ICIC Express Letters ICIC International ©2022 ISSN 1881-803X
Volume **16**, Number **1**, January **2022** pp. **59–65**

INTRUSION DETECTION WITH NEIGHBOURHOOD COMPONENT ANALYSIS AND NEURAL NETWORK CLASSIFIERS

Pooja Agarwal* and Rajendra Kumar Srivastava

Department of Computer Science
Dr. Shakuntala Mishra National Rehabilitation University
Mohaan Road Lucknow, Uttar Pradesh 226017, India
*Corresponding author: pa_csphd2015@dsmnru.ac.in; rksrivastava@dsmnru.ac.in

Received April 2021; accepted June 2021

Abstract. *An increase in the wide use of Internet in the present times has also given way to an increase in the online data security threats. The intrusion detection classification model plays a significant role in this context. Earlier research studies have used the standard dataset NSL-KDD based on the binary-classification to improve the accuracy and detection of intrusion detection (ID). The present study aims to improve the accuracy and detection of intrusion detection system on the basis of multi-class class5 classification. Moreover, a comparative study of the binary-class class2 classification and multi-class class5 classification has been embarked upon to find out the level of precision and exactness in the accuracy and detection of intrusion detection model. In the entire process, the NSL-KDD 20% training data set has been used for training and test set. An analysis of the multi-class class5 classification shows an improvement in the accuracy and detection rate of IDS. However, on comparing with class2, the latter showed comparatively better result.*

Keywords: Network security, Intrusion detection, Classification, Neighbourhood component analysis, Neural network, Feature selection

1. **Introduction.** In the present time, almost all the institutions and agencies are providing online services to the users through the use of Internet. In this age of information, huge servers are installed wherein all types of data are stored. The information always attracts the intruders. They are trying incessantly to intrude the central network system. Intrusion refers to a process wherein an intruder tries to intrude or rather intrudes the data servers with a mal-intention to either modify, spoil or steal the relevant data. The system can be intruded by sending the forged packets illegally on the network. Keeping in view the matter of security, continuous experiments are conducted in the present system so that illegal intrusion can be curtailed. Various shortcomings are still present in the system. Intrusion is successful to intrude the computer network by using unprotected system configuration or program defects [1].

Intrusion detection is one such system wherein, the unauthorized packets, file, scripts and viruses sent by the intruders are identified and are destroyed before reaching the network system. This type of mechanism has reduced the number of IT attacks to some extent. However, this is not enough. Continuous research work is being done for the prevention of IT attacks. Intruder is continuously searching new ways to attack due to which a more secure and robust system is needed [2].

The first such model of the intrusion detection system was invented by Denning in 1987 [13]. The basic objective to conduct this system was to develop a mechanism which would find out the intrusion along with its unauthorized access on the computer networks and to take measures to stop it.

DOI: 10.24507/icicel.16.01.59

This model works in different stages in which feature selection and classification criteria are significant [3]. It was originally designed to analyze network traffic discrepancies through various criteria. The key issue in planning to identify the intruders is the selection of position and subset of appropriate features. On this basis, the mechanism to find the intruders can identify a wide range of the forthcoming interruptions.

In this research paper, a novel method of the feature selection is proposed which uses neighbourhood component analysis technique and neural network technique for the classification of the different types of intrusions on the network system [4].

Generally, the available dataset includes redundant and important as well as general and sound features, too. In the preliminary research, customized techniques and feature selection help in drawing attention to find the appropriate features. Feature selection process is a testing process which considers that intrusion detection system has to manage a large measure of information. The significance of this selection can be understood by the results obtained on the basis of this test. The convenience selection method gives surprisingly outstanding results in choosing this feature selection [6,7].

In our study, Section 2 discusses about the background of technology used, Section 3 describes the methodology implemented, Section 4 describes the model formation, Section 5 discusses the experimental outcomes obtained as well as compares them with previous existing results and Section 6 is conclusion of the proposed study.

2. Background.

In this section, background information is provided which is necessary to understand the logic used behind the proposed model.

2.1. Neighbourhood component analysis (NCA).

Neighbourhood component analysis (NCA) is a non-parametric and rooted technique for choosing highlights with the point of target of benefit from estimate precision of grouping calculations [5]. It is a supervised learning method for classifying multivariate data into distinct classes according to a given distance metric over the data.

$$A^* = \arg\max A f(A) \tag{1}$$

It is working to find a linear transformation of input data by learning a distance metric. The average leave-one-out (LOO) classification performance maximizes in the transformed space. The use of this technique is that the number of classes k can be determined as a function of matrix A, up to scale a scalar constant [8].

