Sādhanā (2020)45:2
https://doi.org/10.1007/s12046-019-1230-x

Intrusion detection system using an optimized kernel extreme learning machine and efficient features

JAMAL GHASEMI[1,*], JAMAL ESMAILY[2] and REZA MORADINEZHAD[3]

[1]Faculty of Engineering and Technology, University of Mazandaran, Babolsar, Iran
[2]Shahid Rajaee Teacher Training University, Tehran, Iran
[3]Drexel University, Philadelphia, PA, USA
e-mail: j.ghasemi@umz.ac.ir; esmaily.jamal@gmail.com

MS received 16 August 2018; revised 21 July 2019; accepted 2 October 2019

Abstract. In the study of Intrusion Detection System (IDS) choosing proper combination of features is of great importance. Many researchers seek to obtain appropriate features with optimization algorithms. There are several optimization algorithms that can properly select a near-optimal combination of features to reach an improved IDS. Genetic Algorithms (GA) as one of the most powerful methods have been used in this research for feature selection. In this paper, voted outputs of built models on the GA suggested features of a more recent version of KDD CUP 99 dataset, NSL KDD, based on five different labels, have been gathered as a new dataset. Kernel Extreme Learning Machine (KELM), whose parameters have been optimally set by GA, is executed on the obtained dataset and results are collected. Based on IDS criteria, our proposed method can easily outperform general classification algorithms which use all the features of the employed dataset, especially in R2L and U2R with the accuracy of 98.73% and 98.22% respectively which is the highest among the current literature.

Keywords. Intrusion detection system (IDS); genetic algorithms (GA); feature selection; kernel extreme learning machine (KELM).

1. Introduction

Computer networks play an important role in the modern days. Similar to technical advances of computer networks, intruders also change their methods to be capable of reaching to sensitive information. Benefiting from Machine Learning methods, Intrusion Detection Systems (IDS) are able to diagnose suspicious traffic and inform the firewalls. Various specific categories of IDSs utilize machine learning and data mining concepts. They attempt to learn from labeled or unlabeled records of network traffic to learn the behavior of the normal and attack traffic. There are two styles of learning in the IDS: online learning and offline learning. In the online learning the IDS is made to learn from online traffic in real time and classify the traffic records [1]. However, an offline learning IDS is made based on a standard and studied datasets [2]. One of the most studied and known datasets in this scope is KDD CUP 99 [3]. This bulky dataset contains more than five million records and 41 features. A newer and more reliable version of popular KDD CUP 99 is NSL-KDD [4, 5] which is used in this work.

In the case of attacking on new and dangerous threats, IDS models should be built based on datasets which contain new forms of attacks. Working on complex and advanced methods to build an IDS based on outdated dataset cannot promise trustable and applicable systems in modern network environments. That is why researchers tend to use more recent and applicable datasets such as NSL-KDD [5–8] which from now on we refer to as NSL. New datasets like NSL also contain the traffic data which are able to monitor the behavior of modern network topologies which are undeterminable in old datasets. Hence advanced network topologies and more importantly fresh attack types make using new datasets inevitable. For these reasons, NSL has been considered as the main benchmark of this work.

Generally, in pattern recognition, that might seem, accounting more feature in the final model, will conclude better outcomes. This means expanding the feature vector dimension will apparently enhance the detection ability. However, the curse of dimensionality suggests containing more features in the final model could not inevitably reach an improved performance [9, 10]. For this reason, researchers focus on selecting proper features to contact to an applicable IDS. NSL contains numerous features. Hence in the case of poor feature selection, curse of dimensionality could damage the system performance. In this paper, the faced problem in IDSs discussions has been considered.

GA is one of the most powerful and widely utilized optimization algorithms [11]. Using classification algorithm alongside evolutionary algorithms such as GA [12] proven

*For correspondence
Published online: 20 December 2019

to be promising in the recent years. When the size of problem space becomes much larger, methods like GA are more helpful [13, 14]. Optimization algorithms are a useful tool to extract features [15]; these algorithms work on various combinations of features to obtain their desired results. In this paper, GA has been employed to extract features based on different labels in NSL dataset.

