sciendo

Transport and Telecommunication, 2022, volume 23, no. 2, 131–141
Transport and Telecommunication Institute, Lomonosova 1, Riga, LV-1019, Latvia
DOI 10.2478/ttj-2022-0011

FEATURE SELECTION METHOD FOR ML/DL CLASSIFICATION OF NETWORK ATTACKS IN DIGITAL FORENSICS

Alexander Grakovski[1], Aleksandr Krivchenkov[2], Boriss Misnevs[3]

Transport and Telecommunication Institute
Riga, Latvia, Lomonosova str. 1
[1]avg@tsi.lv, [2]aak@tsi.lv, [3]bfm@tsi.lv

The research is related to machine learning and deep learning (ML/DL) methods for clustering and classification that are compatible with anomaly detection (network attacks detection) in digital forensics. Research is conducted in the field of selecting subsets of features of a dataset useful for constructing a good predictor (classifier). In this study, a new feature selection method for a classifier based on the Analytical Hierarchy Process (AHP) method is presented and tested. The proposed step-by-step algorithm for the iterative selection of these features makes it possible to obtain the minimum required list of features that are associated with attack events and can be used to detect them. For the classification, Artificial Neural Network (ANN) method is used. The accuracy of attack detection by the proposed method has been verified in numerical experiments.

Keywords: IDS, anomaly detection, machine learning, feature selection, AHP, ANN

1. Introduction

Various security mechanisms have been suggested for detecting cyberattacks (or just attacks) in digital forensics. One of the popular approaches is intrusion detection or prevention systems (IDS/IPS). The IDS/IPS attacks detection methods are commonly categorized as either anomaly-based or signature-based. Signature-based methods search for patterns (signatures) in the analysed network traffic data and generate alarms if the patterns match known attacks behaviour. Anomaly-based schemes estimate the normal behaviour of a system and generate alarms when the deviation from the normal exceeds some threshold.

Most IDS/IPS that are used in practice are signature-based (Faysel and Haque, 2010; SNORT, 2022). However, most of the research conducted in the past years concerns anomaly-based systems (Garcia-Teodoro *et al.*, 2009). In (Ahmed *et al.*, 2016) deep analyses of four important types of anomaly detection methods are presented. They are information theory, clustering, classification, and statistical analysis. We focus our research on machine learning and deep learning (ML/DL) methods like clustering and classification, which are compatible with anomaly detection (Ahmad *et al.*, 2018).

ML methods are categorized into three categories: supervised learning, unsupervised learning, and reinforcement learning (Dasgupta and Nath, 2016). In supervised learning, the problem of classification is solved (Sathya and Abraham, 2013). The training data consists of a set of samples (observations) with which the algorithm is trained. Each sample consists of an input object, typically a vector of variables (that may be called also features), and the desired output value (vector of labels). Classification algorithms supervise the training data and produce a general rule, which can be used for mapping (labelling) new input vectors.

The unsupervised learning approach includes clustering. It finds similarities in the training data. The assumption is often made that the clusters will match reasonably well with an intuitive dividing on groups of samples. Clustering means dividing the available data instances into subgroups based on the similarities between the instances in a certain group.

Data sets for subsequent classification or clustering in the IDS are formed based on the capture of network traffic. This capture of raw packets generates files of considerable size. So, when creating the UNSW-NB15 public available data set, when capturing experimental traffic, 100 GB were accumulated within 31 hours, which were then subjected to appropriate processing to reduce the data dimension (Moustafa and Slay, 2015; Moustafa, 2017). In this step (Figure 1) of data processing the features extraction

and features calculation (features may be based on features) take place. The tables (CSV files) are created with observations (records) in rows and features in columns. The data dimensions and complicity are sufficiently diminished after this step.

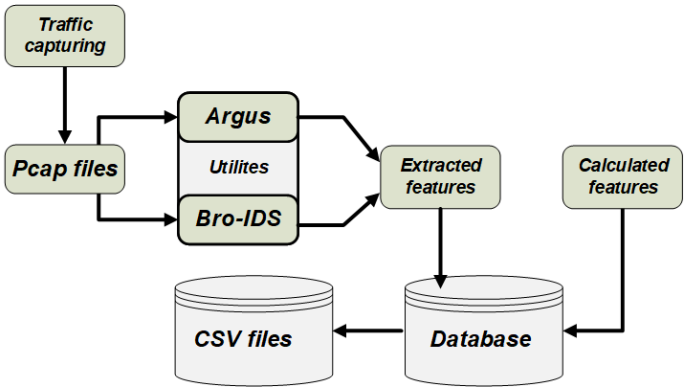


Figure 1. Features creation and data processing for ML methods application in IDS

As it is stated in (Tang *et al.*, 2015), in the past thirty years, the dimensionality of the data involved in ML tasks has increased explosively. Data with extremely high dimensionality has presented serious challenges to existing learning methods. With the presence of many features, a learning model tends to be overfitted. It is also resulting in the degeneration of performance. To address this dimensionality problem, reduction techniques have been studied, which is an important branch of research in the ML area. Feature selection is a widely employed technique for dimensionality reduction. It aims to choose a small subset of the relevant features from the original ones according to certain relevance evaluation criteria, which usually leads to higher learning accuracy for classification or clustering, lower computational cost, and better model interpretability. The place of feature selection in ML classification is presented in Figure 2.