2.2. Neural network.

A neural network is a network or circuit of neurons, or in a modern sense, an artificial neural network, composed of artificial neurons or nodes. A neural network contains layers of interconnected nodes [9]. The input layer collects input patterns. The output layer has classifications or output signals to which input patterns may map. For instance, the patterns may comprise a list of quantities for technical indicators about a security. Supervised neural networks are programmed and trained to infer a set goal or solution based on sets of inputs and desired outputs. A solution is developed based on the result of a specific set of artificial neurons "firing" and being propagated to the output.

The unit computes some function f of the weighted sum of its inputs. While designing the neural network, the individual element inputs are $x_1, x_2, \dots, x_n$ multiplied by the weights $w_{1j}, w_{2j}, \dots, w_{nj}$ and the weighted values are fed to the summing junction. Their sum is simply w_x, the dot product of the single row matrix w and the vector x. n is the number of elements in the input vector.

$$b + \sum w_{1,n} x_n \tag{2}$$

The weighted sum is $\sum w_{1,n}$, and x_n is called the net input to unit 1, often written as net1. Note that $w_{1,n}$ refers to the weight from unit n to unit 1 and b refers to the bias neuron. The function φ is the unit's activation function. In the simplest case, φ is the sigmoid function, and the unit's output is $\varphi(n) = 1/\left(1+e^{n}\right)$ neural networks learnt by example. The learning rule is provided with a set of examples (the training set) of proper network behavior $fx_1, t_{1g}, fx_2, t_{2g}, \ldots, fx_Q, t_{Qg}$ where x_Q is an input to the network, and t_Q is the corresponding correct (target) output. As the inputs are applied to the network, the network outputs are compared to the targets. The learning rule is then used to adjust the weights and biases of the network in order to move the network outputs closer to the targets [10].

MATLAB has been used to apply neural network tool and get result. In the process Levenberg-Marquardt (LM) algorithm has been applied on the selected NSL-KDD dataset for training and testing. LM algorithm combines the advantages of gradient-descent and Gauss-Newton methods. LM steps are linear combination of gradient-descent and Gauss-Newton steps based on adaptive rules. Gradient-descent dominated steps until the canyon is reached, are followed by Gauss-Newton dominated steps. The training dataset is 79% of the actual data while testing and validation set are 21% each.

3. **Proposed Methodology.** In the proposed method, we evaluated the performance of the intrusion detection system based on NSL-KDD dataset through neighbourhood component analysis and neural network, by utilizing the ability of neural network to implicitly detect complex nonlinear relationships between dependent and independent variables. It has the ability to detect all possible interactions between predictor variables and ability to learn and model non-linear relationships which is useful to solve real-time problems. The steps of the proposed method are as follows:

1) Dataset taken from NSL-KDD repository.
2) Dataset preprocessing.
3) Apply feature selection method on dataset.
4) Apply neural network on selected feature dataset NSL-KDD for training and testing.
5) Evaluate the performance of the proposed model on the basis of accuracy and detection rate.

4. **Model Formation.** In our examinations, we use the proposed procedure on NSL-KDD dataset [11]. In the test NSL-KDD dataset, 20 percent is training dataset of which 79% is managed as data to train the model. We arranged the dataset on the basis of target class. In our procedure, we broke down different information models with various feature sets. To change over them in numeric characteristics, special numeric code had been designated to each conceivable estimation of the given label. NSL-KDD is a development rendition of KDD 99 dataset and unraveled some KDD 99 dataset difficulties [12]. NCA performs feature selection by using the labels of standard dataset NSL-KDD. In the process, neighborhood component analysis starts with choosing features using their load and a relative threshold. After the execution, we get the distinctive class predicated information models as in Table 1.

Here we analyzed 3 different data models on attack type of binary-class class2, multi-class class5 and discovered encouraging results in particular outcomes with binary-class class2.

4.1. **Model performance criteria.** In this paper, we evaluated the experimental result on the basis of standard performance assessment criteria.

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN})$$
$$\text{Detection Rate (DR)} = \text{TP}/(\text{TP} + \text{FN})$$

where true positive (TP) represents the normal instances precisely classified as normal, false positive (FP) denotes the normal instances incorrectly classified as malicious, true negative (TN) represents the anomalous instances precisely classified as malicious, false

Table 1. NSL-KDD data models of selected feature

S.No.	Data model	Total selected features	Selected feature detail
1	NCA 8	9	service, flag, src bytes, dst bytes, count, srv count, dst host count, dst host srv count, port
2	NCA 14	15	protocol type, service, flag, src bytes, dst bytes, count, srv count, serror rate, dst host count, dst host srv count, dst host same srv rate, dst host diff srv rate, dst host serror rate, dst host srv serror rate, port
3	All	42	duration, protocol type, service, flag, src bytes, dst bytes, land, wrong fragment, urgent, hot, num failed logins, logged in, num compromised, root shell, su attempted, num root, num file creations, num shells, num access files, num outbound cmds, is host login, is guest login, count, srv count, serror rate, srv serror rate, rerror rate, srv rerror rate, same srv rate, diff srv rate, srv diff host rate, dst host count, dst host srv count, dst host same srv rate, dst host diff srv rate, dst host same src port rate, dst host srv diff host rate, dst host serror rate, dst host srv serror rate, dst host rerror rate, dst host srv rerror rate, port