Among the classification algorithms, DT [16, 17], MLP [18, 19], KNN [20, 21] and SVM [22, 23] have provided powerful results on network traffic data. These four algorithms have been selected for primary classification purposes. These algorithms alongside of feature selection methods and optimization algorithms like GA, shape fundamentals of suggested IDSs in many previous work. In [24] authors use a hybrid KPCA-SVM-GA method for intrusion detection. They use KPCA for feature reduction, SVM for classification and GA for punishment factor of the parameter of C in kernel function of SVM. Simulations on KDD CUP 99 demonstrate the efficiency of their system. In another study [25] the authors used Fuzzy rule-based system which can act as a genetic feature selection for finding optimal feature combination. Their simulations on KDD CUP 99 dataset show efficient detection rate for attack traffic and low false alarm rate for normal records. In [26] with an innovative idea, the authors present a biologically inspired computational approach learn signatures for network traffic using a supervised learning classifier system. Their main approach is to minimize the overlaps and conflicts between signatures by new generalization operator. They utilized KDD CUP 99 dataset as their evaluation benchmark as suggest the effectiveness of their model. In [15] cuttlefish algorithm (CFA), an optimization algorithm, is used as a search strategy to extract optimal subset of features and the decision tree (DT) classifier for the classification phase. KDD CUP 99 has been considered for built model evaluations. In comparison with all features' simulations, the obtained features show better performance. Four mentioned papers employed various artificial intelligence concepts to build an improved IDS. However, since they employed outdated KDD CUP 99 as their major benchmark, their proposed IDSs performance on diagnosing fresh and modern traffic is not yet determinable. In [27] the concept of multi-objective approaches for feature selection and its application in Growing Hierarchical Self-Organizing Maps (GHSOMs) has been proposed. With multi-objective approach they are able to distinguish between attack and normal, and distinguish different types of attacks. Authors use NSL as their benchmark dataset. In [28] with combination of statistical concepts and SOM algorithm, authors are able to reach an efficient model on NSL dataset. They used Principle Component Analysis (PCA) and Fisher Discriminant Ratio (FDR) for feature selection and noise removal phase respectively. For diagnosing traffic types, the SOM algorithm has been applied. In [7] the researchers used Online Sequential Extreme Learning Machine (OS-ELM) for building a time efficient model based on ensemble of Filtered, Correlation and Consistency based feature selection techniques. Their IDS evaluation based on KDD CUP 99 and NSL suggests their proposed method can be considered as an applicable and efficient system. In their classification phase, recent discussed studies exploited single classification algorithms. As a better approach, our proposed method gathers all of four studied classifiers' opinions on a single record and sensibly attempts to distinguish between attack and normal ones. In [29] authors use an outlier based system based on identifying relevant subset of features by mutual information and generalized entropy-based feature selection algorithms. A tree-based clustering technique also has been used for ranking outlier and finding anomalies. They used NSL dataset for their complex method evaluations. In [30] authors use the adapted chaos concept in their proposed time-varying chaos particle swarm optimization (TVCPSO) method to do parameter setting and feature selection for multiple criteria linear programming (MCLP) and support vector machine (SVM). NSL as a more reliable version of KDD CUP 99 has been chosen for their evaluations. These two recent papers have been compared with our proposed method and as it has been illustrated further, the proposed method is able to outperform these two IDSs most notifying R2L (Remot-to-Local) and U2R (User-to-Root) detections, which could be hardly detected due to the dataset server imbalance for these two classes.

Optimization algorithms have been applied as a feature extraction tool and in some cases for parameter setting. In this paper, both applications of these algorithms have been employed. In most cases, there are only one or two types of classification algorithms. In this paper, there are four classification algorithms whose information can be applied to the final model. A combination of best predictions of built models on GA-suggested features will be gathered in the new dataset – this dataset will be referred to as GA-dataset. In other words, GA-dataset contains the predictions of the combination of all classifiers based on different labels. Kernel Extreme Machine Learning (KEML) [31] optimized by GA would be executed on GA-Dataset and results will be obtained. Outcomes suggest the proposed method is more capable of detecting the attack and normal records – especially R2L and U2R attacks – rather than other four well-known algorithms which work on the entire features.

The contents of this paper will be as follows. In the following section, the proposed method will be discussed. In the next section, the results of our method and the performance of the proposed method versus other methods will be provided. Finally, the conclusion of this study will be conveyed.

2. Proposed method

The main idea of the proposed method is to utilize different classifiers' outcomes based on different labels. Every model tries to classify the specific label in the NSL dataset. Having gathered all of this information in GA-dataset,

