TTFSE-Two-Tier Feature Selection and ExtractionMachine Learning Model for Effective Network Attack Detection

c, Dr R Rajavignesh[b], and S.Dilipkumar[c]

[a]Assisatnt Professor, Department of Computer science & Engineering, Sastra Deemed University, India.
[b]Professor, Department of CSE, K.S.K College of Engineering & Technology, India
[c]Assistant Professor, Department of CSE, Arasu Engineering College, India.

Article History: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 20 April 2021

Abstract: The Network Infringement Apperception System is vital tool to act against Infringements on computer networks and also protect the network from attacks by detecting and activating Rollback Mechanism to go back to safe state. This paper proposes a Two-Tier Feature Selection and Extraction Machine learning model, based on SelectKBest and Extra Tree Classifier for selecting, extracting and classifying the attack/normal instances in a network. This model encompasses two stages: The paramount tier is responsible for extracting top 40 features across 44 features in order to eliminate the features that have a less impact on detection of network infringement and the extracted features are used as input to succeeding prediction stage, here only 17 features which have high sway on detection of attack. This system uses Label Encoder to change categorical values of the dataset. By measuring its efficiency, several experiments are performed on a public dataset particularly on UNSW_NB15 dataset. The results shows TTFSE ML model has high performance, reduces the training time and is efficient for UNSW-NB15 Dataset.

Keywords: SelectKBest, Extra Tree Classifier, Label Encoder, Two-Tier Feature Selection and Extraction (TTFSE) approach.

1. Introduction

With the sharp headway of information development in the past twenty years, Computer networks are extensively used by industry, business and innumerable fields of the human existence. the fast headway of information development made a couple of challenges to manufacture trustworthy associations which are an inconvenient task. As associations are considered as the main impetus of correspondences, attackers endeavor to enter them to take huge information or upset PC resources. A Network Intrusion Detection System (NIDS) is procedure to get PC resources against poisonous activities (W. Lee,1999). Next-generation intrusion detection expert system and anomaly detection state is presented in (A. Valdes,1995; N. Moustaf,1999). Netstat of intrusion detection is presented in (G. Vigna,1999). Nevertheless, AD sets up a standard profile of activities, and any strong deviations from this profile are excused as an attack. Regardless of the way that, the FAR of AD is high, AD can recognize novel attacks. Hence, numerous examinations have presented its utilization (P. Garcia-Teodoro,2009; A. S. A. Aziz,2014; M.H.Bhuyan,2014). Considerable endeavors are needed to create such named datasets throughout some undefined time frame. These issues make interruption recognition strategies inadequate at recognizing genuine dangers in enormous scope conditions. Also, they can't proficiently learn highlight portrayals to manufacture the effective predictive model. To shrinkage FAR, the erection of NIDS needs to separate and pick the significance highlights of crude organization traffic. Highlight extraction catches ascribes from network parcels. This paper proposes a novel abuse based interruption identification framework to safeguard our organization from five classes, for example, Exploit, DOS, Probe, Generic, and Normal. (SmithaRajagopal,2020).

2. Literary Survey

The variable determination helps in making a precise prescient model in light of the fact that less qualities will in general decrease computational multifaceted nature, accordingly encouraging better execution. AI, a favored way to deal with interruption identification, shows on the proper use of highlights to improve assault recognition rate. Throughout this examination, 31 potential blends of highlights were contemplated and their significance was analyzed. Experimental outcomes relating to include decrease have indicated that an exactness of 97% could be acquired by utilizing just 23 highlights.

The entirety of the past examination works had regarded commitments and simultaneously the past works present that the single separated AI calculation would not propose the acknowledged discovery rate. In this work, the accompanying AI classifiers, for example, Random Forest, Extra Tree, Naïve Bayes, KNN, Decision Tree were actualized, tried and assessed dependent on UNSW-NB15 dataset.

