Attack Detection Based on Data Mining Techniques

Dr. Buthynna Fahran[1], Dr. Mohammed Najm[2], Mustafa Abdulsamea Abdulhamed[3]

[1]Assistant Professor, Informatics Institute for Postgraduate Studies, Iraq

[2]Assistant Professor, Instructor at Informatics Institute for Postgraduate Studies, Iraq
Working at Department of Computed Engineering, University of Technology, Iraq

[3]Informatics Institute for Postgraduate Studies, Iraq

Abstract: ***In light of the security challenges posed by the reality of today, where the Internet and exchange information are an integral part of our daily lives, we live in a world where data requirements have become dynamic, where things are permanently changing. In order to provide security and decrease the damage of information system caused by attacks on the network; it is important to provide it with Intrusion Detection system (IDS). In this paper, we present intrusion detection model based on Feature extraction and two-stage classifier module, designed to detect anomaly activities. The proposed model using Principal Component Analysis (PCA) of Feature extraction to map the high dimensional dataset to a lower one with effective features. We then apply a two-stage classification module utilizing Naïve Bayes and C4.5 to identify abnormal behaviors. The experiment results using NSL-KDD dataset shows that Our model outperforms the previous model for detection low-frequency attacks.***

Keywords: intrusion detection system, multi-stage classification, anomaly detection, NSL-KDD.

1. Introduction

The study of Central Intelligence Agency (C.I.A). World Factbook[1], showed that approximate to two billion users in the world (29.6% of the world population) are accessing the Internet, and about six billion users use cell phones (84% of the Estimated world population). This made accessing Internet a necessary part of everyday life, and at the same time led to Security issues. Intrusion detection systems (IDS) are an essential tool used for securing computer infrastructure. an IDS screens movement and looks to recognize evidence of ongoing attacks, infiltration attempts, or security policy violations [2]. IDSs have developed since the main model proposed in the late 1980s [3]. IDS methods can be spliced into rule-based detection and anomaly detection, in rule-based detection, compare the monitored events with the previous saved knowledge from known attacks and malicious, while in anomaly detection compare monitored events with a predefined model of normality to detect attacks [4]. Data mining (DM) is the process of extracting relevant knowledge from a large database, IDS is a data analysis process where DM techniques are used to automatically learn and detect normal and malicious patterns. DM usually comprise of four categories of the task. Clustering, Classification, Regression and Association rule learning [5]. Classification is the process of taking each instance in the dataset and recognize the class it is belonging to, meaning that the known structure will be used in the new cases [6]. for evaluating the performance of the proposed system, NSL-KDD Datasets which described specifically in section 3.1, have been used for training and testing stages, the malicious activities are divided into four groups [7]:

Denial of Service Attack (DoS): when the attacker tries to prevent a legitimate user from using service.

User to Root Attack (U2R): when the attacker has local access to the target machine and tries to gain root access to the system.

Remote to Local Attack (R2L): when the attackers try to gain remote access to the victim machine.

Probing Attack: when the attacker tries to gather information about target host.

1.1 Preprocessing

Preprocessing is one of the most important steps in data mining techniques the data are transformed or consolidated so that the resulting mining process may be more efficient, and the patterns found may be easier to understand [8].

1.2 Principal Component Analysis

Principal Component Analysis (PCA) (an unsupervised dimension reduction technique) In order to address the issue of high dimensionality. PCA can be used to perform feature selection and extraction [9]:

a) Feature selection: pick a subset of all features depend on their effectiveness in higher classification (i.e. picking more useful features).
b) Feature extraction: make a subset of new features by mixing existing features.

1.3 Naïve Bayes

Naïve Bayes Classifier (NB) is a supervised machine learning algorithm an statistical method for classification [10]. NB is an efficient and effective widely used classification algorithm, it possesses several properties that make it surprisingly useful and accurate. NB is a simple probabilistic classifier which depends on applying Bayes theorem with strong independence assumption. Depending on the precise nature of the probability model, NB can be trained very efficiently in a supervised learning setting [11].

Licensed Under Creative Commons Attribution CC BY

1.4 C4.5

C4.5 is a well – known algorithm used to generate a decision tree. This algorithm was proposed in 1993 by Ross Quinlan [12] to overcome the limitations of the ID3 algorithm.