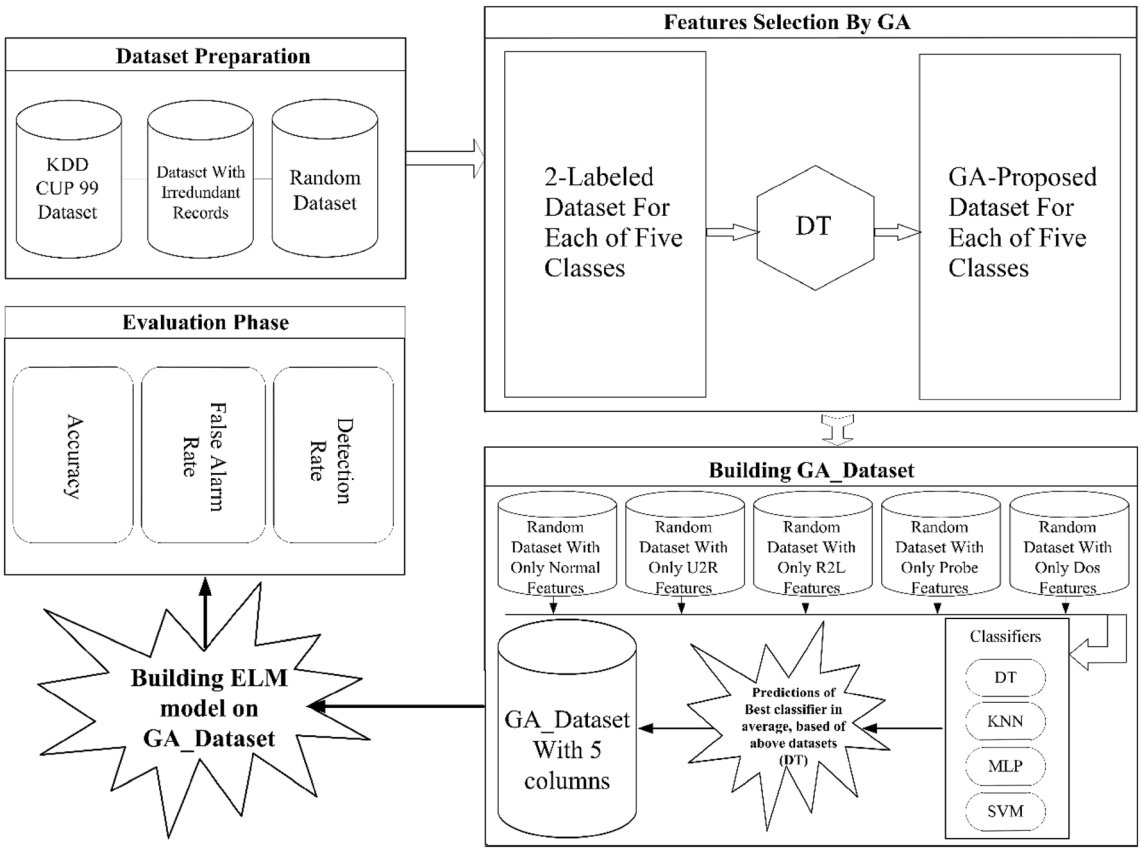

Figure 1. Block diagram of the proposed method.

KELM is able to build a superior IDS. Figure 1 shows the diagram of the proposed method.

According to figure 1, in the first step, the proper dataset for the simulation has to be made. A random dataset from NSL dataset records will be considered as the main dataset for all simulations in this paper. Unlike KDD, NSL dataset does not contain redundant records. This is one of the enhanced attributes of NSL in comparison to KDD. In the Feature Selection by GA phase, proper features based on every different label will be extracted. For every label, the random dataset will turn to a modified one which is 2-labeled. Note that at every simulation based on different labels, there are only two classes. One class of specific labels and the other class for all other labels. For example, if GA tries to obtain proper feature for R2L, then it would be considered as one class in the main database and all the other attack types – Dos, Probe and U2R – and Normal records would be considered as the other class. GA receives this modified 2-labled dataset and by the fitness function [32] attempts to capture a proper combination of features which minimize the numbers of misclassified records for that specific label. At the end of this phase, the proposed feature by GA based on different labels will be obtainable. The GA-proposed features have been shown in "Appendix A". Each row is corresponded to the specific feature and shows it has been proposed by which GA-label based simulation. For example, feature number 1 has been proposed by a GA simulation which has an objective to extract proper feature for Probe attack records and similarly for a GA to extract appropriate features for Normal ones. By looking at each column, the proposed features for each label will be accessible. Therefore, these proposed features can make five different datasets for every label. Every dataset contains all the records of the random dataset but with the features that GA proposes in the previous phase. For example, based on "Appendix A", Dos dataset has 15 features. That means GA by running the classifier on 2-lables dataset of Dos has suggested these 15 features. Hence the random dataset with only Dos feature merely contains these 15 features. Intended for obtaining sufficient information from all classifiers, a voting module has been conducted. The classifiers will be executed on these five datasets and furthermore, the voting of their predictions will be acquired in the new dataset called GA-dataset. This way, all four classifiers opinions have been considered for further simulations. A class which is related to the majority of classifiers prediction will be considered as the final candidate for that specific record in GA-dataset. Since there are five labels and consequently five datasets, there will be five voted results as well. Thus this dataset has the same

number of rows as the main dataset and of course five columns for five different voted predictions in respect of five different labels. In the next step KELM optimized by GA will be run on the built GA-dataset and finally, the results will be gathered based on IDS criteria. In the following discussion, the performance of these two parts (KELM and output method) solely will be examined.

Using the output method – building GA-dataset based on different dataset built by GA suggested features – will help the model to be educated about the information of different behavior of the labels. As it has been considered in the presentation in the next section, this information helps most of the classifiers to perform better in most criteria. Using this method alongside a fast [33], accurate and well-studied classification algorithm like KELM [34–36] is able to promise enhanced results. GA will set parameter of KELM for improved performance. Simulation and comparison of this method with the conventional method will prove our claim.

3. Simulations and results

Working on the entire dataset seems a very cumbersome task. Therefore, many researchers choose random subsets of the original dataset as their main dataset [24, 28–30, 37–39]. In this research, a random dataset has been employed as the main benchmark. Dissimilar to NSL, some records in original KDD dataset are repetitive [39].

In the first simulation, four well-known classification algorithms which build their model on the entire features of the dataset have been considered – we will refer to this simulation as General Simulation. Table 1 demonstrates the performance of these simulations. Before analyzing the outcomes, the specific criteria for evaluating an IDS need to be explained; accuracy (A), detection rate (DR) and false alarm rate (FAR). These measures will be defined as follows:

$$A = (TP + TN)/(TP + TN + FP + FN) \quad (1)$$

$$DR = TP/(TP + FP) \quad (2)$$

$$FAR = FP/(FP + TN) \quad (3)$$

Where:

- TP: True Positive: number of Attack records which were classified as Attack
- TN: True Negative: number of Normal records which were classified as Normal
- FP: False Positive: number of Normal records which were classified as Attack
- FN: False Negative: number of Attack records which were classified as Normal

A superior IDS has higher accuracy and DR and lower FAR. In table 1, best results have been bolded.

