AI Based Supervised Classifiers: an Analysis for Intrusion Detection

Gulshan Kumar
Department of Computer Application
Malout Institute of Management and Information Technology
Malout, Punjab, India
+91-9780435540
gulshanahuja@gmail.com

Krishan Kumar
Department of Computer Science & Engineering
SBS College of Engineering & Technology
Ferozepur, Punjab, India
+91-8054100707
k.saluja@rediffmail.com

ABSTRACT
Researchers investigated Artificial Intelligence (AI) based classifiers for intrusion detection to cope the weaknesses of knowledge based systems. AI based classifiers can be utilized in supervised and unsupervised mode.
Here, we perform a blind set of experiments to compare & evaluate performance of the supervised classifiers by their categories using variety of metrics. The performance of the classifiers is analyzed using subset of benchmarked KDD cup 1999 dataset as training & Test dataset. This work has significant aspect of using variety of performance metrics to evaluate the supervised classifiers because some classifiers are designed to optimize some specific metric. This empirical analysis is not only a comparison of various classifiers to identify best classifier on the whole and best classifiers for individual attack classes, but also reveals guidelines for researchers to apply AI based classifiers to field of intrusion detection and directions for further research in this field.

Categories and Subject Descriptors
I.2 [**Artificial Intelligence**]: I.2.0 [**General**]; C.2 [**Computer-Communication Networks**]: C.2.0 [**General**]: Security and Protection (e.g., firewalls)

General Terms
Security, Performance

Keywords
Artificial Intelligence, classifiers, feature reduction, intrusion, intrusion detection.

1. INTRODUCTION
The online operations become important component of today business scenario [18]. This Internet connectivity and increasing dependence of business applications enables the malicious users to misuse resources and mount variety of attacks. Such large amounts of data need to be managed, monitored and analyzed. This requires development of new strategies to cope with an average load of multiple Gbps. Second, the number of attacks does also continue to grow. The reason behind this is in itself very simple: attacks are getting economically more and more profitable [1].

ACAI '11, July 21 - July 22 2011, Rajpura/Punjab, India
Copyright 2011 ACM 978-1-4503-0635-5/11/10…$10.00.

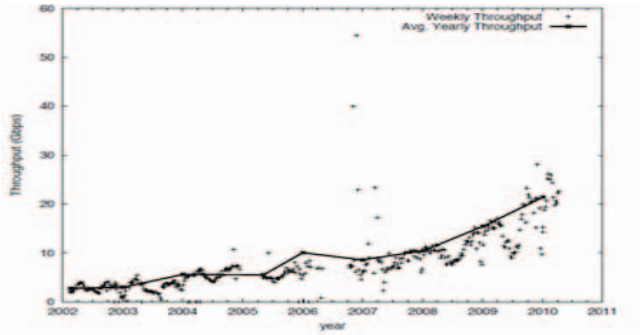

Figure 1. Network throughput (Gbps) for network

DARPA divides the different attacks and intrusions into four different classes viz:
1) Probe; 2) DoS (Denial of Service); 3) U2R (User to Root); 4) R2L (Remote to Local) [15].
Various security oriented technologies have been developed since last decade to prevent ever growing attacks. Now days, Intrusion Detection System (IDS) is a key component in security mechanism for detection of cyber attacks. An intrusion is defined as any set of actions that attempt to compromise the integrity, confidentiality or availability of information resources [4]. An intrusion detection system (IDS) is an effective security technology, which can detect such malicious actions [12]. The main objective of IDS is to detect all possible intrusions in an efficient manner and notify the network security administrator. One major challenge for IDS is to analyze large amount of audit patterns and network traffic. The different features from network connections are recorded in form of network connection records.
These features involve irrelevant and redundant features. Processing and analysis of irrelevant and redundant features leads to
1) Undesirable delay in classification task which in turn loses real time capability of IDS;
2) Increase computation overhead in terms of memory and time; and
3) Deteriorate the classification accuracy [6].
To cope up this problem, various feature reduction techniques are applied to remove irrelevant and redundant features. In this work, we utilized our earlier proposed information theoretic approach for feature selection to reduce the number of features [5]. Due to advantages of Artificial Intelligence (AI) based intrusion detection techniques over other conventional techniques has made it focus of current research in security domain [16]. Witten et al. (2005) divided various AI based supervised classifiers into different categories viz: 1) Rule based; 2) Tree based; 3) Functions; 4) Lazy; 5) Bayes; and 6) Meta [8]. In this work, we compared the performance of various supervised classifiers in different categories based upon defined performance metrics using benchmarked KDD 1999 dataset [11, 11]. Evaluation of the classifiers on variety of metrics is very significant because

