determining the linear connection between two variables. Every feature has its pairwise CCs calculated. As a result, any characteristics having a CC greater than 0.80 were eliminated. There are three host-centric features and 15 host-to-host communication features among the 18 chosen features (such as channel, channel jitter and socket categories).

Classification using RF-DNN

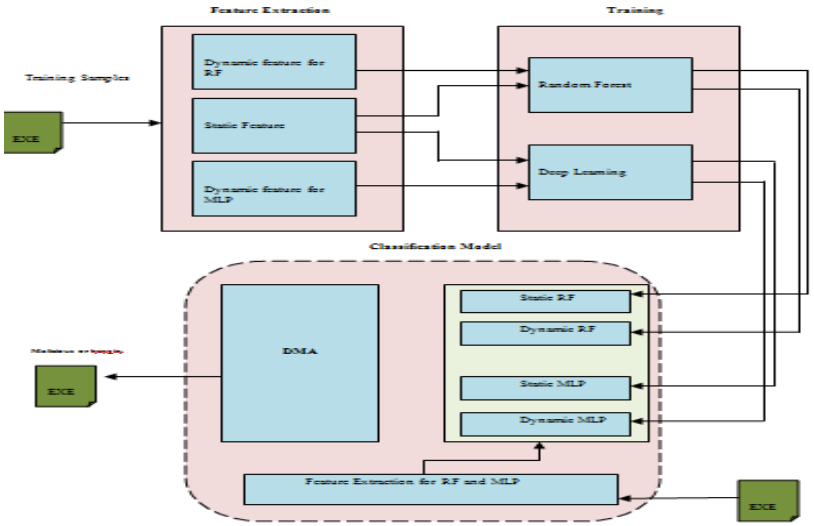

Fig 1: Classification using RF-DNN

We utilised a hybrid model based on machine learning to come up with the best practicable solution. The whole system design is depicted in Figure 1. We divided the detection procedure into three steps using RF and Multi-Layer Perceptron (MLP), as shown in Fig. 1.

Four categorization models in our model are used to gauge the level of maliciousness. Maliciousness in original files is expressed in a scoring range of [0,1] in a standard ML method. When it comes to hybrid models, RF and MLP, the suggested model uses four scores from the hybrid model to provide optimization outcomes by applying our voting model to those four scores, RFstatic, RFdynamic; MLPstatic and MLPdynamic, respectively.

In the first step, the feature extraction stage r collects both static and dynamic features from candidate files. Four alternative categorization models may be constructed as a result of this, each based on a different set of four feature sets.. A majority vote mechanism based on the findings is then used to establish the final decision values.

In spite of the fact that a simple majority voting rule that chooses alternatives with a majority may identify malware effectively, it is quite likely to overlook the classifier that accurately detects benign files with malicious behaviour.

The default parameters of our optimizer, RMSprop, were utilised in the suggested model. keras.optimizers's default value.. Our MLPdynamic employed 39 nodes, 1072 batch sizes, 12 hidden layers, 87 epochs, RMSprop as the optimizer, and 432 features in these studies. 18 nodes, 1520 batch sizes, seven hidden layers, 103 epochs, adamax as the optimizer, and 79 features were utilised in MLPstatic. The activation functions in both classification models were ReLu and softmax. We used a cross-entropy cost function instead of the mean squared error (MSE) cost function to prevent the learning slowdown caused by the (z) term.

In Table 3, RF stands for Random Forest and MLP is a deep learning classifier used in the decision-making process. the harmful probability value of an executable predicted by a RF classifier with static characteristics attached to that is called RFstatic in this rule set.

Table 3: Rules for Decision Making

Rule 1	If RFstatic <=0.5 then return “benign”
Rule 2	If RFdynamic <0.5 then return “benign”
Rule 3	If RFstatic < 0.5 or RFdynamic <0.5 then return “benign”
Rule 4	If RFdynamic < 0.5 or MLPdynamic <0.5 then return “benign”

4. Simulation Results

4.1 Simulation Parameters

The proposed RF-DNN for IoT-WSN is implemented in NS2. The performance of RF-DNN is compared with RF-KNN approach and DL-IDS [3] in terms of the metrics detection accuracy, throughput (in KB/s), computational cost (in seconds) and residual energy (in Joules). Table 4 displays the settings for the simulation.