Table 1. General simulation performance.

Method	Criteria	DOS	Probe	R2L	U2R	Normal
DT	Accuracy	**96.74**	**95.31**	**71**	92.52	**97.45**
KNN		96.62	92.22	65.46	92.05	83.11
MLP		85.83	82.27	62.64	87.59	84.31
SVM		96.4	94	65.66	**93.36**	82.99
DT	False alarm	**3.6**	**3.6**	**3.6**	**3.6**	**1.88**
KNN		4	4	4	4	23.74
MLP		12.33	12.33	12.33	11.92	24.7
SVM		4.3	4.3	4.3	4.3	23.53
DT	Detection rate	**95.29**	**95.13**	**73.72**	41.93	**96.98**
KNN		94.81	94.23	35.48	38.46	68.27
MLP		82.43	80.52	16.35	14.66	70.2
SVM		94.43	94.11	39.43	**49.41**	67.63

Bold numbers indicate best results in each criteria and attack type

Best results in table 1 suggest DT are superior to others. In the matter of time consumption DT plays the superior role as well. For this reason, DT will be selected as the main classification algorithms for fitness function of GA which intended for selecting proper features. R2L and U2R results are not as proper as other labels. This paper tends to improve their results to reach a reliable level for IDSs.

Table 2 shows the performance of popular classifiers on the GA-dataset (we refer to these simulations as output simulation based on output method).

Table 2 shows how output method assists the classifiers to perform better. For example, for DT, the results of all labels except Normal and R2L show improvement. In several scales for most classifiers, the improvement is evident. The output method is able to easily outperform general simulation in the case of all attack types detections. However, in Normal detections, the proposed output method is not working appropriately. A comprehensive IDS should be able to precisely detect Normal records as well; otherwise, the reliability of this kind of IDS will be simply decreased. The following figure will help to observe improvement of output in comparison with the general method.

Table 2. Classifiers performances on GA-dataset.

Method	Criteria	Dos	Probe	R2L	U2R	Normal
DT	Accuracy	97.14	95.94	69.8	93.45	86.51
KNN		97.02	95.08	**75.06**	93.27	**87.46**
MLP		**97.42**	**96.51**	68.53	**95.14**	87.31
SVM		96.51	96.28	69.53	92.80	85.3
DT	False alarm	3.4	3.4	3.4	3.4	18.73
KNN		3.2	3.2	3.2	3.2	**17.31**
MLP		**2.5**	**2.5**	**2.5**	**2.5**	18.16
SVM		4	4	4	4	20.14
DT	Detection rate	95.57	95.44	70.43	50	72.85
KNN		95.8	95.60	**83.15**	48.38	**74.12**
MLP		**96.81**	**96.74**	68.42	**63.63**	74.10
SVM		94.79	94.77	67.48	45.20	70.80

Bold numbers indicate best results in each criteria and attack type

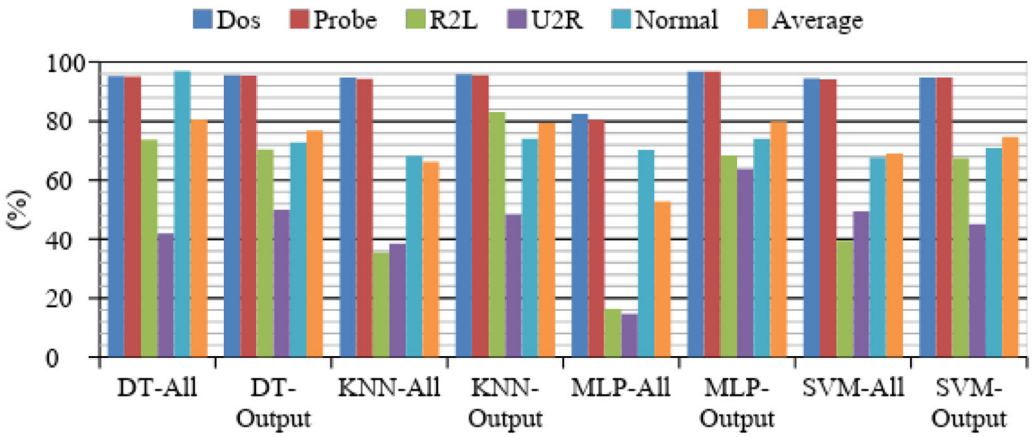

Figure 2. Comparison of detection rate of general and output simulations.

Figure 2 suggests that output method is able to assist the system to detect the label in a more efficient way. DT-All means DT simulation on all features – general simulation – while DT-outputs refer to DT simulation on GA-dataset – output simulation. DR of four classification algorithms in general and outputs simulations – advocate the claim of output method superior performance. MLP performances through output method show the biggest improvement. As a matter of fact, best Dos, Probe and U2R detection performances belong to MLP on GA-dataset. In general, KNN and MLP have superior outcomes while in general simulation their results were inferior to other classifiers. That infers the efficiency of GA-dataset which gathers valuable information from all classifiers by putting their outcomes into the polls.

Our study suggests using KELM as the final classifiers for GA-dataset. Before accomplishing this simulation, KELM has been addressed for general simulation. In table 3 the results of simulation of KELM on the entire features have been shown.