different classifiers are designed by keeping in mind to optimize different criteria. For example, SVM are designed to minimize the structural risk and hence optimize the accuracy whereas neural network are designed to minimize empirical risk and hence optimize root mean squared error (RMSE). It is common that one classifier may show optimal performance on one set of metrics and suboptimal on another set of metrics. The identified standard performance metrics involve F-measure (FM), classification rate (CR), false positive rate (FPR), cost per example (CPE), precision (PR), root mean square error (RMSE), area under ROC curve (ROC) and detection rate (DR). We identify best classifier for each attack class in respective classifier category. Further, we compared best classifiers in respective category to identify overall best classifier for different class attacks. Major contributions of this work are:

1) Employment of information theoretic feature selection technique to reduce features of intrusion detection KDD 1999 dataset.

2) Empirical comparison of various classifiers in different classifier categories to identify best classifier in respective category for different attack classes using reduced test KDD 1999 dataset in terms of defined performance metrics.

3) Empirical comparison of various best classifiers of different classifier categories to identify overall best classification technique for different attack classes using reduced test KDD 1999 dataset in terms of defined performance metrics.

To that end, we analyze the performance of supervised classifiers and evaluate their performance for intrusion detection using subsets of benchmarked KDD cup 1999 dataset as training & test dataset. To the best of our knowledge, this empirical work is most comprehensive analysis in terms of number of classifiers considered and number of metrics used for comparison. Rest of paper is organized as follows. Section 2 describes the experimental setup and methodology for conducting experiments. This includes preparation of evaluation dataset, employment of feature reduction, definition of performance metrics and AI based classifier categories. Section 3 gives the empirical study of various classifiers in terms of five defined metrics. Results of best classifiers are compared and analyzed in terms of defined metrics in section 4. Finally, the conclusive remarks and future research guidelines are highlighted in section 5.

2. EXPERIMENTAL SETUP & METHODOLOGY

This section describes evaluation dataset, preprocessing strategy, selection of training and testing dataset, formation of reduced training and testing dataset by employing of feature selection approach, various classifier categories and other experimental setup.

2.1 Methodology

We performed experiments on Intel PIII 239 MHz with 1GB RAM with Windows XP operating system. We performed 5-class classification of dataset using well known open source publicly available machine learning tool called WEKA [19] to classify KDD cup 1999 dataset. We conducted set of experiments using default parameters of Weka implemented classifiers. The stages of experiment and their interaction is described as follows and depicted in figure 2.

1) Preprocessing stage: In this stage, conversion of symbolic features to numeric features and normalization of features is performed for Training and Test KDD dataset as described in [7].

2) Feature reduction stage: In this stage, an information theoretic feature selection approach [8] is applied to normalized Training and Test Dataset for generating reduced feature set. Hence, reduced training and test dataset with reduced features.

3) Classification stage: classification stage involves two phases namely training phase and testing phase. 3.1) Training Phase: Here, classifier is learnt using reduced training dataset. The output of this phase is trained model which optimized using 10 cross validation.3.2) Testing Phase: Here, trained model is given input of Test dataset to predict the class label.

4) Performance metrics computation: After testing phase, performance metric computation stage computes the defined performance metrics. The performance metrics divided into three classes: threshold, ranking and probability metrics [17]. Threshold metrics include classification rate (CR), F-measure (FM) and Cost per example (CPE). It is not important how close a prediction is to a threshold, only if it is above or below threshold. The value of threshold metrics lies in [0, 1]. Ranking metrics include false positive rate (FPR), Detection rate (DR), precision (PR) and area under ROC curve (ROC). The value of ranking metrics lies in [0, 1]. These metrics depend on the ordering of the cases, not the actual predicted values. As long as ordering is preserved, it makes no difference. These metrics measure how well the attack instances are ordered before normal instances and can be viewed as a summary of model performance across all possible thresholds. Probability metrics include root mean square error (RMSE). Value of RMSE lies between 0 and 1. The metric is minimized when the predicted value for each attack class coincides with the true conditional probability of that class being normal class. These metrics are computed from confusion matrix. The matrix gives the values of True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN). Detail of these metrics can be further elaborated in [10].

