International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 06 | June - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

AN INTELLIGENT ENSEMBLE-BASED SYSTEM THAT USES THE NETWORK INTRUSION DETECTION SYSTEM PARADIGM TO IDENTIFY AND FIGHT INTRUSIONS IN COMPUTER NETWORKS

DUVVURU JYOTHSNA[1], G VENU GOPAL[2]

[1]PG Schoolar, Dept. of Computer Science and Engineering, PBR Visvodaya Institute of Technology & Science Autonomous, Affiliated to JNTUA, Kavali, SPSR Nellore, A.P, India-5242201.
[2]Associate Professor, Dept. of Computer Science and Engineering, PBR Visvodaya Institute of Technology & Science Autonomous, Affiliated to JNTUA, Kavali, SPSR Nellore, A.P, India-5242201.

Abstract - In recent years, the Internet of Things (IoT) has undergone tremendous change, allowing people to automate mundane, everyday activities. Connecting several types of physical devices with distinct functions allows for this to be accomplished. In order to enhance intrusion detection systems, machine learning has emerged as the crucial option. As the Internet has grown in popularity and the number of suspicious activities or intrusions has accelerated, research into network intrusion detection systems (NIDS) has emerged as a pressing concern in the field of information and network security. By using a classification method, intrusion detection systems (IDS) can distinguish between normal and abnormal incoming network traffic, which is represented as a feature vector. This helps in the detection of intrusions that violate a computer network's security policies and mechanisms and compromise CIA (confidentiality, integrity, and availability). It has been noted that classification performance is negatively impacted by feature vectors with large dimensionality in practice. A novel hybrid feature selection strategy was developed to lower the dimensionality while maintaining performance. Its efficacy was evaluated on the KDD Cup'99 dataset using the classifiers Naive Bayes and C4.5. Based on the aforementioned dataset and classifiers, two sets of experiments were carried out using the full feature set and reduced feature sets obtained using four popular feature selection methods: Correlation-based Feature Selection (CFS), Consistency-based Feature Selection (CON), Information Gain (IG), Gain Ratio (GR), and the proposed method. Classifier Naıve Bayes achieved a classification accuracy of 97.5% in the first trial, whereas C4.5 achieved 99.8%. Using the IG approach, the classifiers' greatest performance (accuracy) was 99.1 and 99.8 percent in the second set of testing.

Key Words: *Intrusion detection (ID), Machine Learning (ML), Anomaly Detection (AD), Internet of Things (IOT).*

1. INTRODUCTION

The expansion of computers, networks, and network communication technologies over the last few decades has led to the meteoric rise of the Internet. Internet communication and computer networks have grown tremendously in recent years, which has led to a rise in security concerns. The Internet is constantly evolving, with new vulnerabilities being discovered and assaults happening at a rapid pace. The environment of computer networks becomes increasingly unsafe and susceptible to attacks daily as a result. These weaknesses and dangers have the potential to impede and even alter the functioning of individual, societal, governmental, and organisational endeavours. As a result, network security has emerged as a critical concern for contemporary IT infrastructure. Anomaly and abuse, whether by foreign invaders or by internal users, pose a danger to network security. When it comes to protecting network communications from both internal and external threats, network security is a must-have[1]. Data loss, tampering, destruction, and Denial of Service (DoS) assaults are all things that this safeguards the network environment against. While there are a number of security measures at your disposal, such as authentication and access control systems and peripheral protection mechanisms, none of them can help you prevent intrusions from inside your own system. Consequently, a Network Intrusion Detection and Prevention System (NIDPS) or similar system is critically necessary to prevent network intrusions. Even seemingly innocuous network data might be an entry point for malicious actors.

1.1 Intrusion detection and prevention system

The exponential rise of cyberattacks on computer networks in the last few decades has made network security an absolute must. With the increasing susceptibility of the Internet to both internal and external attacks, NIDPS has become an essential component. It fights against the exploitation, abuse, and misuse of computer network resources. The phrase "intrusion" refers to an unauthorised effort to break the security rules or processes of a computer or network, or to undermine its confidentiality, integrity, or availability. The process of keeping an eye on a computer network, analysing its events for any indications of intrusions, and then reporting on them is known as network intrusion detection (NID). Automating the NID method is the job of Network Intrusion Detection Systems (NIDS), which may be either a software programme or a hardware solution. With all the features of an Intrusion Detection System (IDS), the Intrusion Detection and Prevention System (IDPS) tries to prevent or halt any potential invasive behaviour. Dr. Dorothy Denning suggested the first model for intrusion detection in 1987, called an intrusion detection expert system, and coined the term intrusion detection (ID) in 1987. It was the base upon which the ID development was based. This method has a low false positive rate and a high Accuracy (ACC) for detecting known intrusions, but it can't handle new, unknown, or modified intrusions. An anomaly-based detection method builds a

model using typical behaviour as a foundation and then uses monitoring to spot any changes from the norm.

Although this method can identify both new and "zero days" incursions, it often has a high FPR. Combining anomaly-based and misuse-based detection approaches results in a hybrid-based detection method.