Table 4: Simulation parameters

Number of Nodes	50
Size of the Topology	150m X 150m
MAC Protocol	802.15.4
Attack Interval	10 to 50 sec
Traffic Source	Exponential
Packet size	512bytes
Propagation	Two Ray Ground
Antenna	Omni Antenna
Initial Energy	10 Joules
Transmission Power	0.3 watts
Receiving Power	0.3 watts
Attack Frequency	25Kb to 125Kb

Effect of Attack Intervals

In this section, the results of varying the attack interval from 10 to 50 seconds are presented.

Table 5: Results for Detection Accuracy (Intervals)

Interval (sec)	RF-DNN	RF-KNN	DL-IDS
10	0.9765	0.9684	0.9635
20	0.9918	0.9728	0.9689
30	0.9945	0.9816	0.9752
40	0.9961	0.9841	0.9788
50	0.9975	0.9916	0.9843

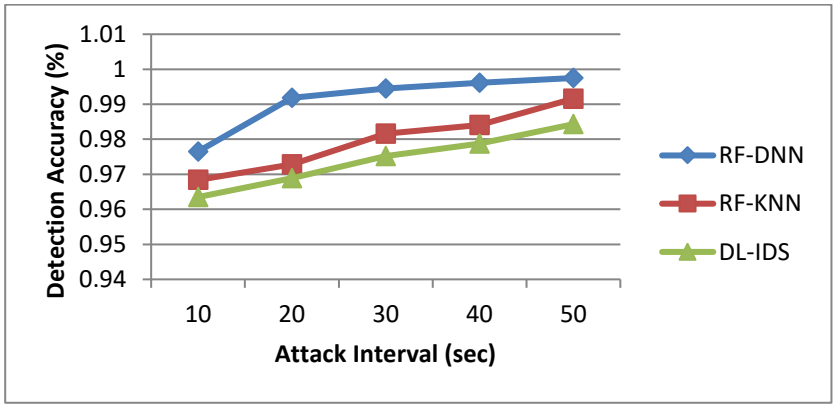

Fig 2: Detection Accuracy Vs Attack intervals

Table 5 and Figure 2 show the results of detection accuracy for varying the attack interval. From the figure, the detection accuracy of RF-DNN is 1.5% high when compared to RF-KNN and 1.7% higher than DL-IDS.

Table 6: Results for Throughput (Intervals)

Interval (sec)	RF-DNN (KB/s)	RF-KNN (KB/s)	DL-IDS (KB/s)
10	127.66	86.14	120.36
20	242.6	174.61	182.41
30	363.18	259.42	227.8
40	477.74	339.7	362.22
50	606.55	424.38	492.51

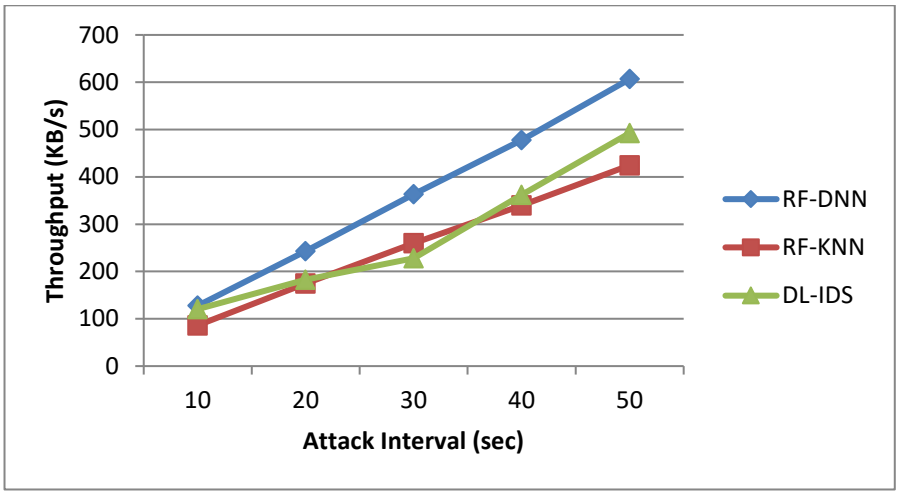

Fig 3: Throughput for attack intervals

Table 6 and Figure 3 show the results of throughput for varying the attack interval. From the figure, the throughput of RFDNN is 30% higher when compared to RFKNN and 22% higher than DL-IDS.