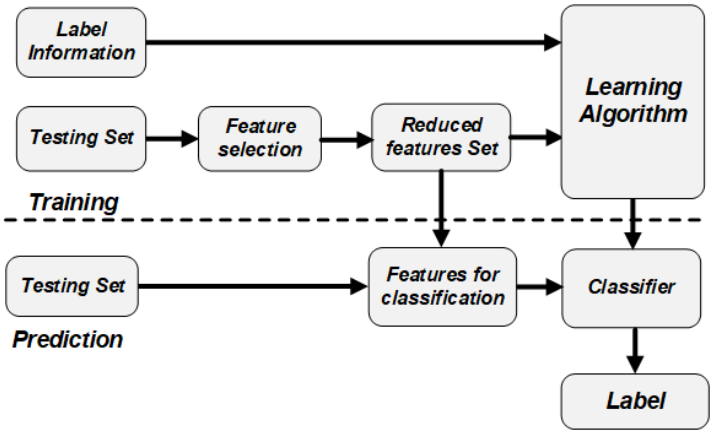


Figure 2. The features selection place in ML classification process

In this research the novel method for feature selection is presented and tested for the classifier based on artificial neural network (ANN) method.

2. Review of features selection and their number reduction

In scientific papers already 20 years ago were fixed that there are many potential benefits of feature selection: facilitating data visualization and data understanding, reducing the measurement and storage requirements, reducing training, defying the curse of dimensionality to improve prediction performance. The papers focus mainly on constructing and selecting subsets of features that are useful to build a good predictor. This contrasts with the problem of finding or ranking all potentially relevant variables. The values of variables just are used for features calculation. Selecting the most relevant variables may exclude many redundant, but relevant, variables. In publicly available data sets, which are used for ML methods efficiency testing, the variables sets are fixed and the reasons for taking them into account are not so clear.

To that end, in (Guyon and Elisseeff, 2003) recommend selecting variables/features in two alternate ways: (1) with a variable/features ranking method using a correlation coefficient with the result of prediction or mutual correlation; (2) with a nested subset some selection method performing forward or backward selection or with multiplicative updates.

Nowadays, many features selection methods are developed, classified and used. Overview of this problem is considered in (Brownlee, 2022; Dwivedi *et al.*, 2020) and the general structure of methods is presented in Figure 3.

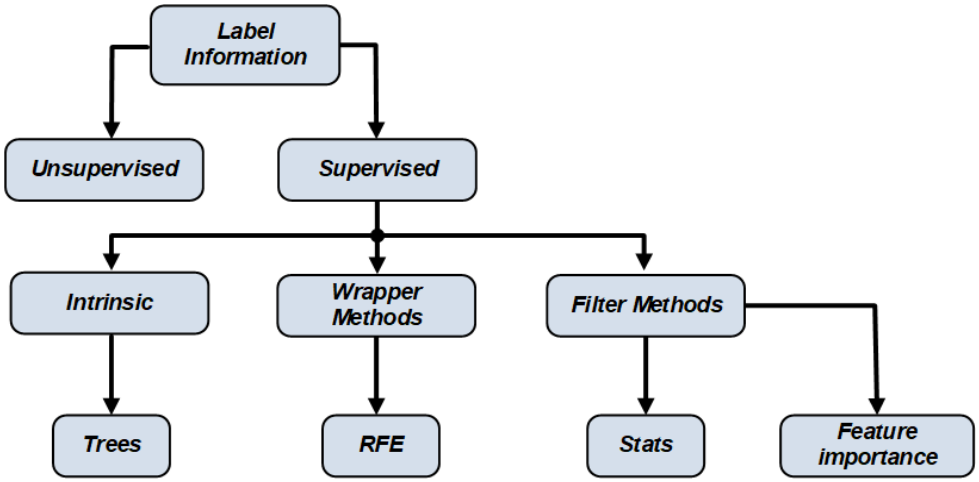


Figure 3. Overview of Feature Selection Techniques

Feature Selection: Select a subset of input features from the dataset.

- Unsupervised: Methods do not use the labels and remove redundant features.
  - Correlation
- Supervised: Use the labels and remove irrelevant variables/features.
  - Wrapper: Search for well-performing subsets of features.
    - RFE
  - Filter: Select subsets of features based on their relationship with the labels.
    - Statistical Methods
    - Feature Importance Methods
  - Intrinsic: Algorithms that perform automatic feature selection during training.
    - Decision Trees

Features reduction is just the reduction of input data into a lower-dimensional feature space. As it is formulated in (Brownlee, 2022) there is no best feature selection method. Just like there is no best set of input variables or best machine learning algorithm (at least not universally). Instead, we must discover what works best for our specific problem using careful systematic experimentation. Such an approach discovers what works best for the specific ML problem. This assumption is confirmed after analysis (and comparison) of experimental data presented in many publications. Such an approach is demonstrated in classification efficiency analysis for the pairs: "features selection method" / "classification or clustering method" (Dwivedi *et al.*, 2020).

One simple and effective method to reduce the number of features consists of applying a Dimensionality Reduction technique in an unsupervised way, i.e., without using the labels themselves. In this case, Dimensionality reduction does not actually select a subset of features but instead produces a new set of features in a lower dimension space. This new set can be used in the classification/clustering process itself. The example (Azevedo, 2022) uses the features on reduced dimensions to do classification. More precisely, it uses the first 2 components of Principal Component Analysis (PCA) as the new set of features.