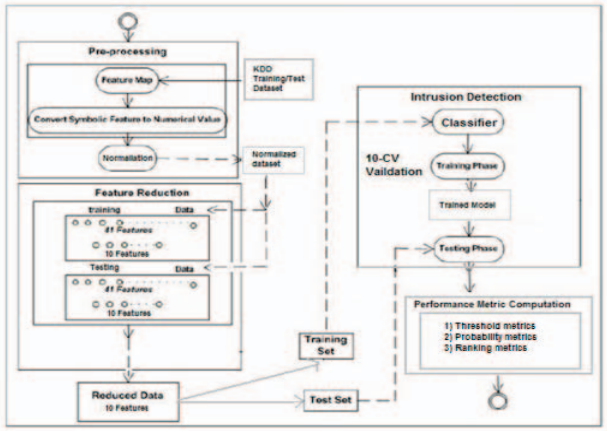

Figure 2. Experiment methodology.

2.2 Intrusion Detection Dataset

We used well known intrusion detection benchmarked KDD cup99 dataset to compare and analyze different classification techniques. Although KDD99 data-set might have been criticized for its potential problems [14, 10], but many researchers give the priority to KDD dataset over other publicly available dataset as benchmark dataset for evaluation of IDS [7, 3]. KDD dataset utilizes TCP/IP level information and embedded with domain-specific heuristics, to detect intrusions at the network level. It contains four major classes of attacks: Probe, Denial of Service (DoS), User-to-Root (U2R) and Remote-to-Local (R2L) attacks. There are 4,898,430 labeled and unlabeled and 3, 11,029 unlabeled connection records in the dataset. The labeled connection records consist of 41 features and 01 attack type. The number of connection records in training set and test dataset is very large and non-uniformly distributed for different attack classes. In order to perform unbiased normalized training and testing of classifier model, we randomly choose the connection records as suggested by Kumar et al. (2010) [7]. There are 66,961 (including normal) connection records selected from entire labeled KDD dataset for training of classifiers. There are 40,603 connection records in the test dataset. KDD dataset contains symbolic as well as continuous features. The dataset is pre-processed to make it compatible before it is used for training and

testing the classifiers. The pre-processing steps involve mapping of symbolic value features to numeric value and normalizing of feature values as suggested by Kumar et al. (2010)[7].

2.3 Feature Reduction

The KDD dataset contains some irrelevant and redundant features. Processing of these irrelevant and redundant features leads to many problems including 1) Undesirable delay in classification task which in turn loses real time capability of IDS; 2) Increase computation overhead in terms of memory and time; and 3) Deteriorate the classification and prediction accuracy. To solve this problem, we employed an information theoretic approach for feature selection suggested by Kumar et al.(2010) [8]. This feature selection approach is a filter approach and independent of any classification technique. Thus, KDD dataset is reduced by using the approach. The reduced KDD dataset is not dependent upon any classification technique. Here, we utilized mutual information to compute relevance of features to predict the class labels. The reduced KDD dataset contains 10 features of 66,961 instances in training dataset and 10 features of 40,603 instances in test dataset. Details of number of instances in training and test dataset for various attack classes are as described in table 1.

Table 2. Statistics of subsets of KDD cup 1999 dataset as Training and Test dataset

	Attack class	#instances		Attack class	#instances
Training dataset	Normal	10,000		Normal	5,000
	Probe	32,316		Probe	4,166
	DoS	23,467	Test dataset	DoS	17,761
	U2R	52		U2R	228
	R2L	1,126		R2L	13,448
Total		66,961	Total		40,603

3. EMPIRICAL STUDY

This section presents the comparative performance evaluation of various classifier categories in terms of identified eight standard metrics.

3.1 Tree Based Classifiers

We compared: 1) Random Forest; 2) Random Tree; 3) NB Tree; 4) J48; 5) Simple CART classifiers. The comparative results are as shown in table 2.

Table 2. Comparative results of tree based classifiers

Tree based techniques / Metrics	Random forest	Random tree	NB tree	J48	Simple CART
CR	0.542	0.599	0.51	0.64	0.635
CPE	0.929	0.802	0.883	0.716	0.718
RMSE	0.347	0.401	0.423	0.373	0.379
Avg DR	0.542	0.599	0.51	0.648	0.635

It is observed that J48 outperformed the other classifiers in terms of threshold metrics. In terms of probability metrics, random forest classifier outperformed the others and J48 is at the second position. For ordering metrics like DR, FPR and PR, J48 show better performance than other classifiers. But in terms of ROC, random forest classifier outperformed J48 by putting it second positions. Last row indicates that on the whole J48 classifier proved superior performance than others classifiers in the category of tree based classifiers.

