ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 15, No. 1, 2020, *pp*.022-030

CRF BASED INTRUSION DETECTION SYSTEM USING GENETIC SEARCH FEATURE SELECTION FOR NSSA

Azhagiri Mahendiran [1], Rajesh Appusamy [2] , Rajesh Prabhakaran[3] and Gowtham Sethupathi[4]
[1] Computer [1]Science and Engineering, SRM Institute of Science and Technology, Chennai, India
[2] Computer Science and Engineering , C.Abdul Hakeem College of Engineering and Technology, Melvisharam, Tamilnadu, India.
[3] Computer Science and Engineering , Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India.
[4] Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India
(Received December 16, 2019, accepted January 25, 2020)

Abstract - Network security situational awareness systems helps in better managing the security concerns of a network, by monitoring for any anomalies in the network connections and recommending remedial actions upon detecting an attack. An Intrusion Detection System helps in identifying the security concerns of a network, by monitoring for any anomalies in the network connections. We have proposed a CRF based IDS system using genetic search feature selection algorithm for network security situational awareness to detect any anomalies in the network. The conditional random fields being discriminative models are capable of directly modeling the conditional probabilities rather than joint probabilities there by achieving better classification accuracy. The genetic search feature selection algorithm is capable of identifying the optimal subset among the features based on the best population of features associated with the target class. The proposed system, when trained and tested on the bench mark NSL-KDD dataset exhibited higher accuracy in identifying an attack and also classifying the attack category.

Index terms: Network Security Situational Awareness (NSSA), Intrusion Detection System (IDS), Network Security, Intelligent Systems, Conditional Random Fields(CRF), Feature selection, Machine learning.

1. Introduction.

The term situational awareness is used in military combat operations to denote "the ability to identify, process, and comprehend the critical elements of information about what is happening to the team with regards to the mission" [1]. Network security situational awareness (NSSA) is the ability to assess the current state of a network based on inputs provided by various sensors at different levels of the network [2]. This is quite a difficult task considering the volume of transactions done on any kind of network.

The NSSA operates at four different levels as in [4]:

- Acquiring information from intrusion detection systems (IDS), firewall logs, scan reports etc.
- Analyze the received information for evidences of any threat.
- Predict future threats based on the information learned from inputs such as IDS, firewall logs, scan reports etc.
- Recommend remedial actions to address a security event when it happens.

In order for the NSSA to function effectively, identification of anomalies in a network is of great importance. Intrusion detection is the process of identifying activities on a network that are violating the security policies of the network [3]. Intrusions affect the integrity, confidentiality of the information on the network and prevent accessibility of the information sources on the network [5, 6, 7]. An IDS with high accuracy will aid in better functioning of Network Security Situational Awareness (NSSA) System. Hence, in this paper we have proposed an IDS that is capable of detecting attacks accurately so that it can be effectively used in a NSSA system.

Our contributions in this research,

[1] Corresponding author. Tel.: +91-9865958062
E-mail address: azhagiri1687@gmail.com.

- An IDS using Conditional Random Field (CRF), capable of detecting various attack categories with high accuracy.
- Identification of a feature selection method for selecting the features that result in optimal operation of the CRF classifier.

The system proposed in [17] also uses CRF based classifier. The proposed system differs from the system in [17] as follows:

The system in [17] uses 4 layers of binary CRF classifier each capable of predicting one of the 4 attack categories whereas our system comprises of a single multi class CRF classifier capable of predicting all 4 attack categories. The system in [17] uses manual feature selection whereas our system uses an automatic feature selection method.

The rest of the paper is organized as follows: Section II describes several state of the art IDS in the literature. Section III describes the proposed system. Section IV discusses the results obtained by the proposed system and Section V concludes this research.

2. Related Work

In this section a brief discussion of some of the state of the art IDS researched in the literature are given.

In [8] the authors have used multiclass support vector machine to identify the various attacks on a network. The chi-square feature selection method was used to reduce the dimensionality of the dataset and choose appropriate attributes for building the model.