In our previous work (Krivchenkov *et al.*, 2022) we also have used PCA for feature reduction tasks in the clustering process. In those experiments for clustering, the K-means method was used for the public available NSL-KDD data set (NSL-KDD, 2022). In our experiments, Accuracy for clustering on 2 clusters was increased from 0.81 to 0.90 for decreasing the number of features in the set from 42 features to 8 principal components. Accuracy is one of "well-known" parameters for ML/DL methods efficiency estimation (Fawcett, 2005). Supervised – Filter – Statistical method was used for feature selection

(Krivchenkov *et al.*, 2021b). In this work the supervised - classification methods k-NN (ML method) and ANN (DL method) were applied to NSL–KDD and UNSW-NB15 data sets (Table 1).

Table 1. Classification efficiency for the experiments. Statistical methods for features reduction in all cases are used

Data set	Labels for attacks type	Applied method	Features number	Classification characteristics			
				Accuracy	Precision	TPR	FPR
UNSW-NB15	All	k-NN	42	0,87	0,98	0,83	0,03
UNSW-NB15	DoS	k-NN	42	0,92	0,93	0,99	0,95
UNSW-NB15	Cat	k-NN	42	0,72	-	-	-
UNSW-NB15	All	k-NN	7	0,85	0,95	0,81	0,08
UNSW-NB15	DoS	k-NN	7	0,92	0,93	0,99	0,97
NSL-KDD	All	k-NN	42	0,76	0,64	0,98	0,4
NSL-KDD	DoS	k-NN	42	0,96	0,98	0,97	0,06
NSL-KDD	DoS	ANN	42	0,96	0,98	0,97	0,06
NSL-KDD	Cat	k-NN	42	0,81	-	-	-
NSL-KDD	All	k-NN	6	0,71	0,61	0,89	0,43
NSL-KDD	DoS	k-NN	6	0,96	0,97	0,97	0,08
NSL-KDD	DoS	ANN	6	0,93	0,95	0,96	0,16

The data presented in Table 1 for various data sets confirm the possibility of a significant reduction in the number of features while maintaining (and sometimes increasing) classification efficiency. In this research, the pairs of methods: Stat - k-NN and Stat – ANN was investigated.

Many authors have performed experiments with the goal to find “the best” anomaly-based ML/DL method for attack detection in IDS/IPS (Table 2).

Table 2. The experiments for Features reduction method – Classification method for different public data sets

Reference	Reduction method	Classification method	Testing data sets	Best results achieved	
				Accuracy	False Alarm Rate
[13]	Hybrid evolutionary algorithm GOSA	SVM	NSL-KDD, UNSW –NB15	0.9989	0.009
[19]	Correlation-based-Feature-Selection-Bat-Algorithm (CFS-BA)	A combination of C4.5, RF, and Forest by Penalizing Attributes (Forest PA)	KDD Cup’99, NSL-KDD, CIC-IDS2017	0.998	0.0017 for KDDCup’99
[20][1]	A new kind of single-layer feedforward neural network (SLFN)	Random weight neural network (RWNN)	NSL-KDD, Naive Bayes, SVM	0.995	
[21][2]	Chi-square feature selection algorithm	Random Forest (RF), AdaBoost (ADB), Gaussian Naive Bayes (GNB), K-Nearest Neighbors (KNN), and Quadratic Discriminant Analysis (QDA)	NSL-KDD, UNSW-NB15, AWID	0.961	0.0035
[22][3]	-	Multivariate Correlation Analysis (MCA)	KDD Cup’99	0.999	0.0013
[23][4]	The feature extraction of more correlated variables	Autoencoder (AE) – some equivalent of ANN	NSL-KDD	~0.91	~0.05
[24][5]	Consistency Feature Selection (CBFS), Correlation FS (CFS), Information Gain FS (IG), Relief Filter	Random Forest (RF)	KDD Cup’99, NSL-KDD, UNSW-NB15, CICIDS2017	~0.999	~0.001
[25][6]	Grey Wolf Optimization (GWO), Particle Swarm Optimization (PSO)	NB, ANN	UNSW-NB15	0.877 0.809	0.088 for PSO-GWO-NB 0.114 for PSO-GWO-ANN
[26][7]	PCA	Earth Mover’s Distance (EMD)	KDDCup’99, ISCX2012	0.999 0.901	0.019 for KDDCup’99 0.079 for ISCX2012

[1] - include the overview of often used classification methods;
[2] - in fact it is a project of complete IDS with several network agents;
[3] - in fact using new MCA-based detection system for protection against DoS attacks in IPS;
[4] - the data of efficiency are taken from presented ROC curves;
[5] - is indicated that there is great difference between selected features lists;
[6] - also, this research examines the Most frequently Repeated Features from (PSO and GWO);
[7] - based on computer vision techniques.

3. Reducing the number of data set features by assessing the importance of AHP

One of the possible approaches to reducing the number of necessary features in data sets is to assess the impact of this number on the accuracy of classification based on the analysis of the mutual correlation of features (Krivchenkov *et al.*, 2021). In developing this approach in this paper, we use a pairwise comparison of features by analogy with the well-known AHP (Analytical Hierarchy Process) method (Saaty, 1977).