3.2 Rule Based Classifiers

Under this classifier category, we utilized: 1) JRIP; 2) Decision Table classifiers and depicted the results in table 3.

Table 3. Comparative results of rule based classifiers

Rule based classifiers	CR	CPE	RMS error	Avg DR	Avg FPR	Avg ROC	Avg FM	Avg PR
JRip	0.556	1.297	0.42	0.556	0.112	0.702	0.55	0.74
Decision Tree	0.362	1.104	0.41	0.362	0.094	0.769	0.35	0.75

It is observed that JRip performed superior performance than decision tree in terms threshold metrics but worse performance than in terms of probability metrics and FPR, ROC, PR ranking metrics.

3.3 Bayes Classifiers

Under this category, we utilized: 1) Bayesnet; and 2) Naïve Bayes classifiers and shown results in table 4.

Table 4. Comparative results of Bayes classifiers

Bayes	CR	CPE	RMS error	Avg DR	Avg FPR	Avg ROC	Avg FM	Avg PR
Naïve bayes	0.566	0.867	0.41	0.566	0.233	0.746	0.56	0.65
Bayesnet	0.511	0.997	0.41	0.511	0.185	0.791	0.49	0.6

It is observable that Naïve bayes has shown superior performance than bayesnet in terms of threshold metrics and DR and PR based ranking metrics.

3.4 Functions Classifiers

Under this category, we utilized: 1) MLP; 2) SMO; 3) RBF Network; and 4) LibSVM classifier [2] and results are shown in table 5.

Table 5. Comparative results of function based classifiers

Functions	CR	CPE	RMS error	Avg DR	Avg FPR	Avg ROC	Avg FM	Avg PR
SMO	0.566	0.887	0.377	0.566	0.205	0.613	0.552	0.681
MLP	0.58	0.868	0.38	0.58	0.202	0.794	0.566	0.673
RBF Network	0.459	0.97	0.405	0.459	0.239	0.764	0.455	0.583
LibSVM	0.578	0.887	0.411	0.578	0.204	0.687	0.565	0.693

In case of LibSVM, we used polynomial type kernel. It is observed from results MLP proved superior performance than other classifiers in this category in terms of threshold metrics and ranking metrics. MLP show comparable performance with SVM and superior than RBF Network.

3.5 Lazy Classifiers

Under this category, we utilized: 1) IB1; 2) IBK; and 3) Kstar classifiers and depicted the results in table 6.

Table 6. Comparative results of Lazy classifiers

Lazy	CR	CPE	RMS error	Avg DR	Avg FPR	Avg ROC	Avg FM	Avg PR
LB1	0.584	0.833	0.41	0.584	0.1	0.742	0.59	0.75
LBk	0.587	0.829	0.41	0.587	0.1	0.818	0.59	0.75
Ksatr	0.521	0.954	0.42	0.521	0.256	0.589	0.47	0.62

It is observed from results that LBk proved better performance than other classifiers in this category in terms of threshold metrics, probability metrics and ranking metrics except PR.

3.6 Meta Classifiers

Under this category, we utilized: 1) Bagging; 2) Boosting; and 3) Random sub space using Decision Tree based J48 classifier and shown the results in table 7.

Table 7. Comparative results of Meta classifiers

Meta	CR	CPE	RMS error	Avg DR	Avg FPR	Avg ROC	Avg FM	Avg PR
Boosted tree-J48	0.592	0.815	0.37	0.592	0.056	0.822	0.61	0.82
Bagged tree-J48	0.661	0.694	0.35	0.661	0.047	0.861	0.7	0.83
Random subspace-J48	0.525	0.833	0.39	0.515	0.064	0.785	0.55	0.83

It is observed from results that bagged tree-J48 proved better performance than other classifiers in this category in terms of threshold metrics, probability metrics and ranking metrics.

4. RESULT ANALYSIS

Best classifiers in various categories further can be compared based upon identified metrics as shown in table 8.

Table 8. Category wise comparative results of best classifiers.