Table 3 suggests that using merely KELM on all features is not a good idea. The poor results for every label and criteria suggest an IDS based on this simulation would be inapplicable in the real world. However, Normal outcomes are interestingly higher than other simulations, especially in DR and FAR. Output method was able to outperform general simulation for attack detection; nevertheless, Normal detection was highly inferior to general simulation. On the other hand, KELM simulations interestingly show exactly the opposite performance to the output method. This observation encourages the idea of using KELM on GA-dataset. For the conclusion, proposed output method and KELM solely are not able to outperform the results of popular classification algorithms on all features. The performance of proposed output method is far better than KELM on all features in attack detection and vice versa for Normal detections.

For enhancement of KELM performance, GA has been utilized for parameter setting. KELM based on C parameter as regularization coefficient and kernel-para as vectors of kernel parameters have different outcomes [40]. GA fitness function attempts to minimize the defined Z scale as shown in equation (4):

$$Z = 2 - (A_{avg} + DR_{avg}) + FAR_{avg} \quad (4)$$

where A_{avg} refers to the average accuracy for the following five labels: similarly, DR_{avg} and FAR_{avg} refer to the average of DR and FAR of the labels. The maximum value of (A_{avg} + DR_{avg}) will be equal to 2 while FAR_{avg} best results equal to zero. Best possible outcome for Z is 0 and higher accuracy and detection rate and lower false alarm rate will assist Z to become closer to this value which is precisely what a high-quality IDS requires. Consequently, the best outcome of Z perfectly reflects the whole system performance based on defined IDS criterion. GA attempts to minimize Z value by manipulation of C and kernel-para. In this study, GA proposes 22 for C and 108 for kernel-para parameters.

In the next simulation, the proposed method will be tested. The following table shows the results of the proposed method.

As it has been shown in table 4, the results of all labels and criteria are promising. The DR of attack types

Table 3. KELM general simulation performance.

Criteria	Dos	Probe	R2L	U2R	Normal
Detection rate	46.568	47.114	5.729	0.549	98.28
False alarm rate	72.4	72.4	72.4	72.4	1.5
Accuracy	51.828	52.628	21.333	26.168	68.787

Table 4. GA-KELM on GA-dataset.

	DOS	Probe	R2L	U2R	Normal
Detection rate	97.529	97.52	96.339	78.651	99.99
False alarm rate	1.9	1.9	1.9	1.9	0.1
Accuracy	98.914	98.914	98.733	98.224	99.381

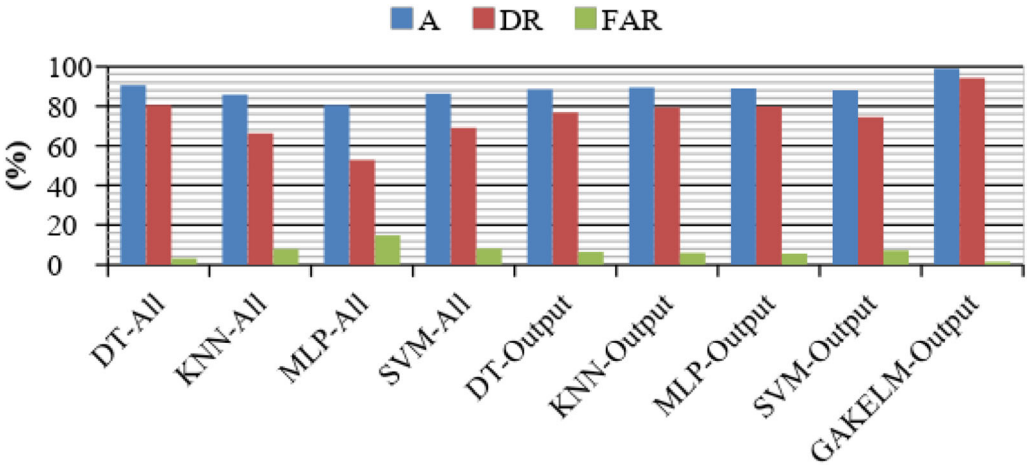

Figure 3. Comparison of general and output simulations performances based on averaged results.

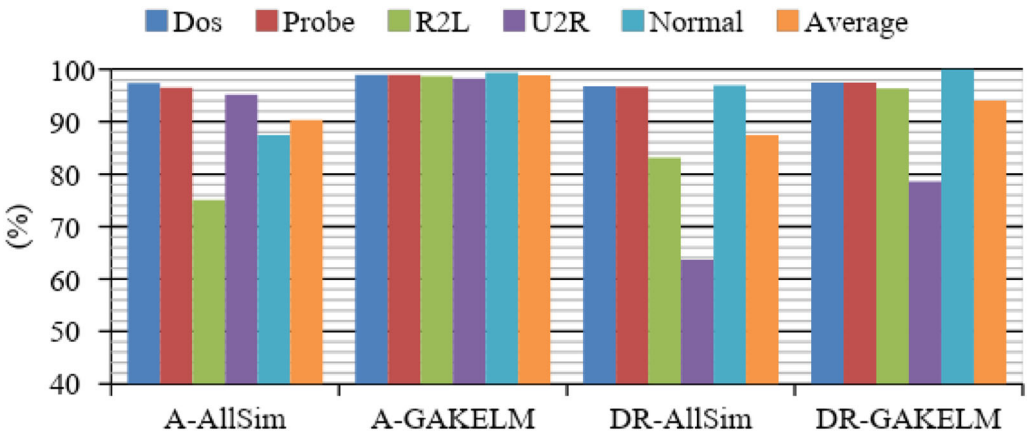

Figure 4. Comparisons of best results of DR and accuracy for general and output simulations.

especially R2L and U2R shows big improvement. Normal records also confirm impressive enhancement. The efficient combination of KELM and output method turns to enhance all attack types and Normal records detections. Output method's interesting performance on attack detection and general KELM performance for Normal detection have been combined in the proposed GA-KELM output method.