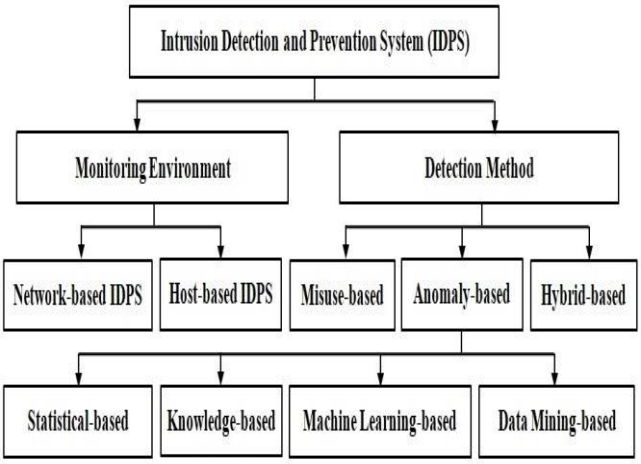

Fig -1: Classification of Intrusion Detection and Prevention System

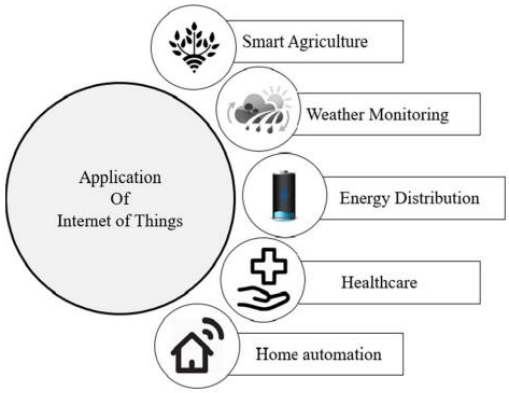

Fig -2: Applications of Internet of Things

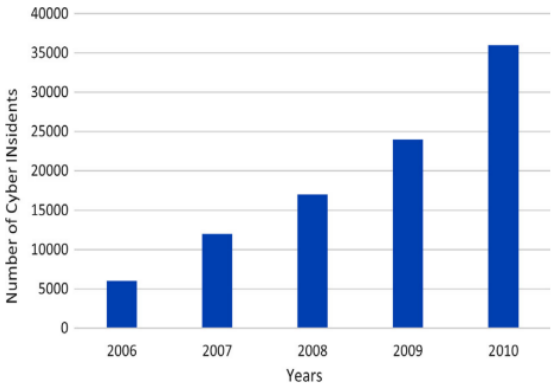

Fig -3: S-CERT: cyber incidents

1.2 KDD Cup 1999 Dataset

There is a dataset for IDS called the KDD CUP 1999. The training data set has 4,940,000 records, whereas the test data set has 311,029 records. The training set is too big, therefore we're using 10% of the KDD Cup'99 data as our experimental set. There are 494,021 connection records total; 396,744 are assault records and 97,277 are normal records. An assault type and a normal label are assigned to each link. These attacks may be classified into four types: Denial of Service (DoS), User to Root (U2R), Remote to Local (R2L), and Probing. With the addition of one class label, each connection record had forty-one features. In Table -1.1, the 41 attributes are listed in the following order: 1, 2, 3, 4, 5, 6, 7, 8, 9,.. of them, nine are symbolic and thirty-two are numerical (discrete values)[1].

1.3 Objectives

- To propose, in order to suggest a lightweight, accurate, and intelligent model for NIDS, we need to know how to identify intrusions (attacks) in real-time from network data.
- To propose, minimised computing complexity while simultaneously improving IDS performance and capabilities for real-time detection.
- To propose, tested the suggested IDS in both binary and multi-class classification settings to see how well it performs.
- To propose, in order to provide a paradigm for countering intrusions once they have been identified.
- To propose, finding suitable assessment measures to assess the suggested model's performance.

To provide the utmost security for smart grids of the future, it employs a completely dispersed management structure that supports the network. In their paper on mobile ad hoc networks, Nadeem and Howarth (2013) provide a standardised method of identification and prevention. It is a hybrid of methods that respond to invasions, identify anomalies, and focus on abuse. In order to react to intrusions and isolate malicious nodes, it used a predefined static way[2].

We suggest a mixed method that combines ID with an adaptive response mechanism. Degradation of network performance follows the selection of an intrusion response depending on the severity of the assault and the confidence of the intrusion detection system[4]. The Audit Expert System is suggested as a host-based usage detection system. An expert system is used to identify intrusions. In addition to sometimes creating and forwarding critical messages to mobile phones, it also sends reports, e-mails, and notifications to system administrators[3]. We provide a hybrid solution that combines logging with IDPS. Here, the SNORT tool is set up inline with the IPS. It checks packets for malicious activity and discards them if detected[6]. The packet that was dropped is also recorded. In response to input from both the host-based NIDPS and the network-based IDPS, the user is given the option to either quarantine risks or block data traffic from certain sources using a hybrid method[5]. To address the needs of smart grid home area networks, we provide an innovative IDPS. It uses an ID method based on models and machine learning for IPS[7]. Software-Defined Networking is used for the design and implementation of an IDPS. In particular, it protects against denial-of-service attacks and port scanning by keeping an eye out for security policy breaches and other harmful actions[8]. The detection module

determines whether the network is in a normal condition by collecting and analysing network information.