Table 7: Results for computational Cost (Intervals)

Interval (sec)	**RF-DNN (sec)_**	**RF-KNN (sec)**	**DL-IDS (sec)**
10	0.266	0.274	0.262
20	0.405	0.409	0.412
30	0.501	0.544	0.528
40	0.622	0.669	0.652
50	0.813	0.877	0.834

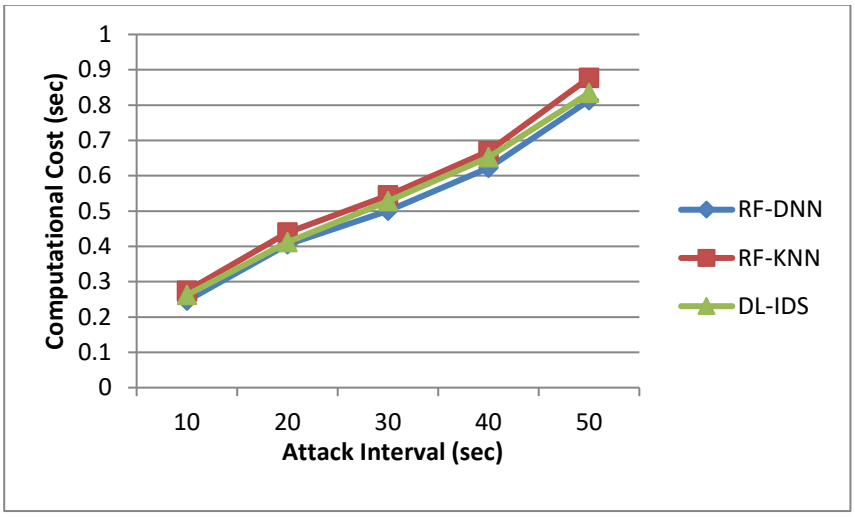

Fig 4: Computational cost for attack intervals

Table 7 and Figure 4 show the results of computational cost for varying the attack interval. From the figure, the computational cost of RFDNN is 5% lesser than RFKNN and 4% lesser than DL-IDS.

Table 8: Results for Residual Energy (Intervals)

Interval (sec)	RF-DNN (Joules)	RF-KNN (Joules)	DL-IDS (Joules)
10	11.56	11.50	11.52
20	11.33	11.22	11.3
30	10.98	10.84	10.9
40	10.60	10.45	10.54
50	10.49	10.15	10.28

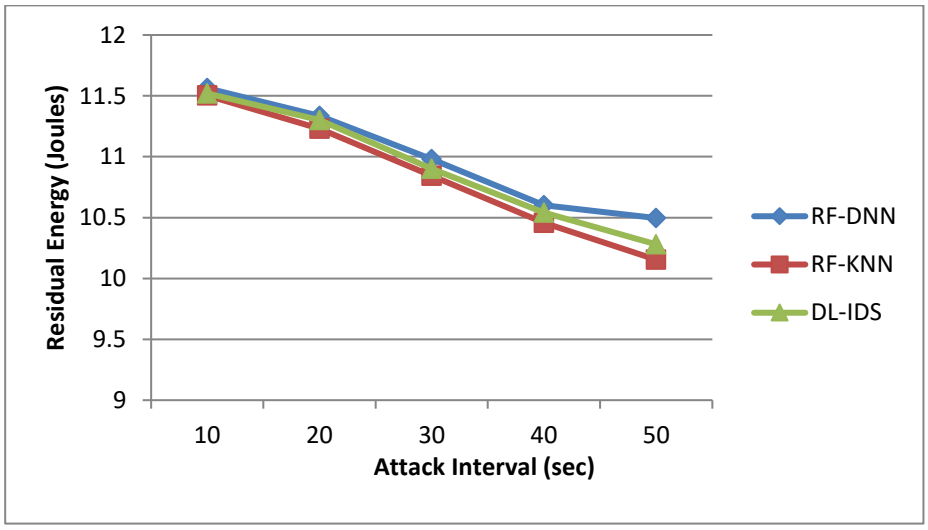

Fig 5: Residual Energy for attack intervals

Table 8 and Figure 5 show the results of residual energy for varying the attack interval. It can be seen that the residual energy of RFDNN is 1% higher than RFKNN and 1% higher than DL-IDS.