3. Methodology

Method presents the hypothetical idea driving our proposed model. At that point, the proposed TTFSE ML model for network interruption discovery framework is portrayed more in detail. Following this we quickly

present the UNSW-NB15 public dataset to assess the model. Finally, the method of processing the data before explaining the model experiments evaluations with previously proposed approaches.

3.1 Theoretical of Proposed Model

The SelectKBest class just scores the highlights utilizing a capacity (for this situation f_classif however could be others) and afterward "eliminates everything except the k most elevated scoring highlights". So it's sort of a covering, the significant thing here is the capacity you use to score the highlights. What's more, truly, f_classif and chi2 are free of the prescient strategy utilized. Here we chose the f_classif which not yet proposed in any of the paper

It takes as a parameter a score function, which must be applicable to a pair
(XX, yy)
X [:,i] X[:,i] of XX
SKB = SelectKBest ()
SKB.fit_transform (X1 , y).

For this situation SelectKBest utilizes the f score work. This deciphers the estimations of yy as class names and processes. The recipe utilized is actually the one given here: one way ANOVA F-test, with KK the quantity of particular estimations. A huge score proposes that the methods for the KK bunches are not all equivalent. I don't perceive any reason why this should hold by and by, and without this suspicion the FF-values are futile. So utilizing SelectKBest() imprudently may toss out numerous highlights for some unacceptable reasons.

4. Model Description

The TTFSE model has machine learning approach that contains two stages of feature selection: Initial stage is responsible for extracting top 40 of 44 features which having dominance over the prediction. Then the next stage is used select 17 dominant features. The labelled features in form of CSV are first converted to pandas data frame, then the data frame columns contain categorical data are converted using Label Encoder.

The two stages are named as data processing stage, data processing cum classifying stage (algorithm 1, algorithm 2). The algorithm 1 is responsible for selection and extraction 40 features from dataset. The intelligent based system is used to detect attacks from the network having fast detection rate , less computational overhead. . The figure 1 represents the architecture of the Proposed TTFSE system as many trials were held to get the prominent features. the output of algorithm 1 is given as input to the algorithm 2.

Table 1: Variables used in UNSW-NB15

Notation	Description
df,df_test	UNSW-NB15 Test and Train Dataset
x_test,x_train	Test and Train other than labels and attack_cat
y_test,y_train	Test and Train Labels
new_x_test,new_ x_train	New dataset with 40 features
LE	Label Encoder
SKB	SelectKBest
TSKB	Trained model of SKB
ETC	Extra Tree Classifier

TETC	Trained model of ETC
y_pred	Values Predicted by model

Table 2 : Algorithm 1 : Algorithm for formation of New Dataset

Algorithm 1 : Algorithm for formation of New Dataset
Input: df, df_test
Output: x_train, y_train, x_test, y_test
1. **Begin** 2. **df,df_test=LE(df,df_test)** **// Encoding categorical columns in dataset** 3. **x_train,x_test=df.drop[‘attack_cat’,’label’],df_test.drop[‘attack_cat’,’label’]** **//Dropping attack_cat , label column** 4. **y_train, y_test=df[‘label’],df_test[‘label’]** **//retrieving label from dataset** 5. **SelectKBest by k=40** 6. **TSKB=SKB (x_train, y_train)** **//Training for SKB** 7. **fit.transform(x_train)** **// Transforming Training set** 8. **fit.transform(x_test)** **// Transforming Test set** 9. **End**