The Saati method (AHP) has been widely used since the 80s of the last century in various fields to verify the results of a survey of experts, search for optimal solutions, and build a hierarchy of a large number of factors in conditions of uncertainty of functional relationships between them. It is based on the presentation of the results of the survey of experts in a universal discrete scale of assessments $\left[\frac{1}{9}, 9\right]$ (Figure 4) to describe the qualitative characteristics of expert opinions.

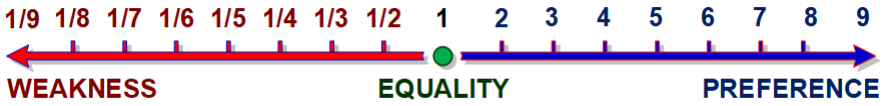


Figure 4. Universal discrete AHP rating scale

Next to quantitative expert estimates of pairwise comparisons of factors shown to the universal discrete scale $a_{ij} = f_i/f_j$, we built a matrix of pairwise comparisons B, string column sums $S = [\sum_{i=1}^{N} a_{i2} \ \sum_{i=1}^{N} a_{i3} \ \dots \ \sum_{i=1}^{N} a_{iN}]$, the matrix C of the weighted pairwise evaluations, and, finally, the vector of weights W, corresponding to the degree of importance of each factor:

$$B = \begin{bmatrix} 1 & a_{12} & a_{13} \dots a_{1N} \\ \frac{1}{a_{12}} & 1 & a_{23} \dots a_{2N} \\ \dots & & 1 \quad \dots \\ \frac{1}{a_{1N}} & \frac{1}{a_{2N}} & \dots \ 1 \end{bmatrix}, \ C = \left[\frac{b_{ij}}{s_j}\right], \ W = \begin{bmatrix} average(c_{1j}) \\ average(c_{2j}) \\ \dots \\ average(c_{Nj}) \end{bmatrix} = \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_N \end{bmatrix}, \tag{1}$$

where for weights w_i the rule $\sum_{i=1}^{N} w_i = 1$ must be fulfilled. This means that for the weighting coefficients w_i can be given the meaning of probabilities and a natural threshold of weighted average probability $w_{eq} = 1/N$ can be introduced. Then the weighting factors of the influence of all factors can be divided into two classes ('*important*' and '*insufficient*') by comparison with the threshold:

$$\begin{cases} w_i \geq w_{eq} - important\ factor, \\ w_i < w_{eq} - insufficient\ factor. \end{cases} \tag{2}$$

The matrix product of a row of sums S by a column of weights W, $\varepsilon = S \times W$ is called the data *conformity index* (for reliable data it must be less than 10% (Saaty, 1977)).

If, instead of expert estimates, comparable numerical values $a_{ij} \in R$ are used to assess the impact, for example, calculated values of the correlation of factors with the event of interest, then a pairwise comparison of every two numbers $f_i \geq 0$ and $f_j \geq 0$ can lead to a universal continuous AHP scale $\left[\frac{1}{9}, 9\right]$ through pre-normalization to a single range $[-1, 1]$ values relative comparison:

$$a_{ij} = A_0 \cdot e^{bt} + C_0, \quad where \ t = \left(\frac{f_i - f_j}{f_i + f_j}\right). \tag{3}$$

Considering the transformation of limit values as the goal of the transformation:

$$t = \{-1, 0, 1\} \rightarrow a = \left\{\frac{1}{9}, 1, 9\right\}, \quad (4)$$

we can easily get the parameters of formula (3): $A_0 = 1,\ C_0 = 0,\ b = \ln(9)$.

Consider as an example the NSL-KDD data (NSL-KDD, 2022) in the task of detecting DoS attacks by the minimum number of features (Zhou *et al.*, 2015, Mohammad, 2021). The basis for reducing the number of features is the data of mutual correlation C_j^i, from which the absolute values of the correlation coefficients of all features with the facts of DoS attacks $\left|C_{DoS}^i\right|$ were selected, where $i = 1, \ldots, N$ is the number of the features (for NSL-KDD dataset $N = 41$).

It is possible to make a pairwise comparison of features by rescaling the paired relative differences into the range $\left[\frac{1}{9}, 9\right]$ in accordance with the scaling principle of the description of the weakness and advantage of Saati (Saaty, 1977). As a result, we obtain a ranking of features by AHP w_i weights as coefficients of the importance of features by the impact on the DoS attack event (see Figure 5). The rule $\sum_{i=1}^{N} w_i = 1$ is performed for a complete set of features, where N is the number of features.