Table 1 Lists of feature number (#) and corresponding name in the KDD Cup'99

S.No	Name	S.No	Name	S.No	Name
1	Duration	15	Su-attempted	29	Same-srv-rate
2	Protocol-type	16	Num-root	30	Diff-srv-rate
3	Service	17	Num-file-creations	31	Srv-diff-host-rate
4	Flag	18	Num-shells	32	Dst-host-count
5	Src-bytes	19	Num-access-files	33	Dst-host-srv-count
6	Dst-bytes	20	Num-outbound-cmds	34	Dst-host-same-srv-rate
7	Land	21	Is-hot-login	35	Dst-host-diff-srv-rate
8	Wrong-fragment	22	Is-guest-login	36	Dst-host-same-src-port-rate
9	Urgent	23	Count	37	Dst-host-srv-diff-host-rate
10	Hot	24	Srv-count	38	Dst-host-serror-rate
11	Num-failed-logins	25	Serror-rate	39	Dst-host-srv-serror-rate
12	Logged-in	26	Srv-serror-rate	40	Dst-host-rerror-rate
13	Num-compromised	27	Rerror-rate	41	Dst-host-srv-rerror-rate
14	Root-shell	28	Srv-rerror-rate		

2. METHODOLOGY

The overall architecture of proposed anomaly based NIDPS is depicted in Figure-4. It consists of three modules—(i) Module I: Methodology for NIDS to detect normal or attack traffic, (ii) Module II: Methodology for NIDS to detect specific attack type, and (iii) Module III: Methodology for IPS to prevent from identified attack traffic.

The network traffic is entered into Module I, which aims to detect whether the incoming traffic is normal or attack. For normal traffic, it is allowed to enter into network without further processing. Whereas, for detected attack traffic, it generates alert / alarm to Module III to forbid this network traffic and invokes the Module II. Module II identifies the specific attack type (DoS, Probe, R2L or U2R) from detected attack traffic at Module I and provides this information to Module III. Module III is used to prevent the network by using the information of specific attack type provided by Module II. The details of these modules are described in the subsequent subsections[1].

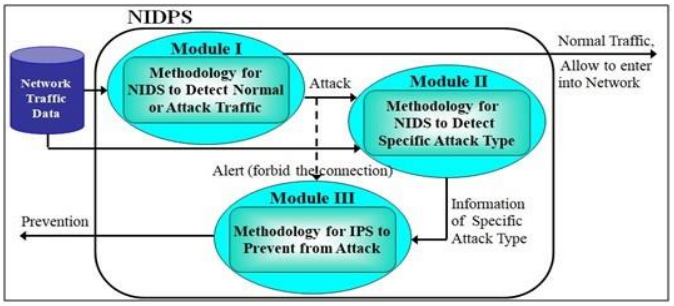

Fig -4: The Architecture of proposed anomaly based NIDPS

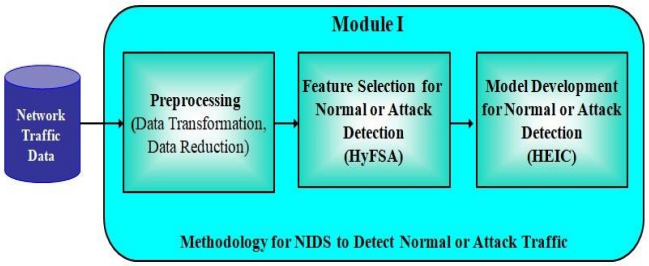

Fig -5: Module I: Methodology for NIDS for normal or attack detection

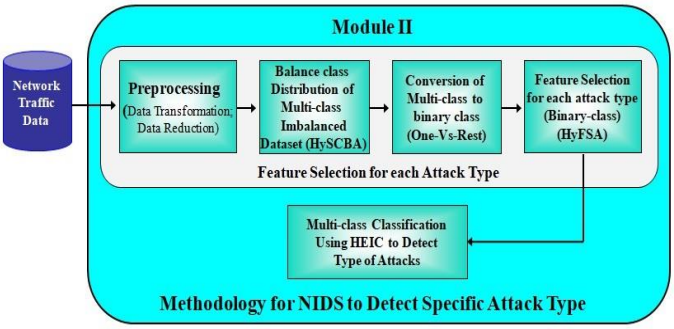

Fig -6: Module II: Methodology for NIDS to detect specific attack types

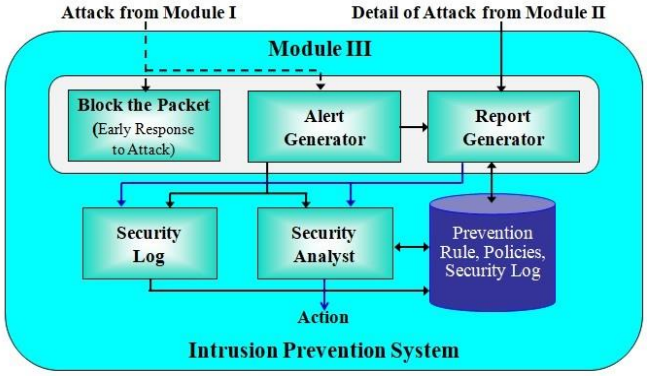

Fig -7: Module III: Methodology for IPS to Prevent from Attack

This subsection presents the performance evaluation metrics utilized to assess the performance of employed methods and proposed methodologies. There are several evaluation metrics available through which performance and effectiveness of an NIDPS can be assessed. The various evaluation metrics can be computed from confusion matrix (Wu and Banzhaf, 2010) depicted in Table-2, which contains detail about the actual and predicted classification results of a classifier.