Table3 : Algorithm 2 : Algorithm for Prediction

ALGORITHM 2: ALGORITHM FOR PREDICTION.
I. INPUT: NEW_X_TRAIN,NEW_X_TEST,Y_TRAIN,Y_TEST II. OUTPUT: Y_PRED

1. BEGIN
2. ETC=EXTRATREECLASSIFIER(MAX_FEATURES=17)

III. //SELECTING 17 FEATURES

3. TETC=ETC.FIT(NEW_X_TRAIN,Y_TRAIN)

IV. //TRAIN THE MODEL

4. Y_PRED =TETC.PREDICT (NEW_X_TEST)

V. //PREDICT THE TEST DATASET

5. END

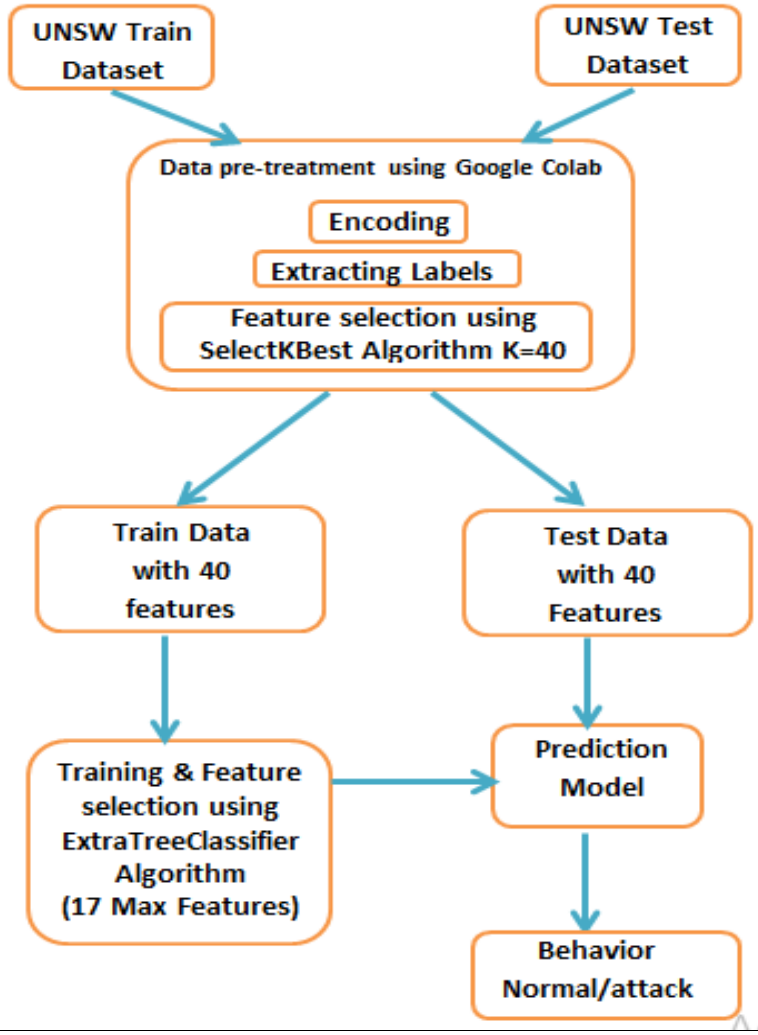

Figure 1: Architecture of proposed TTFSE Machine Learning model Based on SelectKBest &Extra Tree Classifier

5. Results Evaluation

Google Co-Lab having **JupiterNotebook** as base with **12GB of RAM**, **Tesla K80 GPU** through node having Chrome Web surfer Version 80.0.3987.116 having 2GB of RAM..