For better presentation, suitable comparisons between the proposed method and other simulation have been established. In the first comparison, the average results of every method based on every criterion have been shown. That means for every method average of five labels' results has been calculated and represented.

Figure 3 shows the average results of every simulation in order to every criterion. In every case, the average of five results of every label has been calculated. Type of simulation has been indicated in the axis label. GAKELM-outputs – simulation of KELM optimized by GA on GA-dataset- shows superior performance. In figure 3 best results of simulations belong to DT as well. This is another reason why this classifier has been considered as the main

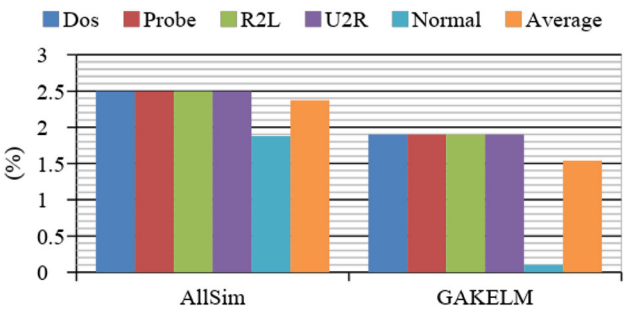

Figure 5. Comparisons of best results of FAR for general and output simulations.

classifier in fitness function of GA in feature selection phase. While MLP performance on general simulation was worst among all four classifiers, this classification algorithm performance in output simulations confirms MLP's average performance is better than all other classifiers.

Average results are a suitable way to evaluate a system performance. However, best results of every method based

Table 5. Comparison of detection rate (DR) of proposed method with recent studies.

Methods	Dos	Probe	R2L	U2R	Normal	Average
OBFE [29]	**98.96**	96.9	87.9	72.54	98.01	90.86
TVCPSO-MCLP/SVM [30]	98.84	89.29	75.08	59.62	99.13	84.392
Proposed method	97.52	**97.53**	**96.34**	**78.65**	**99.99**	**94.01**

Bold numbers indicate best results in each category

Table 6. Comparison of detection rate of the proposed method with recent studies in deep learning networks.

Methods	Dataset	Detection rate
[40] (CNN)	KDD Cup 99	96.30
[41] Deep belief networks	KDD Cup 99	97.81
Proposed method	KDD Cup 99	**97.88**
[41] Deep belief networks	NSL-KDD	88.10
[42] Deep belief networks	NSL-KDD	92.33
Proposed Method	NSL-KDD	**94.01**

Bold numbers indicate best results in each data set

on different criteria and labels are more essential. In figure 4, the best results in every simulation have been compared.

Figure 4 compares the best results of every previous simulation and GAKELM simulation in respect of DR and accuracy. A-AllSim refers to all other simulations which have been made so far except KELM – simulation on all features and GA-dataset. In all labels the proposed method shows better results rather than any other simulation. In terms of accuracy, the enhancements of proposed method are more appealing. However, proposed method DR shows superior performance as well. In R2L and U2R detections – as two of more difficult attack types in the case of detection – proposed method works very well.

The next figure compares the best results of the only remaining criteria – FAR – of previous simulations and proposed method.

According to figure 5 in every label, the proposed method suggests a better result. For Normal label, the proposed method makes the results significantly better. Average results as an important scale in IDS comparison demonstrate the proposed method ability as well.

For better comparison, our proposed IDS has been compared with two recent papers which report the highest detection rate (DR) on this dataset. The DR of provided IDS in [29] and [30] is gathered for evaluation.

Table 5 demonstrates the proposed method ability in IDS label detection compared to recent studies. While Dos detection in the proposed method is inferior to provided studies, other labels are superior to these methods even with bigger margins. In an environment where there is no vision on attack type's abundances, the average of outcomes for all five labels seems a suitable scale for IDS efficiency judgments. As a matter of fact, an IDS with specific ability of excellent detection for one attack type and not appealing detections capability for other attack types, might not seem as interesting as an IDS with admirable detection ability for all attack types. Specifically, R2L and U2R results which are obviously harder to detect based on standard datasets, encourage proposed method efficiency. Because of using DT and KEML in the process of generating proposed IDS, the speed of building model is quite acceptable; especially for online learning IDSs where the time consumption related problem should be taken much more seriously. KELM alongside a well treatment with features based on various labels – output method – is able to push the IDS criteria to an unreachable level in comparison with other mentioned simulations in this paper.

Moreover, we provided another comparison with more recent state of the art Deep Learning Networks. Table 6 shows this comparison. We also made a simulation on KDD CUP 99 dataset with randomly selected records.