Category		Tree based	Rule based	Bayes	Functions	Lazy	Meta
Classifier		J48	JRip	Naïve bayes	MLP	LBk	Bagged Tree-J48
Threshold	CR	0.6485	0.5555	0.5656	0.5797	0.5868	0.6609
	Avg FM	0.684	0.547	0.561	0.566	0.589	0.697
	CPE	0.716	1.2974	0.8671	0.8682	0.8287	0.6941
Prob.	RMSE	0.373	0.4209	0.4136	0.3795	0.4064	0.3498
Ranking	Avg DR	0.648	0.556	0.566	0.58	0.587	0.661
	Avg FPR	0.049	0.112	0.233	0.202	0.1	0.047
	Avg ROC	0.792	0.702	0.746	0.794	0.818	0.861
	Avg PR	0.825	0.742	0.645	0.673	0.75	0.827

Bagged Tree-J48 classifier outperformed all other classifiers by reporting optimal values of various metrics. J48 classifiers performed better than other classifiers after bagged J48 in terms of threshold metrics, probability metrics, DR. Table 4.2 depicts comparison of various best classifiers for different attack class category wise.

For probe attack class, Jrip classifier outperformed in terms of ROC and FM with DR of 89.9% at 1.9% FPR. Bagged tree-J48 classifiers detections probe attacks at a rate of 95.5% at 9.5% FPR. For DoS attack class, bagged tree-J48 outperformed other classifiers with highest values of DR, ROC, FM, PR and lowest FPR. J48 tree based classifier is at second position after bagged tree-J48 for this attack class. For U2R attack class, Jrip rule based classifier outperformed other classifiers having maximum value for ROC, FM with DR of 18.4 % at 0.1% FPR. For R2L attack class, Naïve Bayes classifier outperformed others in terms of ROC but shows less DR of 32.6 % at high value of 3.9 %FPR. Bagged Tree-J48 proved highest DR of 49.1 % at 2.8% FPR.

5. CONCLUDING REMARKS AND FUTURE WORK

In this empirical work, we performed a set of blind experiments of supervised classifiers on benchmarked KDD cup 1999 dataset. Main objective of this work is to analyze common supervised classifiers used in literature for intrusion detection. From empirical results of our experiments, it is concluded that bagged tree-J48 classifier is best and stable classifier for organizations concerned with overall correct classification of malicious traffic with minimum cost per example, FPR and maximum ROC. Further, it is also suggested that rule based JRip & Bagged Tree-J48 for probe, Bagged tree-J48 for DoS, JRip for U2R and Naïve bayes, bagged tree-J48 and neural network based MLP for R2L attack class can be utilized for detection of various attack classes. Although the supervised classifiers are used directly for intrusion detection, our results prove that a single classifier cannot detect all the attack classes efficiently. Accordingly, a set of classifiers might be used to detect attack belonging to different classes. One more point can be highlighted in terms of detecting attack classes. First, all supervised classifiers reported poor results for detection U2R and R2L attack classes. Similar results are also reported by [13]. This might possibly be due to insufficient number of training instances in comparison to other attack classes. We need more analysis to explain these poor results.

Table 9.Attack class wise comparative results of best classifiers category wise.

Attack class	Category	Classifier	DR	FPR	ROC	FM	PR
Probe	Tree based	J48	0.96	0.111	0.89	0.651	0.5
	Rule based	JRip	0.9	0.019	0.986	0.87	0.84
	Bayes	Naïve bayes	0.72	0.008	0.907	0.802	0.91
	Functions	MLP	0.82	0.038	0.912	0.76	0.71
	Lazy	LBk	0.96	0.203	0.941	0.513	0.35
	Meta	Bagged Tree-J48	0.96	0.095	0.976	0.685	0.53
DoS	Tree based	J48	0.67	0.021	0.812	0.792	0.96
	Rule based	JRip	0.66	0.128	0.754	0.726	0.8
	Bayes	Naïve bayes	0.66	0.462	0.547	0.584	0.53
	Functions	MLP	0.65	0.391	0.766	0.605	0.57
	Lazy	LBk	0.63	0.124	0.798	0.705	0.8
	Meta	Bagged Tree-J48	0.67	0.017	0.851	0.793	0.97
U2R	Tree based	J48	0.05	0.08	0.43	0.01	0
	Rule based	JRip	0.18	0.001	0.674	0.262	0.45
	Bayes	Naïve bayes	0.17	0.022	0.445	0.068	0.04
	Functions	MLP	0.1	0	0.663	0.171	0.56
	Lazy	LBk	0.06	0.002	0.639	0.092	0.18
	Meta	Bagged Tree-J48	0.05	0.08	0.523	0.007	0
R2L	Tree based	J48	0.46	0.022	0.722	0.607	0.91
	Rule based	JRip	0.16	0.018	0.514	0.269	0.82
	Bayes	Naïve bayes	0.33	0.039	0.933	0.465	0.81
	Functions	MLP	0.31	0.016	0.786	0.467	0.9
	Lazy	LBk	0.33	0.009	0.778	0.484	0.95
	Meta	Bagged Tree-J48	0.49	0.028	0.844	0.635	0.9

These experiments and their results provide reliable guidelines for future research in applying supervised classifiers for field of intrusion detection and expose some new avenues of research.