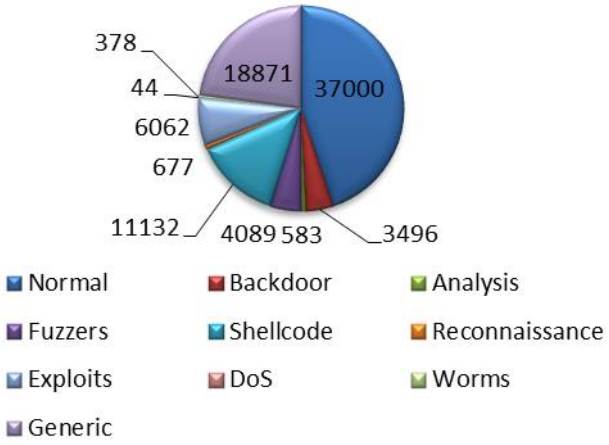

Figure 2: UNSW-NB15 Test Dataset

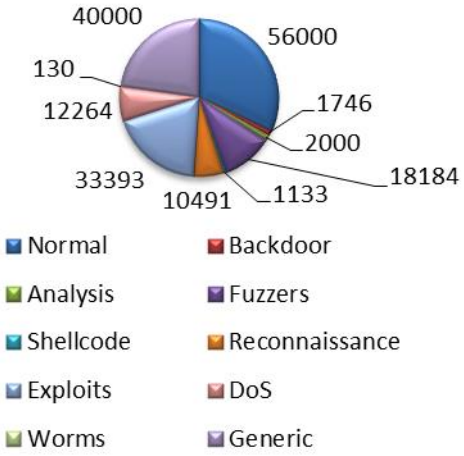

Figure 3 UNSW-NB15 Train Dataset

Table 4 : Features and Classification of Datasets

Feature Selection	**Classifier**	**Accuracy**	**MSE**
SKB-RFE	RFC	75.34	0.2466
SKB-RFE	BNB	73.80	0.2611
SKB-RFE	DTC	66.73	0.3327
SKB	RFC	66.86	0.3314

SKB	BNB	73.89	0.2611
SKB	GNB	67.67	0.3232
SKB	DTC	66.73	0.3327
SKB	KNN	66.73	0.4590
RFE	RFC	67.31	0.3269
RFE	BNB	63.85	0.3165
RFE	DTC	66.73	0.3327
-	RFC	89.18	0.1082
-	BNB	74.89	0.2512
-	GNB	68.21	0.3179
-	DTC	66.73	0.3327
-	KNN	77.41	0.2146
SKB(40)-SKB(17)	ETC	In Range(96 - 99.67)	0.1423-0.0025
SKB	ETC	99.31	0.0

6. Performance Metrics

Following measurement evaluations are determined to get the better outcome for this approach.

Accuracy = (True Pos.+True Neg)/ (T rue Pos.+True Neg.+False Pos. +False Neg)

Precision = True Pos/ (True Pos. +False Pos.)

Recall = True Pos./ (True Pos.+False Neg.)

F −measure = 2× ((Pre×Rec)/ (Pre+Rec))

False Alarm Rate = False Pos/ (False Pos+True Neg)

F-Beta= $((B^2+1)$ Pre. Rec) / $((B^2.$ Pre) +Rec)

Hamming Loss= 1- Accuracy

- (TP): this worth speaks to the right characterization assault bundles as assaults.
- (TN): value speaks to the right arrangement ordinary parcels as typical.
- (FN): value shows that an inaccurately arrangement measure happens. Where the assault bundle named typical parcel, a huge estimation of FN presents a major issue for classification and accessibility of organization assets in light of the fact that the aggressors prevail to go through interruption discovery framework.
- (FP): value speaks to erroneous arrangement choice where the ordinary parcel delegated assault, the expanding of FP esteem builds the calculation time however; then again, it is considered as not exactly destructive of FN esteem expanding.
- Precision: is one of the essential execution markers. The accuracy can be determined by the accompanying condition:

8. Results and Comparisons

TPR (Success pace of recognizing vindictive movement) and FPR are two significant elements are determined. Gathering models are made using the readiness input arrange the testing state as malevolent or affable. In this manner, it is basic to measure the precision of the classifier on future data rather than in the past data. The noticed precision of the classifier on test information is 99.31%. In the accessible UNSW-NB15 dataset furnish us with 44 highlights from which 3 are downright element.

9. Conclusion and Future Work

The approach based on tremendously randomized Trees is presented and discussed to develop an efficient interruption location model. The exploratory outcomes show that the proposed approach can be utilized to build up an Intrusion Detection-Model having high discovery rate, high precision (99.31%) and low False-Positive-Rate. The future work would accumulate constant parcels from the organization and testing them against the effectively ordered preparing dataset.