The C4.5 decision tree used for classification and also referred to as a statistical classifier. The C4.5 algorithm made a number of changes to improve ID3 algorithm [13] some of these are:

1) A possibility to use continuous and discrete data.
2) Handling different weights attribute.
3) Handling training data with unknown (missing) value of attributes.
4) Pruning the decision tree after being created:
   a) Pessimistic prediction error.
   b) Sub-tree raising.

2. The Proposed Model and Methodology

The diagram of the proposed model is illustrated in Figure 1 as shown below.

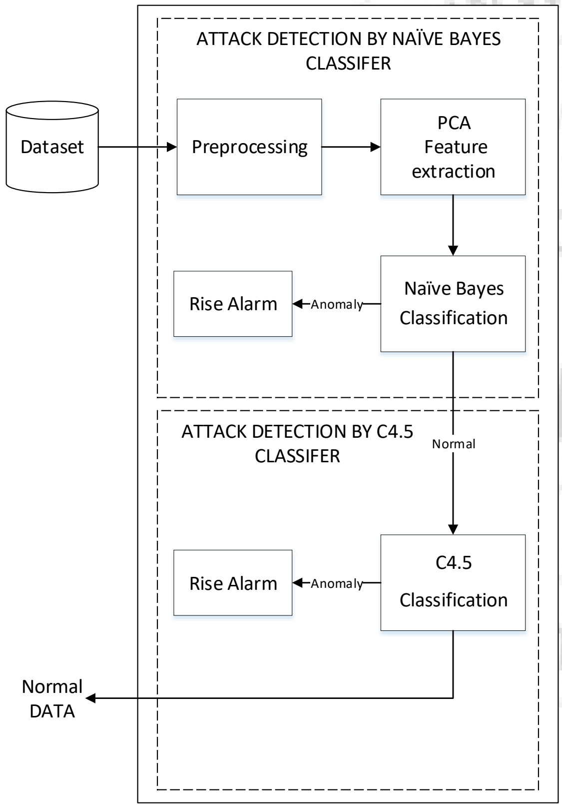

Figure 1: The proposed model

2.1 Preprocessing stage

In this stage the original dataset is mapped into a normal form as follows:

- Each nominal feature value will be specified with a unique integer number.
- Continuous-valued features will be mapped into an integer number, to avoid any bias, as show in (1) for each continuous valued z. continuous-valued feature is normalized using logarithm to base 2 and then casting the result into an integer value.

$$if(z \geq 2) z = \int(\log_2(z) + 1) \qquad (1)$$

2.2 Feature extraction stage

In this stage Principal Component Analysis is used as a feature extraction mechanism to map the NSL-KDD dataset, which consists of 41 features into the lower one by removing the less significant features. Figure 2 shows the feature extraction technique is commonly limited to linear transforms: y=Wx.

$$\begin{bmatrix}x_1\\x_2\\\vdots\\x_n\end{bmatrix} \xrightarrow{linear feature extraction} \begin{bmatrix}y_1\\y_2\\\vdots\\y_m\end{bmatrix} = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1n}\\ w_{21} & w_{22} & \cdots & w_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ w_{m1} & w_{m2} & \cdots & w_{mn}\end{pmatrix}\begin{bmatrix}x_1\\x_2\\\vdots\\x_n\end{bmatrix}$$

Figure 2: Principal Component Analysis linear transformation

Let X be an N-dimensional random vector in the original dataset, and the new feature space consists of lower M-dimensions (M is the number of new dataset features that are transformed) where (M<N). for the operation of transformation, we will need to calculate (2) to (4):

Covariance matrix:

$$\sum_x = \sum_{k=1}^{n}(x_k - \overline{m})(x_k - \overline{m})^T \qquad (2)$$

Where $\overline{m}$ (mean vector) is:

$$\overline{m} = \frac{1}{n}\sum_{k=1}^{n} x_k \qquad (3)$$

Eigenvector – eigenvalue decomposition:

$$\sum v = \lambda v \text{ Where v=Eigenvector } \lambda\text{=Eigenvalue} \qquad (4)$$

The PCA will then sort the eigenvectors in descending order. in which, eigenvectors with lower eigenvalue have the low information about the distribution of data and these are the eigenvectors we want to drop. A common approach is to rank the eigenvector from the highest to the lowest eigenvalue and choose the top K eigenvectors based on eigenvalues. Similarly, in our proposed model, one may decide which eigenvalues are more useful; the new obtained feature apace has 12 dimensions called {PCA1, PCA2, …, PCA12} instead of 41 dimensions.