Table 2 Confusion Matrix

Actual / Predicted	Positive Class (Attack)	Negative Class (Normal)
Positive Class (Attack)	True positive (TP)	False Positive (FP)
Negative Class (Normal)	False Negative (FN)	True Negative (TN)

The Knowledge Discovery and Data mining Cup 1999 (KDD-Cup-1999) (KDD, 1999) is the benchmark dataset for NID. It is based on the 1998 Defense Advanced Research Projects Agency (DARPA) ID Evaluation Program. DARPA'98 consists of 4 gigabytes tcpdump compressed raw (binary) data. KDD-Cup-1999 dataset is extracted from 1998 DARPA IDS Evaluation Program and it is a series of connec- tion records. It consists of 7 weeks of network traffic for training dataset ("Whole KDD"), which contains 4,940,000 records and two weeks of test data ("Corrected Test") of 311,029. The training set contains 22 attack types and the test dataset contains same 22 attacks plus 17 new attack types (Table-3). Since the train- ing dataset is too large, another 10% of KDD-Cup-1999 dataset as "10% KDD" is widely used dataset. Each record has a label of either normal or one specific attack type. The attack type falls into one of the four attacks categories as: (i) DoS, (ii) Probe, (iii) R2L, and (iv) U2R attack.

After eliminating the records, the datasets are reduced significantly, which consist of only unique records, are named as "Uni KDD" and "Uni Corr". Table-3 depicts the statistics of the records of normal and each attack type in "Whole KDD", "10% KDD", "Uni KDD", "Corrected Test" and "Uni Corr" datasets respectively. Each connection record consists of 41 features plus a label as either normal or a specific attack type.

Table 3 Class distribution of datasets

Datasets		Class					Total Attack	Total
		Normal	DoS	Probe	R2L	U2R		
Whole KDD	#Instance	972,780	3,883,370	41,102	1,126	52	3,925,650	4,898,430
	(%)	19.69	79.23	0.83	0.22	0.001	80.31	100
10% KDD	#Instance	97,278	391,458	4,107	1,126	52	396,743	494,021
	(%)	19.69	79.24	0.83	0.23	0.001	80.31	100
Uni KDD	#Instance	87,832	54,572	2,131	999	52	57,754	145,586
	(%)	60.33	37.48	1.46	0.69	0.0035	39.67	100
Corrected Test	#Instance	60,593	229,853	4,166	16,347	70	250,436	311,029
	(%)	19.48	73.9	1.34	5.26	0.02	80.52	100
Uni Corr	#Instance	47,913	23,568	2,682	3,058	70	29,378	77,291
	(%)	61.99	30.49	3.47	3.96	0.09	38.01	100
NSL-KDD	Training	13449	9234	2289	209	11	11743	25192
	Testing	9711	7458	2421	2421	533	12833	22544

2.1 NSL-KKD Dataset

The HyFSA, CFS, CON, IG, and GR are applied on 41 features of "Uni KDD" dataset. The obtained features are 6, 8, 11, 3, and 25 by HyFSA, CFS, CON, IG, and GR respectively. The performance in terms of TPR, FPR, TBM and RMSE of obtained feature subsets at each step of HyFSA are shown in Table-4. The final six features {3, 5, 6, 10, 13, and 29} are obtained by HyFSA, which is 15% of the original feature set. The performance of 6 features assessed by classifiers NB and C4.5 are depicted in Tables 4 and 5 respectively. The HyFSA is also compared with CFS, CON, IG and GR on reduced feature set.

Table 4 Performance of NB on different feature sets

Metrics	Feature Selection Method (# Feature)					
	Full Set (41)	CFS (8)	CON (11)	IG (3)	GR (25)	HyFSA (6)
TPR (%)	97.5	97.4	98.3	99.1	97.6	**99.4**
FPR (%)	3.8	3.8	2.6	1.2	3.7	**0.8**
TBM(sec)	1.44	0.12	0.16	0.08	0.25	0.10
ACC (%)	97.51	97.44	98.26	99.15	97.57	**99.44**
ERR (%)	2.49	2.56	1.74	0.85	2.43	**0.56**
RMSE(%)	15.56	15.72	12.75	9.40	15.50	**6.98**

Table 5 Performance of C4.5 on different feature sets

Metrics	Feature Selection Method (# Feature)					
	Full Set (41)	CFS (8)	CON (11)	IG (3)	GR (25)	HyFSA (6)
TPR (%)	99.8	98.9	99.8	99.8	99.8	**99.9**
FPR (%)	0.2	1.4	0.2	0.2	0.2	**0.2**
TBM (sec)	8.71	1.19	1.37	1.08	3.46	**0.63**
ACC (%)	99.84	98.88	99.83	99.85	99.82	99.86
ERR (%)	0.16	1.12	0.17	0.15	0.18	**0.14**
RMSE (%)	3.82	9.34	3.97	3.52	4.03	**3.48**