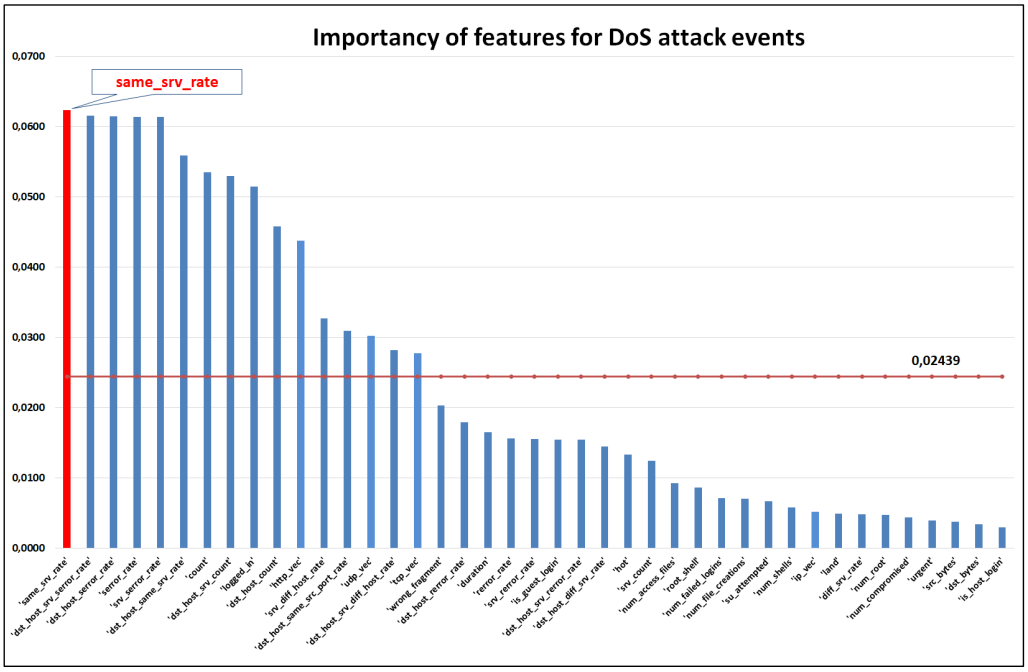


Figure 5. Ranking of 41 features for the NSL-KDD dataset in accordance with the assessment of AHP importance weights calculated by scaling to the Saati scale (Saaty, 1977)

Its comparison with the level of equally probable importance $p_E = {}^1/_N = {}^1/_{41} = 0.02439$ allows us to select a group of 16 most important features from the list (Figure 5) with the leader *'same_srv_rate'* and exclude from consideration features with AHP coefficients less than the threshold value p_E.

The next step of reducing the number of features involves repeating the pairwise comparison (Figure 6) of the cross-correlation coefficients of the 15 remaining features $\left|C_{same_srv_rate}^i\right|$ regarding the most important feature *'same_srv_rate'* to determine the clones associated with it (the most dependent features). Features with AHP weights greater than $p_E = {}^1/_{15} = 0.06667$ are considered dependent on the *'same_srv_rate'* feature and are excluded from further consideration. Of all the remaining ones (Figure 3), the *'dst_host_count'* feature has the largest correlation coefficient with DoS attack events, so it is selected as the next most important feature.

A step-by-step continuation of this operation for all the remaining features allows you to get the minimum necessary list of features that are associated with DoS attack events and can be used to detect them. The two remaining last features have an insufficiently high level of mutual correlation ($C_j^i = -0.77188$), so

both of them should be included in the "short list" of features (Table 3). The result of processing is a reduced number of 4 features (in descending order of importance).

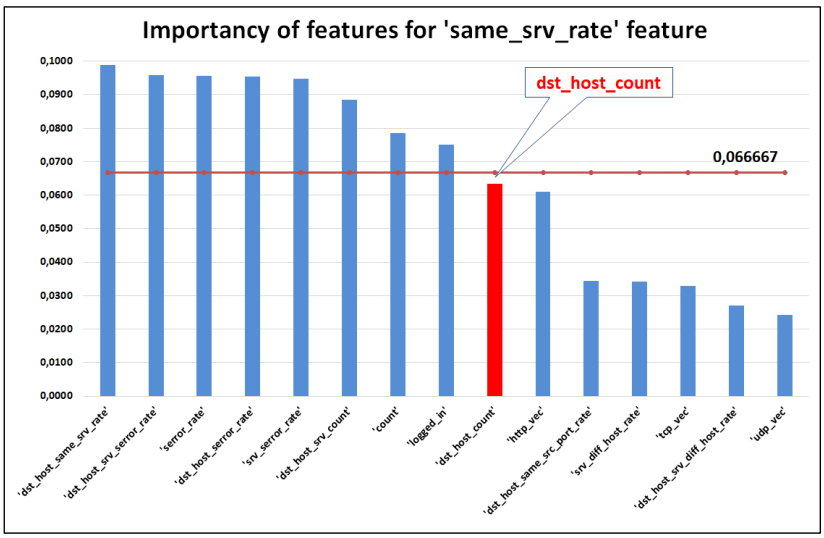


Figure 6. Ranking of the remaining 15 features for the NSL-KDD dataset in accordance with the assessment of AHP importance weights by their relationship to the main feature *'same_srv_rate'*

Table 3. Reduced number of the most important features for detecting DoS attacks in NSL-KDD dataset determined using the pairwise comparison method AHP

No	Features	Correlation to DoS	AHP weight to DoS
1	*'same_srv_rate'*	0,8082	0,0623
2	*'dst_host_count'*	0,4768	0,0458
3	*'udp_vec'*	0,2330	0,0302
4	*'tcp_vec'*	0,2018	0,0277

The described technique of highlighting the most important features allows us to build a hierarchy of features according to the degree of their connection with the events of DoS attacks (Figure 7).