Effect of Attack Frequency

The results of changing the attack frequency from 50 to 150 Kb are shown in this section.

Table 9: Results for Detection accuracy (Frequency)

Attack Frequency (Kb)	RF-DNN	RF-KNN	DL-IDS
25	0.9765	0.9684	0.9635
50	0.9714	0.9622	0.9522
75	0.9674	0.9565	0.9475
100	0.9636	0.9514	0.9417
125	0.9567	0.9434	0.9384

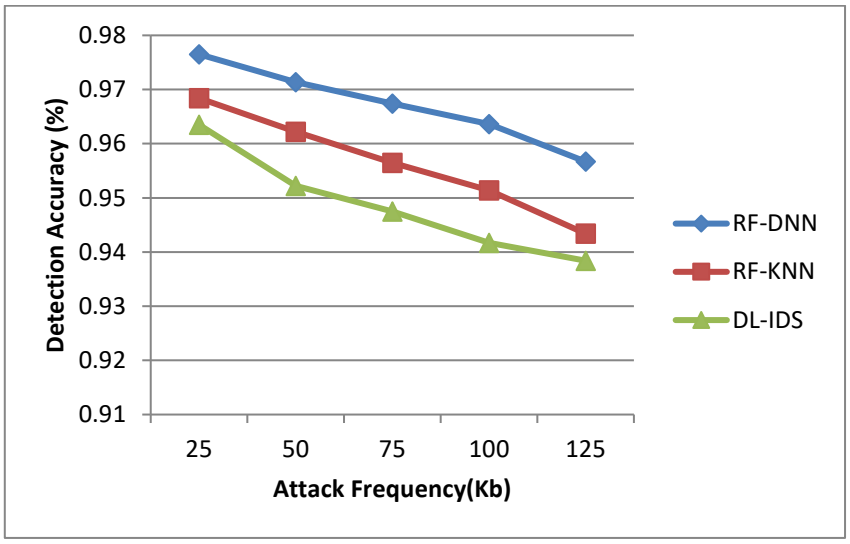

Fig 6: Detection accuracy for Attack frequency

Table 9 and Figure 6 show the results of detection accuracy for varying the attack frequency. It can be seen from the figure, the detection accuracy of RFDNN is 1.1% high when compared to RFKNN and 2% higher than DL-IDS.

Table 10: Results for Throughput (Frequency)

Attack Frequency (Kb)	RF-DNN (KB/s)	RF-KNN (KB/s)	DL-IDS (KB/s)
25	127.66	86.14	120.36
50	125.36	85.80	115.62
75	121.85	84.15	110.55
100	116.61	82.73	107.51
125	112.78	82.40	102.38

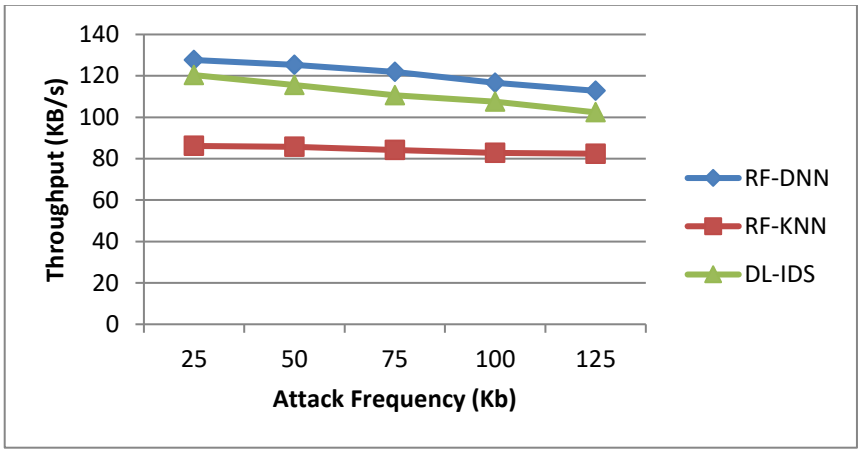

Fig 7: Throughput for Attack frequency

Table 10 and Figure 7 show the results of throughput for varying the attack frequency. It can be seen from the figure, the throughput of RFDNN is 30% higher than RFKNN and 8% higher than DL-IDS.