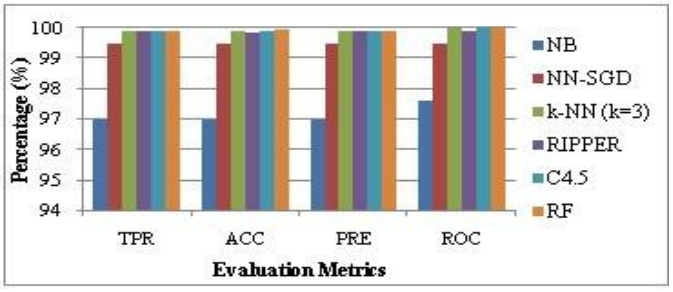

Fig -8: TPR, ACC, PRE and ROC of classifiers

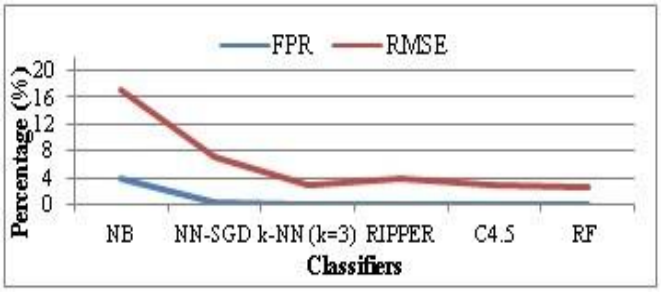

Fig -9: FPR and RMSE of classifiers

The k-NN has highest TTM (5087.73 sec.) among six classifiers, which is extraordi- narily very high. It will increase the computation time and reduce the performance of HEIC. Hence, it is not appropriate as base classifier in ensemble for real-time processing for large volume network traffic. As a result, 5 base classifiers—NB, NN-SGD, RIPPER, C4.5 and RF are selected to build HEIC. Finally, five en- semble models are constructed by using 5 combiners (Average, Product, Majority Voting, Minimum, and Maximum), each employing selected five classifiers. Table 6 illustrates the results of these 5 ensemble models using all 41 features on training dataset "Uni Train" based on different evaluation metrics.

Table 6 Performance of ensembles on training dataset using 41 features

Metrics	Ensemble of Classifiers				
	Average	Product	Majority Voting	Minimum	Maximum
TPR (%)	100.0	99.8	100.0	99.8	98.1
FPR (%)	0.0	0.2	0.0	0.2	2.8
ACC (%)	99.97	99.47	99.97	99.47	98.15
PRE (%)	100.0	99.8	100.0	99.8	98.2
ROC (%)	100.0	99.7	100.0	99.7	100.0
TBM (s)	626.23	590.15	541.61	616.26	599.57
TTM (s)	21.5	17.86	20.34	21.08	18.08
RMSE (%)	4.29	3.91	1.85	3.91	9.17

3. RESULTS WITH DISCUSSION

This section discusses the proposed Module named as HyFSA-HEIC, for intelligent lightweight, accurate, and efficient anomaly based NIDS in detail. The block diagram of Module (HyFSA-HEIC) is represented in Figure 10 It contains following 3 phases:

- Phase 1: Preprocessing of dataset
- Phase 2: Selection of features using HyFSA
- Phase 3: Model development using HEIC

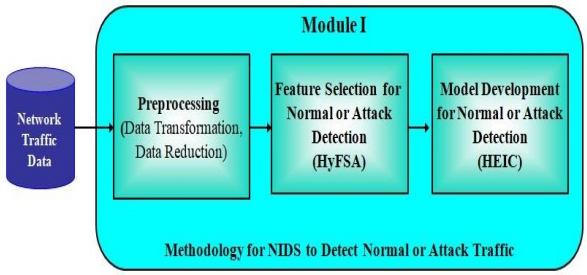

Fig -10: Block diagram of proposed Module (HyFSA-HEIC)

Several experiments have been performed to assess the performance of Module (HyFSA-HEIC) in terms of accuracy and efficiency. All experiments are conducted using Weka. The datasets used in the experiments for training are "Uni Train" and "Red Uni Train", and for testing are "Uni Test" , "Red Uni Test", "Uni Corr" and "Red Uni Corr". Performance metrics are used in the experiments are TPR, FPR, ACC, PRE, ROC, TBM, TTM, and RMSE. The performance of these classifiers using 6 selected features based TPR, FPR, ACC, PRE, ROC, TBM, TTM, and RMSE are evaluated on test dataset "Red Uni Test" is shown in Table 7. and on "Uni Corr" (41 features) and "Red Uni Corr" (6 features) in Table 8. The per- formance of 5 ensembles using 6 selected features on test dataset "Red Uni Test" is illustrated in Table 9 and on "Uni Corr" (41 features).

Table 7 Experimental results of classifiers on training dataset (6 features)

Classifiers	Evaluation Metrics							
	TPR (%)	FPR (%)	ACC (%)	PRE (%)	ROC (%)	TBM (sec)	TTM (sec)	RMSE (%)
NB	95.1	6.1	95.12	95.2	99.2	0.45	1.61	21.97
NN-SGD	97.2	3.7	97.16	97.2	96.7	170.56	1.61	16.86
k-NN(k=3)	99.9	0.1	99.87	99.9	100	0.08	5087.7	3.09
RIPPER	99.8	0.2	99.83	99.8	99.8	46.28	0.21	4.05
C4.5	99.9	0.2	99.88	99.9	100	3.24	0.35	3.32
RF	99.9	0.1	99.9	99.9	100	38.11	8.85	2.85

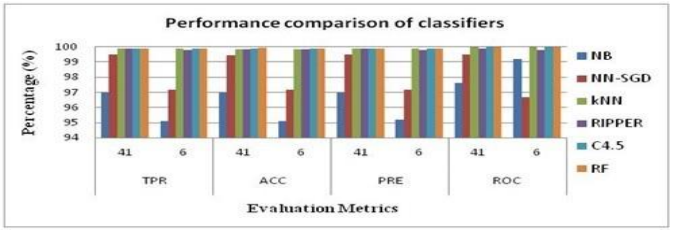

Fig -11: TPR, ACC, PRE & ROC of classifiers (41 & 6 features)

Table 8 Experimental results of ensemble on training dataset (6 features)