In [9] the authors have used a fuzzy based semi-supervised learning approach to efficiently utilize the unlabeled samples and used supervised learning algorithm to improve the performance of the IDS. A single hidden layer feed forward neural network is used for building the model. In the first stage, the unlabelled samples are categorized using a fuzzy quantification process. The categorized output from the first stage is then used to retrain the neural network.

In [10] an anomaly based network intrusion detection system using feature correlation analysis and association impact scale to predict intrusions has been proposed. The usage feature correlation significantly minimized the computational time of measuring association impact.

In [11] the authors have proposed a multi-level hybrid intrusion detection model using support vector machine and extreme learning machine. A modified K means algorithm have been used to significantly improve the quality of the training dataset. This has resulted in reduced training time of the classifiers and also resulted in improved performance of the IDS.

In [12] a modified optimum path forest algorithm [OPF] has been used. The training samples were divided into homogeneous subsets using k-means clustering algorithm. This has resulted in improved scalability, accuracy, detection rate, false alarm rate and execution time than traditional OPF.

In [13] the authors propose a fuzzy membership function which reduces considerably the computational complexity of the intrusion detection process and at the same time increases the accuracies of the classifier algorithms.

In [14] an anomaly based intrusion detection system using hierarchically structured learning automata has been proposed. The automaton learns to choose the optimal action through repeated interactions with the environment thereby resulting in a highly resilient approach that excels in detecting unknown attacks.

In [15] a hybrid feature selection method for intrusion detection has been proposed. The authors have used binary gravitational search algorithm with mutual information based filter for pruning the subset of features. The search direction is controlled using a two objective fitness function to maximize detection rate and minimizing false positive rate. This led to a increase in accuracy and detection rate compared to other wrapper based and filter based methods.

In [16] a hybrid approach integrating evolutionary algorithm with neural networks has been proposed. The authors have come up with two hybrids - gravitational search and gravitational search along with particle swarm optimization to train artificial neural networks. They have shown that these hybrid approaches have out run traditional IDS.

In [17] a layered approach for intrusion detection using conditional random fields has been proposed. The conditional random field achieves high detection accuracy and layered approach helps in improving the efficiency of the detection process. The authors have conducted statistical tests to prove the higher detection accuracy of their method.

The IDS discussed in the literature show good performance at over all detection of an attack where as fails in identifying individual attack categories with the same high accuracy (Table 8). An NSSA system, in order to initiate remedial actions to address a security event needs the type of attack involved in the event [4]. Hence, the IDS part of it should be capable of accurately detecting the various attack categories uniformly. Hence, our focus in this research is in designing an IDS capable of identifying the various attack categories with high accuracy.

3. Proposed System

In this research, we have used the linear chain conditional random field (CRF) (Fig. 1) for classifying a normal connection from an attack. The CRF is a conditional model that models conditional distributions over a set of random variables and can be described as in [18] as follows:
X – Random variable over data sequence to be labeled
Y – Label sequence
G – A graph defined as, G = (V,E)
Let Y = (Yv)vϵ(V) i.e. Y is indexed by the vertices of G
(X,Y) is a CRF if when conditioned on X, the random variables Yv obey the Markov property with respect to the graph: p(Yv |X,Yw,w ≠ v) = p(Yv |X,Yw,w ~ v), where w ~ v means w and v are neighbors in G.

The joint distribution over the label sequence Y given X for a simple sequential (chain) modeling has the form $p(y|x) \propto exp\left(\sum_{e\in E,k} \lambda_k f_k(e, y|_e, x) + \sum_{v\epsilon V,k} \mu_k g_k(v, y|_v, x)\right)$
Where x – data sequence, y – label sequence