References

a. Javaid, Q. Niyaz, W. Sun, and M. Alam, ‘‘A deep learning approach for network intrusion detection system,’’ in Proc. 9th EAI Int. Conf. BioInspired Inf. Commun. Technol. (BIONETICS), New York, NY, USA, May 2016, pp. 21–26.

2. L. Buczak and E. Guven, ‘‘A survey of data mining and machine learning methods for cyber security intrusion detection,’’ IEEE Commun. Surveys Tuts., vol. 18, no. 2, pp. 1153–1176, 2nd Quart., 2016.
3. S. A. Aziz, A. T. Azar, A. E. Hassanien, and S. E.-O. Hanafy, “Continuous features discretization for anomaly intrusion detectors generation,” in Soft computing in industrial applications. Springer, 2014, pp. 209–221.
4. Valdes and D. Anderson, “Statistical methods for computer usage anomaly detection using nides (next-generation intrusion detection expertsystem),”Proceedings of theThird International Workship on Rough Sets and Soft Computing, pp. 306–311, 1995.
5. AbdelouahidDerhab , Abdu Gumaei, Amir Hussain And FarrukhAslam Khan,”A Novel Two-Stage Deep Learning Model for Efficient Network Intrusion Detection.
6. Ali MH, Bahaa Abbas Dawood Al Mohammed, Alyani Ismail &MohamadFadliZolkipli. A new intrusion detection system based on Fast learning network and swarm optimization. IEEE, 2018; 6:20255–61. https://doi.org/10.1109/ ACCESS.2018.2820092.
7. Aradhyula, T.V., Bian, D., Reddy, A.B., Jeng, Y.R., Chavali, M., Sadiku, E.R. and Malkapuram, R., 2020. Compounding and the mechanical properties of catla fish scales reinforced-polypropylene composite–from biowaste to biomaterial. *Advanced Composite Materials*, *29*(2), pp.115-128.
8. Arunkarthikeyan K., Balamurugan K. & Rao P.M.V (2020) Studies on cryogenically treated WC-Co insert at different soaking conditions, Materials and Manufacturing Processes, 35:5, 545-555, DOI: 10.1080/10426914.2020.1726945
9. Babu, U.V., Mani, M.N., Krishna, M.R. and Tejaswini, M., 2018. Data Preprocessing for Modelling the audulteration detection in Gasoline with BIS. *Materials Today: Proceedings*, *5*(2), pp.4637-4645.
10. Kavitha, S. Karthikeyan, and P. S. Maybell, ‘‘An ensemble design of intrusion detection system for handling uncertainty using Neutrosophic Logic Classifier,’’ Knowl.-Based Syst., vol. 28, pp. 88–96, Apr. 2012.
11. Khammassi and S. Krichen, ‘‘A GA-LR wrapper approach for feature selection in network intrusion detection,’’ Comput. Secur., vol. 70, pp. 255–277, Sep. 2017.
12. Yin, Y. Zhu, J. Fei, and X. He, ‘‘A deep learning approach for intrusion detection using recurrent neural networks,’’ IEEE Access, vol. 5, pp. 21954–21961, 2017.
13. Chabathula KJ, Jaidhar CD & Ajay Kumara MA. Comparative study of principal component analysis based intrusion detection approach using machine learning Algorithms. 3rd International Conference on Signal Processing, Communication and Networking (ICSCN); 2015. p. 1−6. https://doi.org/10.1109/ICSCN.2015.7219853.
14. Effendy DA, KusriniKusrini&SudarmawanSudarmawan. Classification of intrusion Detection System (IDS) based on computer network. 2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE). IEEE, Yogyakarta: Indonesia; 2017. p. 90−94. https://ieeexplore.ieee.org/document/8285566.
15. Ezhilarasi, T.P., Kumar, N.S., Latchoumi, T.P. and Balayesu, N., 2021. A Secure Data Sharing Using IDSS CP-ABE in Cloud Storage. In Advances in Industrial Automation and Smart Manufacturing (pp. 1073-1085). Springer, Singapore.