Ensemble of Classifier	Evaluation Metrics							
	TPR (%)	FPR (%)	ACC (%)	PRE (%)	ROC (%)	TBM (sec)	TTM (sec)	RMSE (%)
Average	99.9	0.1	99.9	99.9	100	227.6	8.42	7.49
Product	99.6	0.5	97.16	99.6	98.3	227.12	9.91	6.14
Majority Voting	99.9	0.1	99.91	99.9	99.9	226.54	9.09	3.06
Minimum	99.6	0.5	97.16	99.6	98.3	264.01	10.39	6.14
Maximum	97.8	3.2	97.85	97.9	99.9	253.69	10.47	11.59

Table 9 Experimental results of classifiers on test dataset (6 features)

Classifiers	Evaluation Metrics					
	TPR(%)	FPR(%)	ACC (%)	PRE(%)	ROC (%)	RMSE (%)
NB	95.2	6.1	95.17	95.2	99.3	21.87
NN-SGD	97.2	3.6	97.23	97.3	96.8	16.64
k-NN	99.9	0.2	99.84	99.8	100	3.67
RIPPER	99.9	0.2	99.86	99.9	99.9	3.62
C4.5	99.8	0.2	99.85	99.8	99.9	3.63
RF	99.9	0.1	99.92	99.9	100	2.54

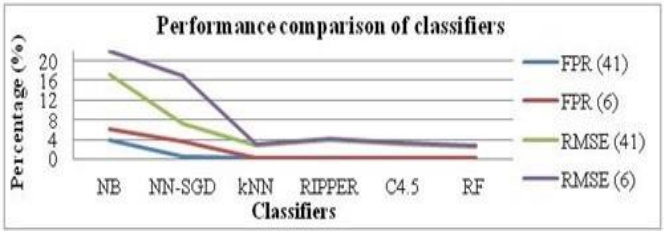

Fig -12: FPR & RMSE of classifiers (41 & 6 features)

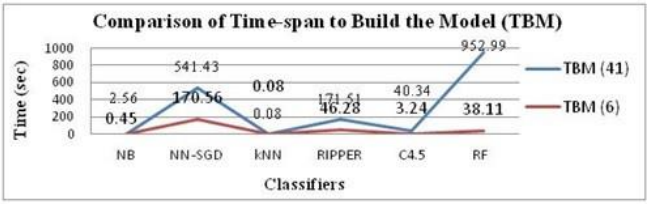

Fig -13: TBM (in sec) of classifiers (41 & 6 features)

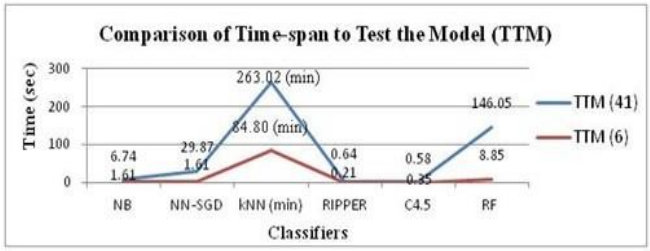

Fig -14: TTM (in sec & for k-NN in min) of classifiers (41 & 6 features)

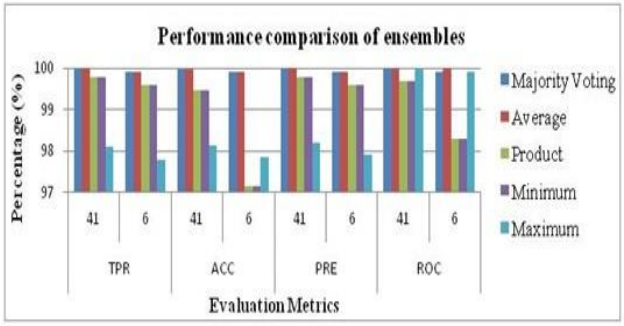

Fig -15: TPR, ACC, PRE & ROC of ensembles (41 & 6 features)

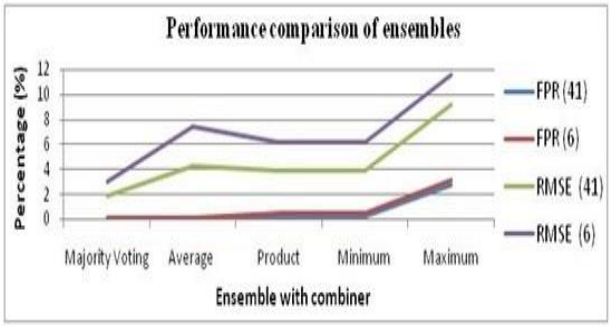

Fig -16: FPR & RMSE of ensemble (41 & 6 features)

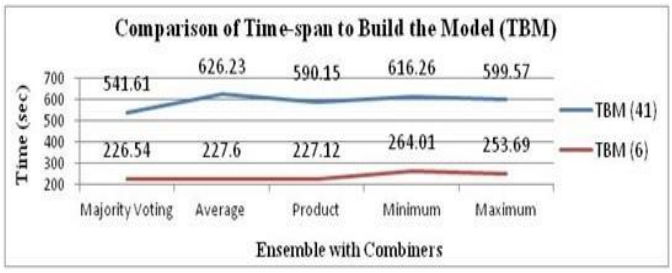

Fig -17: TBM (in sec) of ensembles (41 & 6 features)

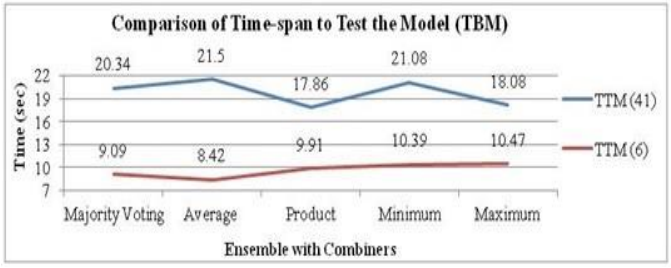

Fig -18: TTM (in sec) of ensembles (41 & 6 features)