Table 4. Reduced number of the most important features for detecting DoS attacks and 'Normal' status in the NSL-KDD and UNSW-NB15 datasets determined using the AHP pairwise comparison method

	NSL-KDD				UNSW-NB15		
DoS Attack	**No**	**Features**	**Correlation**	**AHP weight**	**Features**	**Correlation**	**AHP weight**
	1	*'same_srv_rate'*	0,8082	0,0623	*'udp_vec'*	0,1507	0,0526
	2	*'dst_host_count'*	0,4768	0,0458	*'sttl'*	0,0990	0,0376
	3	*'udp_vec'*	0,2330	0,0302	*'ct_dst_ltm'*	0,0887	0,0343
	4	*'tcp_vec'*	0,2018	0,0277	*'dur'*	0,0516	0,0213
	NSL-KDD				**UNSW-NB15**		
Status NORMAL	**No**	**Features**	**Correlation**	**AHP weight**	**Features**	**Correlation**	**AHP weight**
	1	*'same_srv_rate'*	0,7519	0,0589	*'sttl'*	0,5042	0,0509
	2	*'srv_rerror_rate'*	0,2535	0,0301	*'ct_dst_sport_ltm'*	0,3937	0,0424
	3	*'dst_host_diff_srv_rate'*	0,2429	0,0294	*'dns_vec'*	0,3639	0,0400
	4	*'udp_vec'*	0,2172	0,0275	*'FIN_vec'*	0,3174	0,0361
	5	*'ip_vec'*	0,2005	0,0263	*'dtcpb'*	0,2829	0,0331

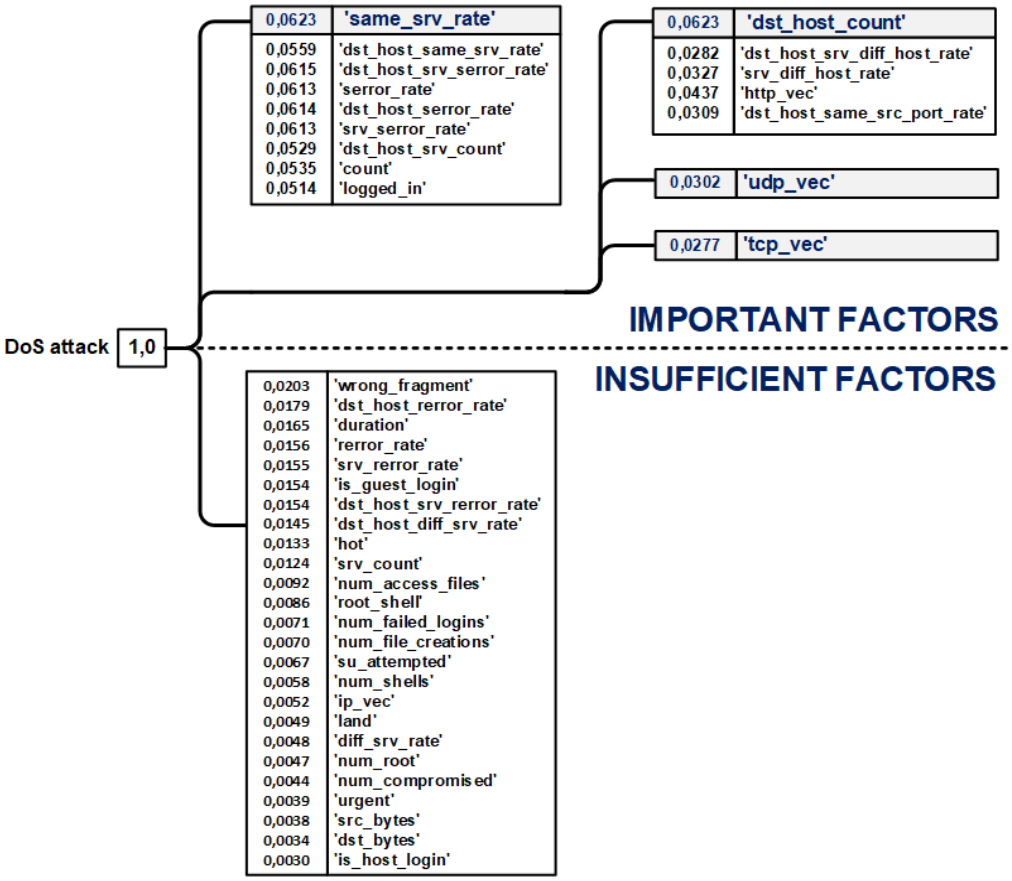


Figure 7. Hierarchy of 41 features for the NSL-KDD dataset according to the AHP importance weights estimate

The same approach can be applied to another UNSW-NB15 dataset (48 features) (Moustafa and Slay, 2015), where the abbreviated list of the most important functions needed to detect DoS attacks includes only 4 features: '*udp_vec*', '*sttl*', '*ct_dst_ltm*', and '*dur*' (Table 4). Here, the number of records corresponding to DoS attacks is no more than 5% in both data sets.

Therefore, to test the AHP hypothesis, the method of constructing hierarchies of features described above was also applied to records with the status '*Normal*' (facts of the absence of any types of attacks), which account for about 45% for both datasets (NSL-KDD and UNSW-NB15). The generalized results of reducing the number of features are shown in Table 4. Here the variety of options for the absence of attacks is much greater, so the minimum number of features for detection has increased to 5.

4. Experimental verification of the method on data sets for solving the classification problem

The next stage of the study was experimental verification of the efficiency of the method proposed by the authors on the already mentioned NSL-KDD and UNSW-NB15 data sets. The validation of the method was carried out using the feature compositions indicated in Table 4, which were obtained using the proposed method.