16. G Harikrishnan and A Rajaram, "Enhanced Packet covering and Stitching over MAN in the Middle attacks in Wireless Sensor Network," ARPN Journal of Engineering and Applied Sciences, 13(5): 1761-1769, March 2018.
17. G. Vigna and R. A. Kemmerer, "Netstat: A network-based intrusion detection system," in Journal of Computer Security. Citeseer, 1999.
18. http://scikit-learn.org/stable/modules/feature_selection.html
19. http://scikitlearn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html#sklearn.feature_selection.SelectKBest
20. https://colab.research.google.com/
21. https://datascience.stackexchange.com/questions/10773/how-does-selectkbest-workdown
22. https://intellipaat.com/blog/roc-curve-in-machine-learning/
23. https://machinelearningmastery.com/feature-selection-machine-learning-python/
24. https://scikit-learn.org/stable/modules/feature_extraction.html#text-feature-extraction
25. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html
26. https://scikit-learn.org/stable/modules/model_evaluation.html
27. https://scikit-learn.org/stable/modules/neural_networks_supervised.html
28. https://stats.stackexchange.com/questions/253086/selectkbest-feature-selection-python-scikit-learn
29. https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e
30. https://www.dezyre.com/article/top-10-machine-learning-algorithms/202
31. J. A. Khan and N. Jain, ''A survey on intrusion detection systems and classificationtechniques,''Int.J.Sci.Res.Sci.,Eng.Technol.,vol.2,no.5, pp. 202–208, 2016.
32. J. Wu, W. Zeng, and F. Yan, ''Hierarchical Temporal Memory method for time-series-based anomaly detection,'' Neurocomputing, vol. 273, pp. 535–546, Jan. 2018. [35] M. Nawir, A. Amir, N. Yaakob, and O. B. Lynn, ''Multi-classification of UNSW-NB15 dataset for network anomaly detection system,'' J. Theor. Appl. Inf. Technol., vol. 96, no. 15, 2018.
33. J. Wu, Y. Zhang, and W. Lin, ''Good practices for learning to recognize actions using FV and VLAD,'' IEEE Trans. Cybern., vol. 46, no. 12, pp. 2978–2990, Dec. 2016
34. J. Zhang, M. Zulkernine, and A. Haque, ''Random-forests-based network intrusion detection systems,'' IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 38, no. 5, pp. 649–659, Sep. 2008.
35. K. Alrawashdeh and C. Purdy, ''Toward an online anomaly intrusion detection system based on deep learning,'' in Proc. IEEE 15th Int. Conf. Mach. Learn. Appl., Anaheim, CA, USA, Dec. 2016, pp. 195–200.
36. M. Mok, S. Sohn, and Y. Ju, ''Random effects logistic regression model foranomalydetection,''ExpertSyst.Appl.,vol.37,no.10,pp.7162–7166, 2010.
37. M.H.Bhuyan,D.K. Bhattacharyya,andJ.K.Kalita,"Networkanomaly detection: methods, systems and tools," Communications Surveys & Tutorials, IEEE, vol. 16, no. 1, pp. 303–336, 2014.
38. N. Farnaaz and M. A. Jabbar, ''Random forest modeling for network intrusion detection system,'' ProcediaComput. Sci., vol. 89, pp. 213–217, Jan. 2016.
39. N. Moustaf and J. Slay, "Creating novel features to anomaly network detection using darpa-2009 data set," in Proceedings of the 14th European Conference on Cyber Warfare and Security. Academic Conferences Limited, 2015, p. 204.
40. N. Moustafa and J. Slay, ''The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set,'' Inf. Secur. J., Global Perspective, vol. 25, nos. 1–3, pp. 18–31, 2016.
41. N.Shone,T.N.Ngoc,V.D.Phai,andQ.Shi,''Adeeplearningapproachto network intrusion detection,'' IEEE Trans. Emerg. Topics Comput. Intell., vol. 2, no. 1, pp. 41–50, Feb. 2018.
42. P. Garcia-Teodoro, J. Diaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," computers & security, vol. 28, no. 1, pp. 18–28, 2009.
43. P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, ''Stacked denoisingautoencoders: Learning useful representations in a deep network with a local denoising criterion,'' J. Mach. Learn. Res., vol. 11, no. 12, pp. 3371–3408, Dec. 2010
44. P.BaldiandZ.Lu,''Complex-valuedautoencoders,''NeuralNetw.,vol.33, no. 8, pp. 136–147, 2012.
45. R Thanuja and A Umamakeswari. Black hole detection using evolutionary algorithm for IDS/IPS in MANETs,Cluster Computing, 22(2):3131–3143, 2019.
46. S. M. Erfani, S. Rajasegarar, S. Karunasekera, and C. Leckie, ''Highdimensional and large-scale anomaly detection using a linear one-class SVM with deep learning,'' Pattern Recognit., vol. 58, pp. 121–134, Oct. 2016.