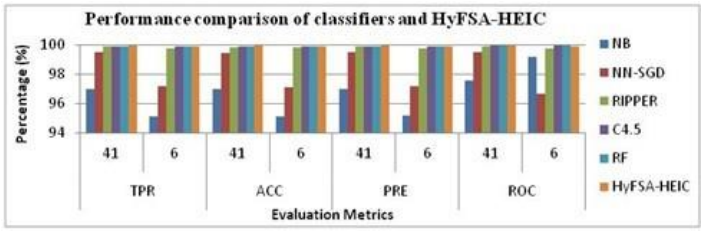

Fig -19: TPR, ACC, PRE, & ROC of classifiers and HyFSA-HEIC

Table 10 Binary class accuracy performances at top 15 features with 4 feature selection models

	ANOVA (%)	CHI-2 (%)	RF (%)	LinearSVM (%)
DT	83.85	90.72	90.47	88.45
NB(G)	83.74	82.78	85.23	85.24
NB(M)	80.32	80.32	87.97	81.33
RF	89.48	90.29	91.95	89.93
LR	87.99	87.86	88.03	88.58
LinearSVM	87.96	87.91	83.64	88.45
SGDClassifier	87.98	87.99	87.92	88.49
Avg results	86.83	87.21	87.87	88.19

Table 11 Multi-class accuracy performances at top 15 features with four feature selection models

	ANOVA (%)	CHI-2 (%)	RF (%)	LinearSVM (%)
DT	81.17	93.36	88.97	93@.50
NB(G)	15.01	10.36	64.24	64.08
NB(M)	80.32	84.66	82.32	83.92
RF	88.81	89.92	97.46	93.51
LR	87.01	85.45	86.26	87.6
LinearSVM	86.25	85.5	85.72	88.93
SGDClassifier	87.01	86.58	87	87.4
Avg results	75.08	76.55	84.57	85.56

Table 12 Comparison of multi-class and binary class accuracy with all and selected features (M = Multi-class, B = Binary-class)

	M(All) (%)	M(Selected) (%)	B(All) (%)	B(Selected) (%)
DT	91.22	93.5	98.68	88.45
NB(G)	80.26	64.08	80.65	85.24
NB(M)	84.89	83.92	85.34	81.33
RF	99.68	93.51	99.67	89.93
LR	92.99	87.61	92.45	88.58
LinearSVM	92.85	88.93	89.61	88.45
SGDClassifier	91.56	87.4	92.26	88.49

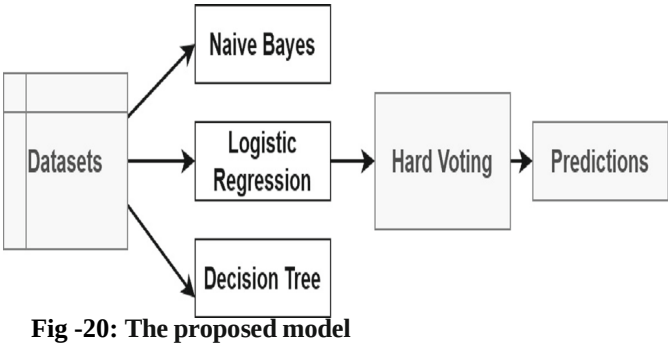

Fig -20: The proposed model

4. CONCLUSION

- Presented next is the proposed HEIC, which aims to address the drawbacks of employing a single classifier to detect intrusions in incoming data and to determine the kind of intrusions in real-time using characteristics selected by HyFSA.
- At conclusion, we provide HySCBA, a hybrid sampling technique that we propose for multi-class unbalanced datasets.
- To address the mismatch between the kinds of attacks in the dataset, this approach combines under-sampling and over-sampling strategies.
- We cover each suggested approach in detail, including the algorithm, experimental setup, results, and analysis.
- Proposes an approach for an intelligent, lightweight, accurate, and efficient anomaly-based NIDS module called HyFSA-HEIC.
- The experimental setup, results, and analysis are detailed in this chapter, along with a block schematic of Module (HyFSA-HEIC).
- Determine whether incoming network communication is malicious or not is the goal of this module.
- In intrusion detection systems, the most common problems are dealing with big datasets with many dimensions, increasing overall ACC while decreasing false alarms, and so on.
- The HyFSA-HEIC Module takes care of these problems by combining the two systems.
- A total of six features—representing only fifteen percent of the initial forty-one features—were used to generate the final conclusions of five classifiers: NB, NN-SGD, RIPPER, DT (C4.5), and RF and Majority Voting. Based on the findings, Module I(HyFSA-HEIC) performed better than other approaches with just 6 characteristics in terms of TPR (99.9%), ACC (99.91%), PRE (99.9%), ROC (99.9%), low FPR (0.1%), and RMSE (3.16%).
- Furthermore, it cut the TTM by 55.30 percent and the TMB by 50.79 percent.
- These six aspects produced better results than the four standard methods: CFS, CON, IG, and GR.
- A smaller feature set, less time to create the model, a higher TP rate, lower FP rate, and fewer mistakes are all positive outcomes.
- Classifiers Naıve Bayes and C4.5 obtained rate scores of 99.4% and 99.9% for TP and 0.8% and 0.2% for FP, respectively.
- By replacing ANN and DL methods with ML algorithms in an ensemble paradigm, it achieves high accuracy while using little resources and producing few false alarms.