Table 11: Results for Computational cost (Frequency)

Attack Frequency (Kb)	RF-DNN (sec)	RF-KNN (sec)	DL-IDS (sec)
25	0.429	0.462	0.435
50	0.431	0.482	0.441
75	0.438	0.526	0.453
100	0.451	0.531	0.464
125	0.441	0.561	0.47

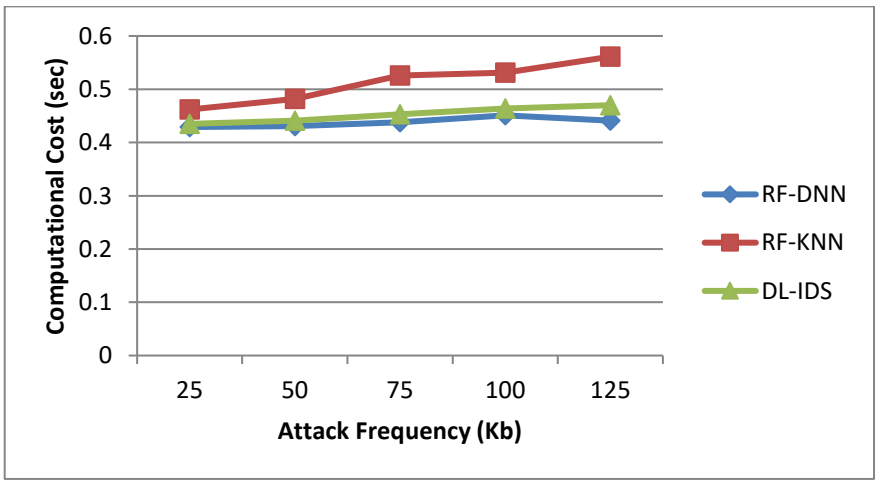

Fig 8 Computational cost for Attack frequency

Table 11 and Figure 8 show the results of computational cost for varying the attack frequency. It can be seen from the figure, the attack frequency of RFDNN is 14% lesser than RFKNN and 3% lesser than DL-IDS.

Table 12: Results for Residual energy (Frequency)

Attack Frequency (Kb)	RFDNN (Joules)	RFKNN (Joules)	DL-IDS (Joules)
25	10.87	10.67	10.75

50	10.78	10.55	10.71
75	10.62	10.44	10.54
100	10.52	10.27	10.48
125	10.48	10.17	10.42

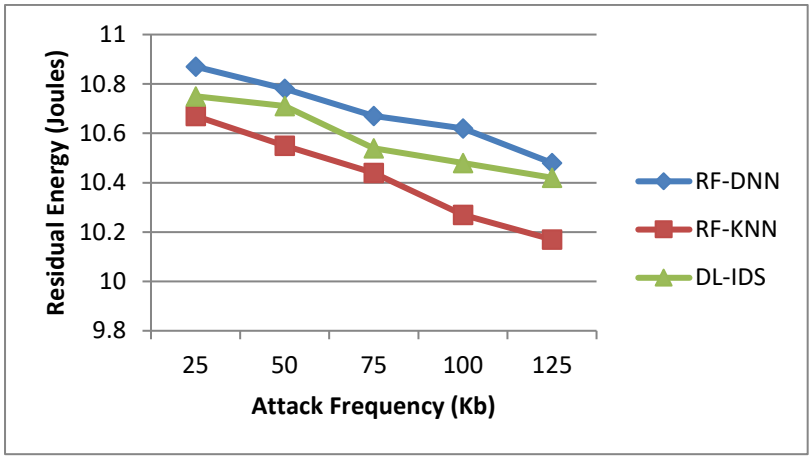

Fig 9: Residual energy Vs Attack frequency

Table 12 and Figure 9 show the results of residual energy for varying the attack frequency. It can be seen from the figure, the residual energy of RFDNN is 2% high when compared to RFKNN and 1% higher than DL-IDS.