Table 2. NSL-KDD data model class detail

S.No.	Class		Detail
1	binary-class	class2	Attack, Normal
2	multi-class	class5	DoS, Probe, R2L, U2R, Normal

negative (FN) denotes the malicious instances incorrectly classified as normal and accuracy (AC) is the proportion of aggregate quantities of TP in addition to add up to quantities of TN separated by add up to number of FP in addition to add up to number of FN.

4.2. **Implementation.** In our experiment, we have used NSL-KDD 20 percent training dataset for feature selection followed by classification technique to acquire the more precise result to utilize and implementation with analysis of results is performed by using MATLAB 2018b. During the process, we observed dataset model with different feature sets, extricated by NCA based feature selection technique. NCA contains the dataset, fitting information, feature weights, and other parameters. In the process, it learns the feature weights using a diagonal adaptation of NCA. In the classification process, we have used Levenberg-Marquardt backpropagation algorithm to train our model. The whole setup is implemented using Intel(R) Core(TM)i7 CPU having 06GB RAM. To validate the execution of the NCA and neural network model, two conclusion metrics are obtained, i.e., accuracy and detection rate.

In our examination, we have utilized NSL-KDD 20 percent training dataset pursued by arrangement procedure to secure the more exact outcome with the help of MATLAB 2018b. During the experiment, we study dataset model on the basis of various features, which was selected by using NCA technique based on fitting data, highlight loads, and different parameters. To perform the appropriate classification, we have utilized Levenberg-Marquardt backpropagation algorithm to prepare our model. The entire experiment is executed on the system having Intel(R) Core(TM)i7 CPU with 06GB RAM. In the study, accuracy and detection rate are used as evaluation metric to test the performance of the NCA and neural network system based model.

5. **Results and Discussion.** The experimental results are evaluated by benchmark measurements like accuracy and detection rate. In our study, we compared the execution of the proposed NCA and neural network model with similar model on the basis of accuracy and detection rate. We have categorized our result of the proposed model of intrusion detection system in following three cases.

Case 1: In case 1, we had taken the NSL-KDD 20 percentage training data and divided it in 79 : 21 ratio used as training and testing respectively. On the same dataset, we had applied NCA and by changed threshold, acquired the 9 features.

Table 3. Experiment results with feature set 9 on class2 and class5

Data model	**Features**	**Accuracy class2**	**Accuracy class5**	**Detection rate class2**	**Detection rate class5**
NCA 8	9	0.9873	0.9461	0.9887	0.9817
NCA 8	9	0.9873	0.9449	0.9876	0.9846
NCA 8	9	0.9852	0.9476	0.9883	0.9836

Case 2: In case 2, we had taken the NSL-KDD 20 percentage training data and divided it in 79 : 21 ratio used as training and testing respectively. On the same dataset, we had applied NCA and by changed threshold, acquired the 15 features.

Table 4. Experiment results with feature set 15 on class2 and class5

Data model	**Features**	**Accuracy class2**	**Accuracy class5**	**Detection rate class2**	**Detection rate class5**
NCA 14	15	0.9967	0.9794	0.9963	0.9890
NCA 14	15	0.9960	0.9782	0.9956	0.9883
NCA 14	15	0.9956	0.9734	0.9938	0.9861

Case 3: In case 3, we had taken the NSL-KDD 20 percentage training data and divided it in 79 : 21 ratio used as training and testing respectively. In the dataset, we had taken all 42 features.

Table 5. Experiment results with feature set 42 on class2 and class5

Data model	**Features**	**Accuracy class2**	**Accuracy class5**	**Detection rate class2**	**Detection rate class5**
All	42	0.9965	0.9877	0.9945	0.9890
All	42	0.9961	0.9838	0.9963	0.9861
All	42	0.9961	0.9883	0.9956	0.9898

The result is obtained from 3 cases as above in which NSL-KDD 20 percent training data divide in 79 : 21 ratio using as training and testing respectively. Now we find the result having 42 features; on this basis we get the accuracy and detection rate with our proposed data. The output in terms of accuracy and detection rate of 3 proposed approaches has been used in developing neural model and its output by taking back-propagation algorithm. The different numbers of features have been used in neural model to view possible success of result in terms of accuracy and detection rate. One can see the performance of our proposed model that can make good accuracy and intrusion detection rate for the proposed classification model. The NSL-KDD 20 percent dataset

Table 6. Comparison of different data models with best accuracy and detection rate on class2 and class5

Data model	Features	Accuracy class2	Accuracy class5	Detection rate class2	Detection rate class5
NCA 8	9	0.9873	0.9476	0.9887	0.9846
NCA 14	15	0.9967	0.9782	0.9963	0.9883
All	42	0.9965	0.9883	0.9945	0.9898

used for training and testing was taken from https://www.unb.ca/cic/datasets/nsl.html. The results of various data models with best features are compared in Table 6.