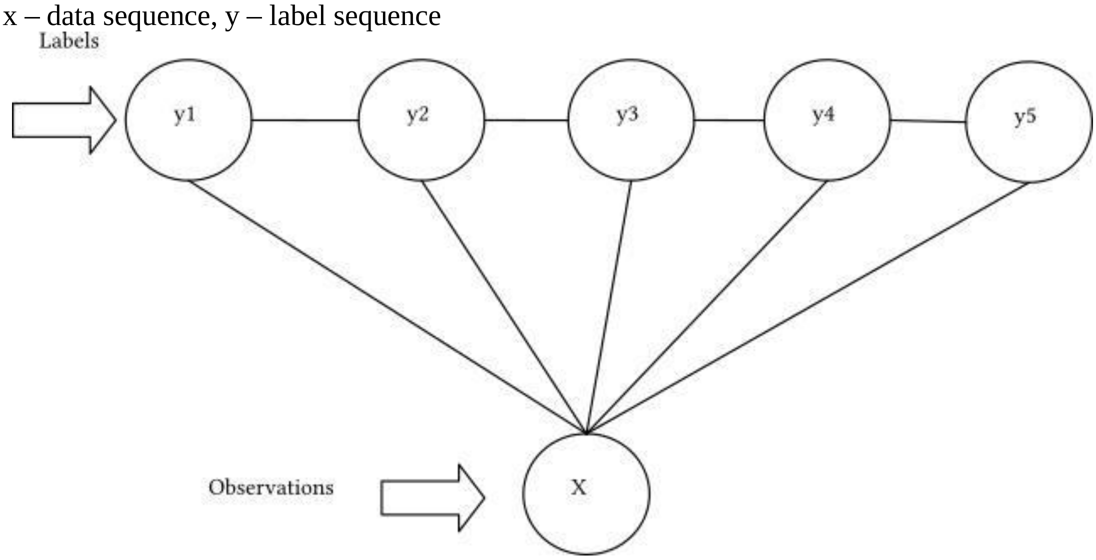

Fig. 1. Graphical Representation of Linear Chain CRF

y|S – set of components of y associated with the vertices in sub graph S

In Fig. 1, the observations are the attributes (features) describing the connection and the labels can be one of the following - “dos”, “u2r”, “r2l”, “probe” and “normal” respectively. We have used the R [22, 23] and WEKA [24] tools to perform our experimentations

We have used KDDTrain+ data from the bench mark NSL-KDD dataset [19] for training and testing our system. The NSL-KDD dataset is an improved version obtained by eliminating the pitfalls in KDDcup99 dataset as identified in [20]. The KDDTraint+ data contains 125,973 records of simulated connection information labeled as either normal or a particular type of attack. The data contains records of 22 attack types along with the normal records. The attack types can be grouped into one of the following four main attack categories:

- DOS: denial-of-service, e.g. syn flood;
- R2L: unauthorized access from a remote machine, e.g. guessing password;
- U2R: unauthorized access to local superuser (root) privileges, e.g., various ``buffer overflow'' attacks;
- Probing: surveillance and other probing, e.g., port scanning.

Each record in the dataset contains the 41 attributes listed in Table 1 along with the label.

Table 1: Features in the NSL-KDD Dataset

Sr. No	Feature Name	Sr. No	Feature Name
1	Duration	22	Is_guest_login
2	Protocol_type	23	Count
3	Service	24	Srv_count
4	Flag	25	Serror_rate
5	Src_bytes	26	Srv_serror_rate
6	Dst_bytes	27	Rerror_rate
7	Land	28	Srv_rerror_rate
8	Wrong_fragment	29	Same_srv_rate
9	Urgent	30	Diff_srv_rate
10	Hot	31	Srv_diff_host_rate
11	Num_failed_logins	32	Dst_host_count
12	Logged_in	33	Dst_host_srv_count
13	Num_compromised	34	Dst_host_same_srv_rate
14	Root_shell	35	Dst_host_diff_srv_rate
15	Su_attempted	36	Dst_host_same_src_port_rate
16	Num_root	37	Dst_host_srv_diff_host_rate
17	Num_file_creations	38	Dst_host_serror_rate
18	Num_shells	39	Dst_host_srv_serror_rate
19	Num_access_files	40	Dst_host_rerror_rate
20	Num_outbound_cmds	41	Dst_host_srv_rerror_rate
21	Is_host_login		

To build and test our proposed system, we have taken a sample of 500 records of the KDDTrain+ data with the attack/normal data distribution as in Table 2.

The CRF implementation in R works only with numerical input, so all the nominal features in the dataset was converted to numeric type by replacing their nominal values with their respective levels. This is then followed by normalization of the features. After normalization, the following attributes – "land", "num_outBound_cmds" and "is_host_login" were found to contain non-numeric values and hence was removed. The normalized dataset with the remaining 39 features was then used to train and test our proposed system.