4.1 FUTURE SCOPE OF STUDY

- Further improvements in performance, detection accuracy, and reduction of FNR and FPR may be achieved in the future by extending or modifying the current technique with more robust intelligent agents or other intelligent paradiam.
- A new paradigm has evolved, and it's called the Internet of Things (IoT).
- It can help build "smart environments" by connecting real-world items to the web.
- Internet of Things (IoT) settings now need special attention to privacy and security.

REFERENCES

[1]. Amrita, and Ahmed, P. (2013) A Hybrid-Based Feature Selection Approach for IDS. In: Meghanathan, N., Nagamalai, D., and Rajasekaran, S. (eds.) Networks and Communications (NetCom2013). Lecture Notes in Electrical Engineering, Springer, Cham, 284, pp. 195–211.

[2]. Adeel Abbas, Muazzam A. Khan (2021) A New Ensemble-Based Intrusion Detection System for Internet of Things, Arabian Journal for Science and Engineering (2022) 47:1805–1819, https://doi.org/10.1007/s13369-021-06086.

[3]. Amrita, and Ahmed, P. (2012) 'A study of feature selection methods in intrusion detection system: a survey', International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR), 2(3), 1–25.

[4]. Aslahi-Shahri, B.M., Rahmani, R., Chizari, M., Maralani, A., Eslami, M., Golkar, M.J., and Ebrahimi, A. (2015) 'A hybrid method consisting of GA and SVM for intrusion detection system', Neural Computing and Applications, 27(6), 1669– 1676. doi:10.1007/s00521-015-1964-2.

[5]. Balakrishnan, S., Venkatalakshmi, K., and Kannan. A. (2016) 'Intrusion detection system using feature selection and classification technique', International Journal of Computer Science and Application (IJCSA), 3(4), 145–151.

[6]. Amrita, and Ravulakollu, K.K. (2018) 'A Hybrid Intrusion Detection System: Integrating Hybrid Feature Selection Approach with Heterogeneous Ensemble of Intelligent Classifiers', International Journal of Network Security, 20(1), 41–55. doi: 10.6633/IJNS.201801.20(1).06) 41.

[7]. Amrita, and Shri Kant (2019) 'Machine Learning and Feature Selection Approach for Anomaly based Intrusion Detection: A Systematic Novice Approach', International Journal of Innovative Technology and Exploring Engineering (IJITEE), 8(6S), 434–443.

[8]. Aburomman, A.A. and Reaz, M.B.I. (2016a) 'A novel SVM-kNN-PSO ensemble method for intrusion

detection system', Applied Soft Computing, 38, 360–372. doi:10.1016/j.asoc.2015.10.011.

[9]. Aburomman A.A. and Reaz, M.B.I. (2017) 'A novel weighted support vector machines multiclass classifier based on differential evolution for intrusion detection systems', Information Sciences, 414, 225–246.

[10]. Akashdeep, Manzoor, I., and Kumar, N. (2017) 'A feature reduced intrusion detection system using ANN classifier', Expert Systems With Applications, 88, 249–257.

[11]. Bamakan, S.M.H., Wang, H., and Yong, S. (2017) 'Ramp loss K-Support Vector Classification-Regression; a robust and sparse multi-class approach to the intrusion detection problem', Knowledge-Based Systems, 126, 113–126. doi:10.1016/j.knosys.2017.03.012.

[12]. Bhuyan, M.H., Bhattacharyya, D., and Kalita, J. (2016) 'A multi-step outlierbased anomaly detection approach to network-wide traffic', Information Sciences, 348, 243–271.

[13]. Birkinshaw, C., Rouka, E., and Vassilakis, V.G. (2019) 'Implementing an intrusion detection and prevention system using software-defined networking: Defending against port-scanning and denial-of-service attacks', Journal of Network and Computer Applications, 136, 71–85.

[14]. Hadi, A.A.A. (2018) 'Performance Analysis of Big Data Intrusion Detection System over Random Forest Algorithm', International Journal of Applied Engineering Research, 13(2), 1520–1527.

[15]. Han, C., Lv, Y., Yang, D., and Hao, Y. (2011) An Intrusion Detection System Based on Neural Network. In: International Conference on Mechatronic Science, Electric Engineering and Computer (MEC), IEEE, pp. 2018–2021. doi:10.1109/mec.2011.6025886.

[16]. Jokar, P. and Leung, V. (2018) 'Intrusion detection and prevention for zigbeebased home area networks in smart grids', IEEE Transactions on Smart Grid, 9(3), 1800–1811.

[17]. Sultana, N., Chilamkurti, N., Peng, W., and Alhadad, R. (2019) 'Survey on SDN based network intrusion detection system using machine learning approaches', Peer-to-Peer Networking and Application', 12(2), 493–501. https://doi.org/10.1007/s12083-017-0630-0.

[18]. Salo, F., Nassif, A.B., and Essex, A. (2019) 'Dimensionality Reduction with IGPCA and Ensemble Classifier for Network Intrusion Detection', Computer Networks, 148, 164–175. doi:10.1016/j.comnet.2018.11.010.

[19]. Gao, J., Tao, C., Jie, D., and Lu, S. (2019) What is AI Software Testing? and Why. In: IEEE International Conference on Service-Oriented System Engineering (SOSE), San Francisco East Bay, CA, USA. DOI: 10.1109/SOSE.2019.00015.