It is clear from the comparison of various data models by using class2 and class5 with different numbers of features based on three cases of different data models using NCA and neural technique, it shows that NCA 14 data model with class2 is giving better accuracy among various iterations. It is also clear from Table 6 that when we are considering only 15 number of features then it is giving best result among considering 9, 15 and 42 features, so the proposed study shows best finding that NCA 14 with 15 number of features is giving minimum error. Hence, it illustrates that the number of features should be in medium range that is 15 features which is neither low nor high in terms of selecting the features.

Our main motivation of our proposed model has the success rate of accuracy 0.9967 in the case of NCA 14 and detection rate 0.9963 proves the success of model. By choosing the proposed data model, performance of our study could have been further improved.

6. Conclusions.

The motivation of our proposed study for intrusion detection classification model is to study NCA and artificial neural network technique with the use of standard NSL-KDD dataset for training as well as testing the model. We have calculated the accuracy and detection rate on different features of data models using class2 and class5. Our study shows success to get the result in terms of better accuracy and detection rate in class5 classification, too. On comparing different data models on class2 and class5 with different number of features based on 3 cases, it is NCA 14 data model with class2 that is giving better accuracy in various iterations.

Acknowledgment. We are utilizing this chance to offer our thanks to everybody who supported me all through this task. We are grateful for their aspiring guidance, priceless useful feedback, constructive criticism and advice during the study. We are sincerely thankful to them for sharing their honest and lighting up thoughts on various research issues.

REFERENCES

[1] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin and K.-Y. Tung, Intrusion detection system: A comprehensive review, *Journal of Network and Computer Applications*, vol.36, no.1, pp.16-24, 2013.

[2] J. Tang, S. Alelyani and H. Liu, Feature selection for classification: A review, *Data Classification: Algorithms and Applications*, p.37, 2014.

[3] M. S. Pervez and D. M. Farid, Feature selection and intrusion classification in NSL-KDD CUP 99 dataset employing SVMs, *2014 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pp.1-6, 2014.

[4] N. Gao, L. Gao, Q. Gao and H. Wang, An intrusion detection model based on deep belief networks, *2014 2nd International Conference on Advanced Cloud and Big Data*, pp.247-252, 2014.

[5] S. Aljawarneh, M. Aldwairi and M. B. Yassein, Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model, *Journal of Computational Science*, vol.25, pp.152-160, 2018.

[6] Y. Hamid, M. Sugumaran and V. Balasaraswathi, IDS using machine learning-current state of art and future directions, *British Journal of Applied Science & Technology*, vol.15, no.3, 2016.

[7] S. Ganapathy, K. Kulothungan, S. Muthurajkumar, M. Vijayalakshmi, P. Yogesh and Kannan, Intelligent feature selection and classification techniques for intrusion detection in networks: A survey, *EURASIP Journal on Wireless Communications and Networking*, vol.2013, no.1, p.271, 2013.

[8] J. Goldberger, G. E. Hinton, S. T. Roweis and R. R. Salakhutdinov, Neighbourhood components analysis, *Advances in Neural Information Processing Systems*, pp.513-520, 2005.

[9] E. Hodo, X. Bellekens, A. Hamilton, P.-L. Dubouilh, E. Iorkyase, C. Tachtatzis and R. Atkinson, Threat analysis of IoT networks using artificial neural network intrusion detection system, *2016 International Symposium on Networks, Computers and Communications (ISNCC)*, pp.1-6, 2016.

[10] N. Rezazadeh, Initialization of weights in deep belief neural network based on standard deviation of feature values in training data vectors, *International Journal of Scientific Research in Computer Science and Engineering*, vol.5, no.4, 2017.

[11] B. Ingre and A. Yadav, Performance analysis of NSL-KDD dataset using ANN, *2015 International Conference on Signal Processing and Communication Engineering Systems*, pp.92-96, 2015.

[12] P. Aggarwal and S. K. Sharma, Analysis of KDD dataset attributes-class wise for intrusion detection, *Procedia Computer Science*, vol.57, pp.842-851, 2015.

[13] D. E. Denning, An intrusion-detection model, *IEEE Trans. Software Engineering*, vol.SE-13, no.2, pp.222-232, 1987.