Table 2: Characteristics of the Sample KDD train+ Dataset used for the Experimentation

DOS	NORMAL	PROBE	R2L	U2R
50	300	50	50	50

Since the complexity of the CRF increases with the increase in the number of features used to train it [18], we have used feature selection to reduce the number of features required for efficient classification of the connection. To obtain the features that result in efficient operation of the CRF, we have used a genetic search based feature selection approach [21] to select the most appropriate features for classifying the connections as attack or normal. Feature subset selection helps in reducing the hypothesis search space, thereby improving the efficiency of operation of a classifier.

We have used the implementation of the genetic search based feature subset selection algorithm in the WEKA [24] platform to select the optimal subset of features. The output of the selection process is shown in Table 3.

The selected features of the dataset were then used as the observation sequence and the CRF was trained. We have used 10-fold cross validation to train and test the dataset.

4. Results and Discussion

The confusion matrix of our experimentation is shown in Table 4. The overall accuracy of our proposed system is shown in Table 5. The precision, recall and f-measure obtained by our proposed system for each of the connection types are shown in Table 6. It can be seen from the results obtained that the proposed system is capable of detecting the different attack categories individually with good accuracy.

Table 7, Table 8 and Fig. 2 show the performance comparison of the proposed system with some of the state of the art IDS in the literature. Though some systems have shown higher overall attack detection accuracy, their capability in classifying the attack type is non-uniform. Their accuracy in detecting "u2r" and "r2l" attacks is relatively low. In Table 8 only the systems that have given performance in terms of individual attack category types is shown. It can be seen from the comparisons that the proposed system shows good performance in terms of both individual attack category detection as well as over all attack detection

Table 3: Ranking of the Features of the KDDTrain+ dataset

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval -P 1 -E 1
Search:       weka.attributeSelection.GeneticSearch -Z 20 -G 20 -C 0.6 -M 0.033 -R 20 -S 1
Relation:     nsample-weka.filters.unsupervised.attribute.Remove-R1
Instances:    500
Attributes:   39
              duration
              protocol_type
              service
              flag
              src_bytes
              dst_bytes
              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
              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
              category
Evaluation mode:    evaluate on all training data
=== Attribute Selection on all input data ===

Search Method:
               Genetic search.
               Start set: no attributes
               Population size: 20
               Number of generations: 20
               Probability of crossover:  0.6
               Probability of mutation:  0.033
               Report frequency: 20
               Random number seed: 1

Initial population
merit          scaled      subset
```

```
 0.31967  0.31975  27
 0.48744  0.63918  3 9 17 19 21 30 36 37 38
 0.35389  0.3849   1 30
 0.33087  0.34107  2 3 4 5 6 8 9 11 12 13 14 15 16 18 23 25 26 28 31 32 34 35 37 38
 0.22884  0.14681  2 3 4 7 8 10 13 14 15 16 17 18 21 25 26 32 34 35 36 38
 0.44329  0.55511  1 7 11 16 17 22 24 27 31 34 35
 0.19841  0.08886  19 38
 0.18237  0.05832  13
 0.26421  0.21414  3 4 10 12 13 18 33 36
 0.3238   0.32761  2 4 5 7 8 9 11 15 18 19 22 23 27 32 34 36 38
 0.38303  0.44037  11 15 16 17 18 20 22 27 29 30 32 33 34 36 37 38
 0.40052  0.47369  1 2 4 13 15 23 25 32
 0.26328  0.21238  1 2 3 6 7 8 9 10 11 14 16 17 19 20 21 28 29 35
 0.43566  0.54059  2 3 5 6 7 9 11 13 14 15 16 19 20 23 24 27 31 34 35 37
 0.26411  0.21396  1 5 7 8 9 14 18 19 21 26 30 33 35 36 37
 0.1861   0.06543  4 8 11 14 18 23 24 31 32 37
 0.25922  0.20464  6 14 18 21 23 30 31 34
 0.35923  0.39507  3 11 14 16 18 20 21 22 23 25 26 27 28 29 30 31 34 35 37
 0.36976  0.4151   5 15
 0.3381   0.35483  4 5 6 8 9 10 12 17 20 23 24 26 29 35 36 38