Further research is required to explore more supervised classifiers for intrusion detection. This fact is supported in our experiments that overall classification rate of 66.09 % for best classifier in

different categories. Furthermore, the overall performance of supervised classifiers might be enhanced by combining the strengths of individual classifiers. These hybrid models may be supported with voting type strategies that keep strengths and weakness of individual classifiers into consideration.

For improving the performance of classifiers for detection of U2R and R2L attack classes, more informative features can explored in future.

6. REFERENCES

[1] A. Sperotto, Flow-Based Intrusion Detection, CTIT Ph.D.-thesis Series No. 10-180, Centre for Telematics and Information Technology, University of Twente, 2010.

[2] C. C. Chang, C. J. Lin, LIBSVM: a library for support vector machines, Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[3] C. F. Tsai, Y. F. Hsu, C. Y. Lin and W. Y. Lin, Intrusion detection by machine learning: A review, Expert Systems with Applications, Vol 36, Issue 10, pp. 11994-12000, December 2009.

[4] E. Hernandez-Pereira, J. A. Suarez-Romero, O. Fontenla-Romero, A. Alonso-Betanzos, Conversion methods for symbolic features: A comparison applied to an intrusion detection problem, Expert Systems with Applications, 36 (2009) 10612–10617.

[5] G. Kumar, K. Kumar, An information theoretic approach for feature selection, Security and communication networks, Wiley Blackwel publisher- Accepted.

[6] G. Kumar, K. Kumar, M. Sachdeva, An Empirical Comparative Analysis of Feature Reduction Methods For Intrusion Detection, International Journal of Information and Telecommunication, 1 (2010) 44-51, ISSN: 0976-5972.

[7] G. Kumar, K. Kumar and M. Sachdeva, The Use of Artificial Intelligence based Techniques For Intrusion Detection – A Review, Artificial Intelligence Review, vol. 34, No. 4, pp. 369-387, Springer, Netherlands, DOI: 10.1007/s10462-010-9179-5 ISSN: 0269-2821.

[8] I. H. Witten, E. Frank, Data Mining-Practical machine learning tools and techniques, Second Edition, Morgan Kaufmann, An imprint of Elsevier, ISBN 0-12-088407-0, 2005.

[9] Internet 2. Internet 2 research network. Available: http://www.internet2.edu/, Sept. 2010.

[10] J. McHugh, Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory, ACM Transactions on Information and System Security, 3- 4, 262–294, 2000.

[11] KDD99, KDD cup 1999 data. (1999). Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[12] L. R. Halme, R. K. Bauer, AINT Misbehaving: A Taxonomy of Anti-Intrusion Techniques, Proceedings of the 18th National Information Systems Security Conference. Baltimore, MD 1995.

[13] M. Sabhnani, G. Serpen, Application of machine learning algorithms to KDD intrusion detection dataset within misuse detection context, In International conference on machine learning: Models, technologies and applications, New York, 623–630, 2003.

[14] M. Tavallaee, E. Bagheri, W. Lu, A. A. Ghorbani, A Detailed Analysis of the KDD CUP 99 Data Set, In proceedings of IEEE symposium on computational intelligence in security and defense applications (CISDA), 2009.

[15] MIT Lincolon Laboratory, 1999 DARPA intrusion detection evaluation design and procedure, DARPA Technical report, Feb 2001.

[16] Ponce Mario Castro, Intrusion Detection System with Artificial Intelligence, FIST Conference - June 2004 edition-1/28 Universidad Pontificia Comillas de Madrid, 2004.

[17] R. Caruana, A. Niculescu-Mizil, Data mining in metric space: an empirical analysis of supervised learning performance criteria, KDD '04 Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, 2004, ISBN:1-58113-888-1.

[18] T. Shon, J. Moon, A hybrid machine learning approach to network anomaly detection, Information Sciences, 177 (2007) 3799-3821.

[19] Weka, An open source data mining software tool developed at university of Waikato, New Zealand, Available: http://www.cs.waikato.ac.nz/ml/weka.