4. Conclusion

This paper proposed an efficient hybrid method based on predictions of four well-known classification algorithms and optimized KELM by GA. NSL's – enhanced version of KDD CUP 99 – effect of course of dimensionality suggests features selection algorithms are able to enhance the performance. GA has been utilized for feature selection sake. By voting on four famous classifiers' prediction which built their models on constructed dataset by GA suggested features, GA-dataset has been assembled. Simulations on this proposed dataset are able to outperform attack detections of general simulation which works on the original dataset with entire features. However, this method is not able to detect normal records properly. Although KELM simulation on all features illustrates disappointing outcomes in attack labels detections, it has interestingly high-quality performance for normal detection. These two procedures construct our main intuition for using these both methods alongside each other. KELM optimized by GA which has been executed on GA-dataset is capable of exploiting both method advantages. Hence proposed GA-KELM simulation on proposed GA-dataset as results demonstrate is able to reach best results in all label detections in comparison to general simulation and recent studies.

Acknowledgements

The authors would like to express their gratitude toward reference [43] for putting NSL-KDD dataset in public access.

Abbreviations

IDS	Intrusion detection system
GA	Genetic algorithms
KELM	Kernel extreme learning machine
DT	Decision TREE
KNN	K-nearest neighbor
MLP	Multilayer perceptron
SVM	Support vector machine
KPCA	Kernel principle component analysis
CFA	Cuttlefish algorithm
GHSOMs	Growing hierarchical self-organizing maps
FDR	Fisher discriminant ratio
OS-ELM	Online sequential extreme learning machine
TVCPSO	Time-varying chaos particle swarm optimization
MCLP	Multiple criteria linear programming
R2L	Remote-to-local
U2	User-to-root

Appendix A: Proposed feature by GA

Features	Normal	Dos	Probe	R2l	U2R
1	1	0	1	0	0
2	1	1	0	1	0
3	1	1	1	1	1
4	1	1	1	0	0
5	1	0	1	0	1
6	1	1	0	1	1
7	0	1	1	1	0
8	0	0	1	1	0
9	1	1	0	1	0
10	1	0	0	0	1
11	0	1	1	1	0
12	0	0	1	0	0
13	1	0	0	0	0
14	1	0	1	0	0
15	1	0	0	1	0
16	1	0	1	0	0
17	1	0	0	1	0
18	1	1	1	0	0
19	0	1	0	1	1
20	1	0	1	1	0
21	0	1	1	1	1
22	1	0	0	1	1
23	0	1	1	0	0
24	0	0	0	0	0
25	0	0	0	1	1
26	1	0	0	0	0
27	1	0	1	1	0
28	1	0	0	1	1
29	0	0	0	1	0
30	0	1	1	1	0
31	1	1	0	1	0
32	0	0	0	0	0
33	0	1	0	1	0
34	1	0	0	0	0
35	1	0	1	1	0
36	0	1	1	0	0
37	1	0	1	1	0
38	1	0	0	0	0
39	1	0	0	0	1
40	0	0	0	1	0
41	0	0	1	0	0

References

[1] Inayat Z, Gani A, Anuar N B, Khan M K and Anwar S 2016 Intrusion response systems: Foundations, design, and challenges. *J. Netw. Comput. Appl.* 62: 53–74

[2] Elhag S, Fernández A, Bawakid A, Alshomrani S and Herrera F 2015 On the combination of genetic fuzzy systems and pairwise learning for improving detection rates on intrusion detection systems. *Expert Syst. Appl.* 42: 193–202

[3] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99/ Accessed 10.28.99

[4] http://www.unb.ca/research/iscx/dataset/iscx-NSL-KDD-dataset/Accessed 2015

[5] Tavallaee M, Bagheri E, Lu W and Ghorbani A 2009 A detailed analysis of the KDD CUP 99 data set. In: *Proceeding of the IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp. 53–58

[6] de la Hoz E *et al* 2013 Network anomaly classification by support vector classifiers ensemble and non-linear projection techniques. *Hybrid Artif. Intell. Syst.* 103–111

[7] Singh R, Kumar H and Singla R 2015 An intrusion detection system using network traffic profiling and online sequential extreme learning machine. *Expert Syst. Appl.* 42: 8609–8624

[8] Lakhina S, Joseph S and Verma B 2010 Feature reduction using principal component analysis for effective anomaly–based intrusion detection on NSL-KDD. *Int. J. Eng. Sci. Technol.* 2: 1790–1799

[9] Jain A K, Duin R P W and Jianchang M 2000 Statistical pattern recognition: a review. *Pattern Anal. Mach. Intell.* 22: 4–37

[10] Trunk G V 1979 A problem of dimensionality: a simple example. *Pattern Anal. Mach. Intell.* PAMI 1: 306–307

[11] Goldberg D E *et al* 1989 Genetic Algorithms in Search Optimization and Machine Learning, vol. 412 pp. 211

[12] Li W 2004 Using genetic algorithm for network intrusion detection. In: *Proceedings of the United States Department of Energy Cyber Security Group*, pp. 1–8

[13] Fidelis M V, Lopes H and Freitas A 2000 Discovering comprehensible classification rules with a genetic algorithm. In: *Proceedings of the 2000 Congress on Evolutionary Computation* 1, pp. 805–810

[14] Rastegari S, Hingston P and Lam C-P 2015 Evolving statistical rule sets for network intrusion detection. *Appl. Soft Comput.* 33: 348–359

[15] Eesa A S, Orman Z and Brifcani A M A 2015 Novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems. *Expert Syst. Appl.* 42: 2670–2679