Generation: 20
merit     scaled   subset
 0.62306  0.78578  1 3 5 6 9 21 26 27 30 34 35 36 38
 0.62306  0.78578  1 3 5 6 9 21 26 27 30 34 35 36 38
 0.51311  0.43844  2 4 6 7 16 17 19 20 21 22 23 30 31 34 36 38
 0.58126  0.65375  1 3 6 9 16 17 19 20 21 22 24 26 27 30 31 32 33 34 35
 0.5563   0.57489  1 3 6 9 16 17 19 20 21 22 23 30 31 34 36 38
 0.55568  0.57292  1 3 6 9 15 17 19 22 25 26 27 30 34 36
 0.57225  0.62527  1 3 6 9 21 23 26 27 30 34 35 36 38
 0.37432  0        1 3 5 6 8 9 14 16 17 19 20 21 22 23 30 34 36 38
 0.58959  0.68006  2 3 4 6 9 26 27 30 34 36 38
 0.56906  0.61519  1 3 4 6 7 11 16 17 19 20 21 22 24 26 27 29 31 32 33 34 35
 0.57221  0.62514  1 3 6 16 26 27 30 36 38
 0.56183  0.59235  1 2 3 6 9 26 29 30 32
 0.40559  0.09879  1 2 3 9 18 24 26 27 30 34 35 36 38
 0.4111   0.11617  1 2 3 9 18 26 27 30 34 35 36 38
 0.55434  0.56869  1 3 6 9 16 17 19 20 21 22 23 30 31 32 33 36 38
 0.5615   0.59133  1 3 6 16 26 27 30 31 36 38
 0.60494  0.72855  1 3 4 6 9 26 27 30 34 36 38
 0.59039  0.68259  1 3 6 9 26 27 30 34 35 36 38
 0.56265  0.59496  1 3 6 16 24 26 27 30 36
 0.57158  0.62317  2 3 4 6 7 16 19 20 21 22 24 26 27 30 32 33 34 35 38
Attribute Subset Evaluator (supervised, Class (nominal): 39 category):
        CFS Subset Evaluator
        Including locally predictive attributes
Selected attributes: 1,3,5,6,9,21,26,27,30,34,35,36,38 : 11
           duration
           service
           src_bytes
           dst_bytes
           hot
           srv_count
           same_srv_rate
           diff_srv_rate
           dst_host_srv_count
           dst_host_srv_diff_host_rate
           dst_host_serror_rate
```

Table 4. Detection Details of the Different Attack Categories of the Proposed System

Attack	DOS	U2R	R2L	PROBE	NORMAL
DOS	50	0	0	0	0
U2R	0	43	0	0	7
R2L	0	0	48	0	2
PROBE	0	0	0	48	2
NORMAL	0	5	1	2	292

Table 5. Classification Statistics of the Proposed System

Total Records	500
Correctly Classified	481
Wrongly Classified	19
Accuracy	96.2

Table 6. Precision, Recall and F-measure of the Proposed System

Attack	Precision	Recall	F-measure
DOS	100	100	100
U2R	89.58	86	87.76
R2L	97.96	96	96.97
PROBE	96	96	96
NORMAL	96.37	97.33	96.85

Table 7. Accuracy of the various IDSs

Methods	Accuracy
Proposed System	98.2
chi-square multiclass SVM	98
Fuzziness semi-supervised IDS	84.12
FCAAIS	90.4
LFCL	99.16
LA-IDS	98.9
Hybrid SVM and ELM	95.75
MI-BGSA	88.36
GSPSO-ANN	98.13
Naıve Bayes and CF-KNN	94.56
modified OPF	91.74
Layered CRF	90

Table 8. Performance Comparison of the various IDSs

Methods	Accuracy					
	OVERALL	DOS	U2R	R2L	PROBE	NORMAL
Proposed System	98.2	100	89.58	97.96	96	96.37
chi-square multiclass SVM	98	99.9	73.9	98.7	99.2	99.6
Hybrid SVM and ELM	95.75	99.54	21.93	31.39	87.22	98.13
Naıve Bayes and CF-KNN	94.56	84.68	67.16	34.81	79.76	94.56
modified OPF	91.74	96.89	77.98	81.13	85.92	98.55
Layered CRF	90	97.4	86.33	29.62	98.62	98.62

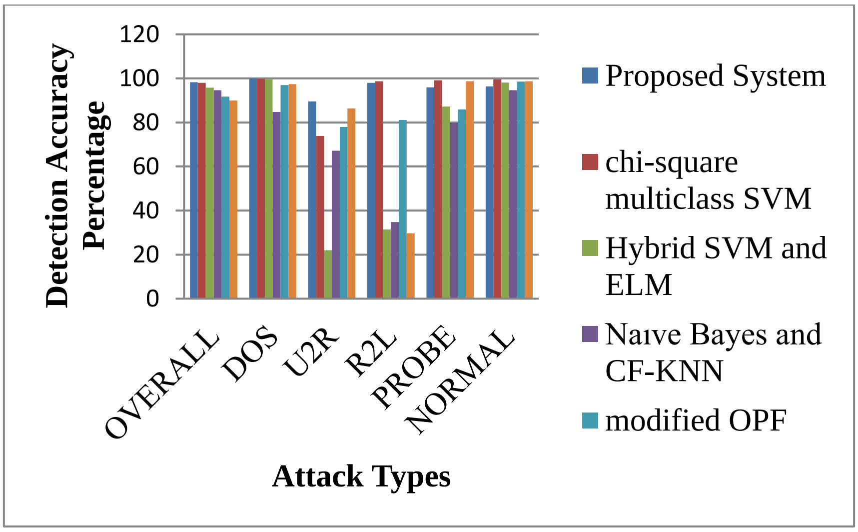

Fig. 2. Performance Comparison of the various IDSs

5. Conclusion