5. Conclusion

In this paper, Hybrid Feature Selection and Classification using RF-DNN for Anomaly Detection technique is proposed in IoT-WSN. In our proposed method, first Filter method using Fisher' score and correlation coefficient is applied to select the candidate feature set. Then we use the RF-DL model as the classifier for the feature selection. The static features are divided into five major kinds. The dynamic features are similarly divided into five primary categories: location, network, protocol, registry, IP, and MAC address. Simulation results show that the proposed RF-DNN algorithm achieves 1% and higher detection accuracy, 30% and 8% higher throughput, 14% and 3% lesser computational cost and 2% and 1% higher residual energy, when compared to the RF-KNN and DL-IDS techniques, respectively.

Conflict of Interest: The authors don't have any conflict of Interest.

Competing interests: The author's don't have any competing interests.

Funding: No Funding was granted for this research

Availability of data and materials: The authors didn't use any third party data.

Authors' contributions: In this manuscript preparation author 1 prepared the concept, implementation part, grammatical errors and prepared the journal formatting. Author 2 reviewed the manuscript.

References

[1] Dehua Zheng, Zhen Hong , Ning Wang and Ping Chen, "An Improved LDA-Based ELM Classification for Intrusion Detection Algorithm in IoT Application", Sensors,20, 1706; doi:10.3390/s20061706,2020.

[2] Eirini Anthi, Lowri Williams, MałgorzataSłowi´nska, George Theodorakopoulos, Pete Burnap, "A Supervised Intrusion Detection System for Smart Home IoT Devices", IEEE Internet of Things Journal,Vol-6,No-5,pp:9042-9053,2019.

[3] Yazan Otoum,Dandan Liu and AmiyaNayak, "DL-IDS: a deep learning-based intrusion detection framework for securing IoT",Trans Emerging Tel Tech,https://doi.org/10.1002/ett.3999,2019.

[4] Simone Facchini, GiacomoGiorgi, Andrea Saracino and GianlucaDini, "Multi-level Distributed Intrusion Detection System for an IoT based Smart Home Environment",6th International Conference on Information Systems Security and Privacy,Egypt,2020.

[5] Yulong Fu, Zheng Yan,Jin Cao,Ousmane Kone, and Xuefei Cao, "An Automata Based Intrusion Detection Method for Internet of Things",Hindawi Mobile Information Systems,Volume 2017, Article ID 1750637, 13 pages,2017.

[6] Upul Jayasinghe, GyuMyoung Lee, Tai-Won Um, Qi Shi,"Machine Learning based Trust Computational Model for IoT Services", IEEE

Transactions on Sustainable Computing, TSUSC, Vol-4, pages 39-51, March 2019.

[7] Rohan Doshi,Noah Apthorpe and Nick Feamster, “Machine Learning DDoS Detection for Consumer Internet of Things Devices”, ArXiv Journal Publications, May-2018.

[8] Mahmudul Hasan ∗, Md. Milon Islam , Md Ishrak Islam Zarif , M.M.A. Hashem,"Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches", Internet of Things, Elsevier, 7 (2019) 100059

[9] Geethapriya Thamilarasu and Shiven Chawla,"Towards Deep-Learning-Driven Intrusion Detection for the Internet of Things", Sensors, 27 April 2019

[10] Khirod Chandra Sahoo and Umesh Chandra Pati, “IoT Based Intrusion Detection System Using PIR Sensor”, 2nd IEEE International Conference On Recent Trends in Electronics Information-&-Communication-Technology-(RTEICT), pages-1641-1645, 2017.

[11] Rohan Doshi,Noah Apthorpe and Nick Feamster, “Machine Learning DDoS Detection for Consumer Internet of Things Devices”, ArXiv Journal Publications, May-2018.

[12] Alejandro Guerra-Manzanares, Hayretdin Bahsi, Sven N˜omm,” Hybrid Feature Selection Models for Machine Learning Based Botnet Detection in IoT Networks”, IEEE International Conference on Cyberworlds (CW), 2019

[13] Suyeon Yoo, Sungjin Kim, Seungjae Kim an dBrent Byunghoon Kang,” AI-HydRa: Advanced hybrid approach using random forest and deep learning for malware classification”, Information Sciences , Elsevier, 546 (2021) 420–435