For numerical experiments, the *Anaconda* environment with *Tensorflow* and *Keras* packages was used. ANN was trained in two hidden layers of 16 neurons per layer. The activation algorithms *'relu'* and *'sigmoid'* (in the output layer) were used. All experiments were performed for fixed values of *epochs* = 10, *batch_size* = 512, and other neural network training parameters. The classification problem was solved to recognize the state of a DoS attack or its absence – the '*Normal*' state.

The following characteristics were used as parameters for validation (Krivchenkov *et al.*, 2021a):

- Accuracy;
- Normalized Confusion Matrix (TP, FP, TN, FN);
- ROC curve.

As a criterion for validating the method, we used not some maximum achieved value of a parameter, for example, Accuracy, but the relative difference in the indicators that were obtained for all specified parameters of the experiment under fixed conditions for conducting it (data sets, ANN parameters, learning algorithm, etc.) for two cases - a complete set of features and a reduced set of features defined using the proposed method.

An analysis of the discrepancy in the results should have indicated the potential use of a new method to reduce the number of features for training a neural network to recognize an attack.

Table 5. Numerical results of experimental method validation

Data Set	Event Class	Number of features	Accuracy	True Positive (TP)	False Positive (FP)	True Negative (TN)	False Negative (FN)	ROC curve
NLS-KDD	DoS	42	0.95	0.71	0.04	0.01	0.24	0.986
		4	0.92	0.69	0.05	0.02	0.23	0.956
	Normal	42	0.76	0.42	0.01	0.23	0.34	0.833
		5	0.74	0.42	0.01	0.24	0.33	0.881
UNSW-NB15	DoS	48	0.93	0.93	0.00	0.07	0.00	0.911
		4	0.93	0.93	0.00	0.07	0.00	0.873
	Normal	48	0.89	0.31	0.01	0.10	0.58	0.982
		5	0.85	0.28	0.04	0.11	0.57	0.943

Analysis of the results of the method application validation for the NLS-KDD set:

- For the class of DoS attacks Accuracy for 4 features and for 42 features is almost the same (0.92 and 0.95) – the difference is 3%. ROC area is also only slightly worse (0.956 vs. 0.986) – the difference is 3%.
- Accuracy for DoS (both experiments #1 and #2) was achieved higher than for Normal (both experiments #3 and #4) - 0.92 and 0.95 versus 0.74 and 0.76.
- Accuracy for 5 features and for 42 features Normal is almost the same (0.74 and 0.76) – the difference is 2%.
- ROC area for 42 Normal features is even slightly worse than for 5 features (0.881 vs 0.833) – the difference is 5%.

Analysis of the results of the method application validation for the UNSW-NB15 kit:

- Recognition accuracy for 5 features for the Normal class (no attacks) Accuracy = 0.85 – almost the same as for the full set (48 features – 0.88) – the difference is 3.5%.
- The obtained recognition values True Positive = 0.28 (no attacks) and True Negative (there is an attack) = 0.57 are about the same – 0.31 and 0.57 – the difference is 10% and 0%.
- The obtained values of non-recognition (errors) False Positive = 0.1 (no attacks) and True Negative (there is an attack) = 0.008 are almost the same as for the full set (48 features – 0.11 and 0.04) – the difference is 10% and 20 %.
- Area under ROC curve 0.943 vs 0.982 for full set – the difference is 4%.

5. Conclusions

An AHP-like method is proposed for building a hierarchy and highlighting the most important characteristics of the network traffic state for attack detection. As shown by numerical experiments, the proposed method can be successfully used to select and reduce features for two proven popular datasets NLS-KDD and UNSW-NB15 without a noticeable loss in recognition accuracy. On average, a decrease in the number of features from 42 or 48 to 4 or 5, that is, by a factor of 10, leads to a decrease in recognition quality indicators by an average of less than 10%.

With the help of a formally (algorithmically) determined threshold of feature importance as a level of equiprobability, the proposed method makes it possible to obtain an objective classification. This classification is based on measured data, not subjective expert opinion, and can be determined automatically. An interesting continuation of this study in the future may be the use of the found hierarchical groups of features to form some calculated integral features of the invasion. Of particular interest is the continuation of research in the field of comparing the characteristics of the accuracy of detecting attacks with the results of the considered ML/ML methods with a full and reduced set of features at the input of the analysis.