47. S. S. S. Sindhu, S. Geetha, and A. Kannan, ‘‘Decision tree based light weight intrusion detection using a wrapper approach,’’ Expert Syst. Appl., vol. 39, no. 1, pp. 129–141, 2012.
48. S.-W. Lin, K.-C. Ying, C.-Y. Lee, and Z.-J. Lee, ‘‘An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection,’’ Appl. Soft Comput., vol. 12, no. 10, pp. 3285–3290, 2012.
49. Sharma, J., Giri, C., Granmo, O. et al. Multi-layer intrusion detection system with ExtraTrees feature selection, extreme learning machine ensemble, and softmax aggregation. EURASIP J. on Info. Security2019, 15 (2019). https://doi.org/10.1186/s13635-019-0098-y
50. SmithaRajagopal, KatiganereSiddaramappaHareesha, PoornimaPandurangaKundapur,”Feature Relevance Analysis and Feature Reduction of UNSW NB-15 Using Neural Networks on MAMLS”Advanced Computing and Intelligent Engineering, 321-332, 2020
51. T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, ‘‘Deep learning approach for network intrusion detection in software defined networking,’’ in Proc. Int. Conf. Wireless Netw. Mobile Commun. (WINCOM), Oct. 2016, pp. 258–263.
52. T. Janarthanan and S. Zargari, ‘‘Feature selection in UNSW-NB15 and KDDCUP’99 datasets,’’ in Proc. IEEE 26th Int. Symp. Ind. Electron. (ISIE), Jun. 2017, pp. 1881–1886.
53. U. Fiore, F. Palmieri, A. Castiglione, and A. De Santis, ‘‘Network anomalydetectionwiththerestrictedBoltzmannmachine,’’Neurocomputing, vol. 122, pp. 13–23, Dec. 2013.
54. W. Lee, S. J. Stolfo, and K. W. Mok, “A data mining framework for building intrusion detection models,” in Proceedings of the 1999 IEEE Symposium, Security and Privacy, 1999, pp. 120–132.
55. Y. Y. Chung and N. Wahid, ‘‘A hybrid network intrusion detection system using simplified swarm optimization (SSO),’’ Appl. Soft Comput., vol. 12, no. 9, pp. 3014–3022, 2012.
56. Yarlagaddaa, J., Malkapuram, R. and Balamurugan, K., 2021. Machining Studies on Various Ply Orientations of Glass Fiber Composite. In Advances in Industrial Automation and Smart Manufacturing (pp. 753-769). Springer, Singapore.
57. Yarlagaddaa, J. and Malkapuram, R., 2020. Influence of carbon nanotubes/graphene nanoparticles on the mechanical and morphological properties of glass woven fabric epoxy composites. *INCAS Bulletin*, *12*(4), pp.209-218.
58. Z. Wang, ‘‘The applications of deep learning on traffic identification,’’ BlackHat, Tech. Ref., 2015.