[16] Sindhu S S S, Geetha S and Kannan A 2012 Decision tree based light weight intrusion detection using a wrapper approach. *Expert Syst. Appl.* 39: 129–141

[17] Kim G, Lee S B and Kim S 2014 A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. *Expert Syst. Appl.* 41: 1690–1700

[18] Saied A, Overill R E and Radzik T 2016 Detection of known and unknown DDoS attacks using artificial neural networks. *Neurocomputing* 172: 385–393

[19] Wang G, Hao J, Mab J and Huang L 2010 A new approach to intrusion detection using artificial neural networks and fuzzy clustering. *Expert Syst. Appl.* 37: 6225–6232

[20] Meng W, Li W and Kwok L-F 2014 EFM: enhancing the performance of signature-based network intrusion detection systems using enhanced filter mechanism. *Comput. Secur.* 43: 189–204

[21] Lin W-C, Ke S-W and Tsai C-F 2015 CANN: an intrusion detection system based on combining cluster centers and nearest neighbors. *Knowl. Based Syst.* 78: 13–21

[22] Erfani S M, Rajasegarar S, Karunasekera S and Leckie C 2016 High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. *Pattern Recognit.* 58: 121–134

[23] Catania C A, Bromberg F and Garino CG 2012 An autonomous labeling approach to support vector machines algorithms for network traffic anomaly detection. *Expert Syst. Appl.* 39: 1822–1829

[24] Kuanga F, Xua W and Zhang S 2014 A novel hybrid KPCA and SVM with GA model for intrusion detection. *Appl. Soft Comput.* 18: 178–184

[25] Tsang C-H, Kwong S and Wang H 2007 Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection. *Pattern Recognit.* 40: 2373–2391

[26] Shafi K and Abbass HA 2009 An adaptive genetic-based signature learning system for intrusion detection. *Expert Syst. Appl.* 36: 12036–12043

[27] de la Hoz E, de la Hoz E, Ortiz A, Ortega J and Martínez-Álvarez A 2014 Feature selection by multi-objective optimization: application to network anomaly detection by hierarchical self-organizing maps. *Knowl. Based Syst.* 71: 322–338

[28] De la Hoz E, De La Hoz E, Ortiz A, Ortega J and Prieto B 2015 PCA filtering and probabilistic SOM for network intrusion detection. *Neurocomputing* 164: 71–78

[29] Monowar, Bhuyan H, Bhattacharyya D K and Kalita J K 2016 A multi-step outlier-based anomaly detection approach to network-wide traffic. *Inf. Sci.* 348: 243–271

[30] Bamakan S M H, Wang H, Yingjie T and Shi Y 2016 An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization. *Neurocomputing* 199: 90–102

[31] Fossaceca J M , Mazzuchi T A and Sarkani S 2015 MARK-ELM: application of a novel multiple kernel learning framework for improving the robustness of network intrusion detection. *Expert Syst. Appl.* 42: 4062–4080

[32] Reeves C R and Rowe J E 2003 *Genetic algorithms: principles and perspectives: a guide to GA theory*. US: Springer, vol. 20, pp 112

[33] Avci E and Coteli R 2012 A new automatic target recognition system based on wavelet extreme learning machine. *Expert Syst. Appl.* 39: 12340–12348

[34] Cheng C, Tay W P and Huang G B 2012 Extreme learning machines for intrusion detection. In: *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8

[35] Creech G and Jiang F 2012 The application of extreme learning machines to the network intrusion detection problem. In: *Numerical Analysis and Applied Mathematics ICNAAM: International Conference of Numerical Analysis and Applied Mathematics* 1479, pp. 1506–1511

[36] de Farias G P M, de Oliveira A L and Cabral G G 2012 Extreme learning machines for intrusion detection systems. *Neural Inf. Process.* 535–543

[37] Amiri F, Rezaei Yousefi M, Lucas C, Shakery A and Yazdani N 2011 Mutual information-based feature selection for intrusion detection systems. *J. Netw. Comput. Appl.* 34: 1184–1199

[38] Sangkatsanee P, Wattanapongsakorn N and Charnsripinyo C 2011 Practical real-time intrusion detection using machine learning approaches. *Comput. Commun.* 34: 2227–2235

[39] Pereira C R, Nakamura R Y M, Costa K A P and Papa J P 2012 An optimum-path forest framework for intrusion detection in computer networks. *Eng. Appl. Artif. Intell.* 25: 1226–1234

[40] Liu Y, Liu S and Zhao X 2017 Intrusion detection algorithm based on convolutional neural network. In: *State Key Laboratory of Mathematical Engineering and Advanced Computing*, *Zhengzhou, Henan, China 2017 4th International Conference on Engineering Technology and Application* ISBN: 978-1-60595-527-8

[41] Shone N, Ngoc T N , Phai V D and Shi Q 2017 A deep learning approach to network intrusion detection. *IEEE Trans. Emerg. Top. Comput. Intell.* 2: 1–16

[42] Xin Y, Kong L, Liu Z, Chen Y, Li Y, Zhu H, Gao M, Hou H and Wang C 2017 Machine learning and deep learning methods for cybersecurity. *IEEE Transl. Content Min.* 1: 1–10

[43] http://www.ntu.edu.sg/home/egbhuang/elm_kernel/ Accessed 2013