With more and more usage of social media, online transactions and ecommerce, security of data on a network has become quite a challenge. NSSA systems play a crucial role in detecting attacks on a network and taking remedial measures. In order for a NSSA system to perform effectively, the IDS in the system should be capable of detecting various types of attack with high accuracy. To this end, we have proposed an IDS using CRF based classifier. To improve the operational efficiency of the classifier we have also proposed a feature selection method using correlation based subset feature selection algorithm. From the experimentation of the proposed system, it has been shown that the system is capable of detecting various attacks with good accuracy. In future, the system can be tested upon various other datasets to check its efficacy and also steps can be taken to further improve its operational efficiency and accuracy using better feature selection methods.

6. References

[1] United States Department of Homeland Security, “Team Coordination Training, Student Guide”, May 2004.

[2] P. Barford, Y. Chen, A. Goyal, Z. Li, V. Paxson, and V. Yegneswaran, “Employing Honeynets For Network Situational Awareness”, In S. Jajodia et al., (eds.), Cyber Situational Awareness, Advances in Information Security 46, DOI 10.1007/978-1-4419-0140-8.

[3] K. Scarfone, and P. Mell, “Guide to Intrusion Detection and Prevention Systems (IDPS)”, Recommendations of the National Institute of Standards and Technology. [Online].Available: http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf.

[4] Onwubiko C, “Functional requirements of Situational Awareness in Computer Network Security”, In Proc of the IEEE International Conference on Intelligence and Security Informatics, pp. 209-213, June 2009.

[5] M. Qiu , L. Zhang , Z. Ming , Z. Chen , X. Qin , and L. Yang, “Security-aware optimization for ubiquitous computing systems with SEAT graph approach”, J. Comput. Syst. Sci, Vol. 79, no. 5, pp. 518–529, 2013.

[6] E. Hernndez-Pereira , J. Surez-Romero , O. Fontenla-Romero , and A. Alonso-Betanzos, “Conversion methods for symbolic features: a comparison applied to an intrusion detection problem”, Expert Syst. Appl, Vol. 36, no. 7, pp. 10612–10617, 2009.

[7] Q. Yan and F. Yu, “Distributed denial of service attacks in software-defined networking with cloud computing”, IEEE Commun. Mag, Vol. 53, no. 4, pp. 52–59, 2015.

[8] Sumaiya Thaseen. I and Aswani Kumar. C, “Intrusion detection model using fusion of chi-square feature selection and multi class SVM”, Journal of King Saud University – Computer and Information Sciences, 2016. DOI: http://dx.doi.org/10.1016/j.jksuci.2015.12.004.

[9] Rana Aamir Raza Ashfaq, Xi-Zhao Wang, Joshua Zhexue Huang, Haider Abbas, and Yu-Lin He, “Fuzziness based semi-supervised learning approach for intrusion detection system”, Information Sciences, Vol. 378, pp. 484-497, Feb. 2017. DOI: http://dx.doi.org/10.1016/j.ins.2016.04.019.