References

1. Ahmed, M., Mahmood, A.N. and Hu, J. (2016) A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60, 19–31.
2. Ahmad, B., Jian, W. and Ali, Z.A. (2018) Role of Machine Learning and Data Mining in Internet Security: Standing State with Future Directions. *Journal of Computer Networks and Communications,* Volume 2018, Article ID 6383145, Open access. DOI: 10.1155/2018/6383145.
3. Azevedo, G. (2022) Feature selection techniques for classification and Python tips for their application. In: *Towards Data Science WEB site*, https://towardsdatascience.com/feature-selection-techniques-for-classification-and-python-tips-for-their-application-10c0ddd7918b, [Accessed 04/02/2022].
4. Binbusayyis, A., Vaiyapuri, T. (2019) Identifying and Benchmarking Key Features for Cyber Intrusion Detection: An Ensemble Approach. In: *IEEE Access*, July 2019, DOI: 10.1109/ACCESS.2019.2929487.
5. Brownlee, J. (2022) How to Choose a Feature Selection Method For Machine Learning. In: *Machine Learning Mastery WEB site*, https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/, [Accessed: 04/02/2022].
6. Dasgupta, A. and Nath, A. (2016) Classification of Machine Learning Algorithms. *International Journal of Innovative Research in Advanced Engineering (IJIRAE)* ISSN: 2349-2763, 3(03).
7. Dwivedi, S., Vardhan1, M., Tripathi, S. (2020) Incorporating evolutionary computation for securing wireless network against cyberthreats. *The Journal of Supercomputing*. Published online 20 Jan 2020. DOI:10.1007/s11227-020-03161-w
8. Garcia-Teodoro, P., Diaz-Verdejo, J., Macia-Fernandez, G. and Vazquez, E. (2009) Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers and Security*, 28(1–2), 18–28.
9. Guyon, I., Elisseeff, A. (2003) An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3, 1157-1182.
10. Fawcett, T. (2005) An introduction to ROC analysis. *Pattern Recognition Letters,* 27(2006), 861–874. DOI:10.1016/j.patrec.2005.10.010. Available on line www.elsevier.com/locate/patrec
11. Faysel, M.A. and Haque, S. S. (2010) Towards Cyber Defense: Research in Intrusion Detection and Intrusion Prevention Systems. *Journal of Computer Science*, 10(7), 316–325.
12. Ieracitano, C., Adeel, A., Morabito, F., Hussain, A. (2019) A Novel Statistical Analysis and Autoencoder Driven Intelligent Intrusion Detection Approach. *Neurocomputing*. https://www.sciencedirect.com/science/article/pii/S0925231219315759. DOI: 10.1016/j.neucom.2019.11.016.
13. Kamalakanta, E., Rupesh, S., Kumar, R., Padmalochan, Y., Madhav, V. (2020) A context-aware robust intrusion detection system: a reinforcement learning-based approach. *International Journal of Information Security*. DOI: 10.1007/s10207-019-00482-7
14. Krivchenkov, A., Misnevs, B. and Grakovski, A. (2021a) Using Machine Learning for DoS Attacks Diagnostics. In: *Reliability and Statistics in Transportation and Communication. RelStat 2020. Lecture Notes in Networks and Systems*, 45–53. Springer.
15. Krivchenkov, A., Misnevs, B., Grakovski, A. (2021b) Experimental Comparison of ML/DL Approaches for Cyberattacks Diagnostics. In: *Zamojski W. et al. DepCoS-RELCOMEX 2021*, AISC, Springer, 1389, 213-223.
16. Krivchenkov, A., Misnevs, B. and Grakovski, A. (2022) Structural Analysis of the NSL-KDD Data Sets for Solving the Problem of Attacks Detection Using ML/DL Methods. In book: *Reliability and Statistics in Transportation and Communication, RelStat 2021*, 3-13. Springer.
17. Mohammad, A. (2021) Intrusion Detection Using a New Hybrid Feature Selection Model. *Intelligent Automation & Soft Computing*, 30(1). DOI:10.32604/iasc.2021.016140.
18. Moustafa, N., Slay, J. (2015) UNSW-NB15: A Comprehensive Data set for Network Intrusion Detection systems. In: *Military Communications and Information Systems Conference (MilCIS)*. Open access, https://www.researchgate.net/publication/287330529
19. Moustafa, N. (2017) Designing an online and reliable statistical anomaly detection framework for dealing with large high-speed network traffic. Thesis for: PhD degree. Open access, https://www.researchgate.net/publication/328784548
20. NSL-KDD. (2022) NetworkSecurity, InformationSecurity, CyberSecurity WEB site, https://www.kaggle.com/hassan06/nslkdd, [Accessed 2022/02/04].
21. Raza, R., Ashfaq, He, Y., Chen, D. (2016) Toward an efficient fuzziness based instance selection methodology for intrusion detection system. Springer, published online 2016. DOI: 10.1007/s13042-016-0557-4.

22. Saaty, T. L. (1977) A scaling method for priorities in hierarchical structures. *Journal of Mathematical Psychology,* 15(3), 234–281.
23. Sathya, R. and Abraham, A. (2013) Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. *International Journal of Advanced Research in Artificial Intelligence (IJARAI)*, 2(2).
24. SNORT. Source: project "Snort" [Online]. https://www.snort.org/, [Accessed: 27/01/2022].
25. Tan, Z., He, A., Nanda, P., Liu, R. (2014) A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis. *In IEEE Transactions on Parallel and Distributed Systems*, February 2014. DOI: 10.1109/TPDS.2013.146
26. Tan, Z., Jamdagni, A., Hez, X., Nanda, P., Liu, R., Hu, J. (2015) Detection of Denial-of-Service Attacks Based on Computer Vision Techniques. In*: IEEE Transactions on Computers*, May 2015. DOI: 10.1109/TC.2014.2375218, https://www.researchgate.net/publication/268225728
27. Tang, J., Alelyani, S. and Liu, H. (2015) Feature Selection for Classification: A Review. Published in: *Data Classification: Algorithms and Applications*. Open access, https://www.semanticscholar.org/paper/Feature-Selection-for-Classification%3A-A-Review-Tang-Alelyani/310ea531640728702fce6c743c1dd680a23d2ef4?p2df
28. Zhou, Y., Cheng, G., Jiang, S. and Dai, M. (2015) An Efficient Intrusion Detection System Based on Feature Selection and Ense mble Classifier. *Journal of LATEX class files*, 14(8).