[10] V.Jyothsna and V.V.Rama Prasad, “FCAAIS: Anomaly based network intrusion detection through feature correlation analysis and association impact scale”, ICT Express, Vol. 2, no. 3, pp. 103–116, 2016.

[11] Wathiq Laftah Al-Yaseen , Zulaiha Ali Othman , and Mohd Zakree Ahmad Nazri, “Multi-Level Hybrid Support Vector Machine and Extreme Learning Machine Based on Modified K-means for Intrusion Detection System”, Expert Systems with Applications, Vol. 67, pp. 296-303, Jan. 2017. DOI: http://dx.doi.org/10.1016/j.eswa.2016.09.041.

[12] Hamid Bostani and Mansour Sheikhan, “Modification of Supervised OPF-based Intrusion Detection Systems using Unsupervised Learning and Social Network Concept”, Pattern Recognition, Vol. 62, pp. 56–72, Feb. 2017. DOI: http://dx.doi.org/10.1016/j.patcog.2016.08.027.

[13] G.R. Kumar, N. Mangathayaru, G. Narsimha, and G.S. Reddy, “A Self Constructing Feature Clustering Approach for Anomaly Detection in IoT”, Future Generation Computer Systems, Vol. 74, pp. 417-429, Sep. 2017. DOI: http://dx.doi.org/10.1016/j.future.2016.12.040.

[14] Jamali. S, and Jafarzadeh. P, “An intelligent intrusion detection system by using hierarchically structured learning automata”, Neural Comput & Applic, Vol. 28, no. 5, pp. 1001–1008, May 2017. DOI: https://doi.org/10.1007/s00521-015-2116-4.

[15] Bostani. H, and Sheikhan. M, “Hybrid of Binary Gravitational Search Algorithm and Mutual Information for Feature Selection in Intrusion Detection Systems”, Soft Comput, Vol. 21, no. 9, pp. 2307–2324, May 2017. DOI: https://doi.org/10.1007/s00500-015-1942-8.

[16] Dash. T, “A Study on Intrusion Detection using Neural Networks Trained with Evolutionary Algorithms”, Soft

Comput, Vol. 21, no. 10, pp. 2687–2700, May 2017. DOI: https://doi.org/10.1007/s00500-015-1967-z.

[17] Kapil Kumar Gupta, Baikunth Nath, and Ramamohanarao Kotagiri, "Layered Approach Using Conditional Random Fields for Intrusion Detection", IEEE Transactions on Dependable and Secure Computing, Vol. 7, no. 1, pp. 35 – 49, Jan-March 2010.

[18] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data", In Proc. of 18th Int'l Conf. Machine Learning (ICML '01), pp. 282-289, 2001.

[19] NSL-KDD Dataset. Retrieved from http://www.unb.ca/cic/research/datasets/nsl.html.

[20] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set", In Proc. 2nd IEEE International Conference on Computational Intelligence for Security and Defense Applications, USA: IEEE Press, pp. 53–58, 2009.

[21] David E. Goldberg, "Genetic algorithms in search, optimization and machine learning", Addison-Wesley, 1989.

[22] R Core Team, "R: A language and environment for statistical computing", R Foundation for Statistical Computing, Vienna, Austria, 2013. URL http://www.R-project.org/

[23] Ling-Yun Wu, "CRF: Conditional Random Fields. R package version 0.3-14", 2017. https://CRAN.R-project.org/package=CRF

[24] Eibe Frank, Mark A. Hall, and Ian H. Witten, "The WEKA Workbench", Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition, 2016