2.3 Naïve Bayes Classifier stage

In this stage, Naïve Bayes classifier (NB) is used as a first stage classifier. NB has two types of variables: the class variable C and a set of features X = {X1; X2; …; Xn}, on a dataset D Which consists of {E1, E2, …, Et} instances and can be defined as in (5), then with the consideration of the Naïve independence assumption of the attributes given the class as in (6)[14].

$$c(E) = argmax_{c\in C} P(c) \times P(a_1, a_2, \dots, a_n|c) \qquad (5)$$

$$P(E|c) = P(a_1, a_2, \dots, a_n|c) = \prod_{i=1}^{n} P(a_i|c) \qquad (6)$$

The conditional independence assumption leads to posterior probabilities, NB classifier is constructed easily because of the simplicity of computing P(C) and P$(a_i|c)$[15]. After this stage of classification, for more purity in detection the output which classified as normal behavior and which not correctly classified will be chosen again by using C4.5 classifier to classify them.

Licensed Under Creative Commons Attribution CC BY

2.4 C4.5 Classifier stage

Because most of the low frequency and dangerous malicious behavior had completely overlap with normal behavior ones in the distinguished dataset. That is why most of classifiers like Naïve Bayes make a wrong decision to gain good separation boundary between these classes. To obtain better separation between anomalous and normal objects the outputs of last classifier which are labeled as normal or un labeled will be considered as suspected input to C4.5 classifier.

At each node of the tree, C4.5 pick one attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. C4.5 compute the normalized information gain for chosen attribute and pick the attribute with highest normalization information gain to make decision, the C4.5 algorithm then continues with the same steps on the smaller sub-lists having next highest normalization information gain [16]. To build C4.5 decision tree we need to compute (7) and (8):

For class label of train dataset compute Entropy;

$$Entropy(p) = -\sum_{i=1}^{n} p_i \times \log_2(p_i) \qquad (7)$$

Where Pi is a probability distribution.
For each attribute (T) compute information gain;

$$informationgain = Entropy(p) - \sum_{j=1}^{n}(p_j \times Entropy\ (p_j)) \qquad (8)$$

Where values of Pj is the set of all possible values for attribute (T).

3. Implementation

In this section we will first discuss a detailed description of the applied data set, then the IDS performance indicator will be determined and lastly evaluate the proposed model will be argued.

3.1 NSL-KDD Dataset

The benchmark dataset NSL-KDD is used to implement the proposed system. NSL-KDD [17] dataset is a reduced version of the original KDD 99 (KDD Cup 1999) [18]dataset this dataset introduce for NIDS (network intrusion detection systems) competition. NSL-KDD Records consists of a host-to-host connection which has 41 features (e.g., protocol type, service, flag … etc.) plus one class attribute the same features as KDD 99. The class attribute has four types of attacks as Table (1) presented: Probe attacks, User to Root (U2R) attacks, Remote to Local (R2L) attacks and Denial of Service (DoS) attacks [19]. The feature vector consists of three categorical values; five symbolic values and the rest of them are a continuous value.

Table 1: Classification of attacks in NSL-KDD dataset

Main class	Attacks type
DoS	back, land, Neptune, pod, smurf, teardrop.
Probe	ftp write, guess passwd, IMAP, multhop, phf, spy, warezclient, warezmaster.
U2R	buffer overflow, perl, loadmodule, rootkit.
R2L	ipsweep, nmap, portsweep, satan.

3.2 Performance indicator

Generally, the performance of the IDS can be evaluated using four major criteria that are [15] :

TP (true positive)	number of attack events correctly classified as an attack.
FN (false negative)	Numbers of attack events where are incorrectly classified as normal.
FP (false positive)	Numbers of normal events where are incorrectly classified as attacks.
TN (true negative)	number of normal events correctly classified as normal.

The detection Rate (DR): is a measure of the classifier correctly detection malicious samples of all malicious objects, it's computed as (9):

$$\text{DR} = \frac{TP}{FN+TP} \qquad (9)$$

False Alarm Rate (FAR): is a measure of the classifier wrongly detecting benign samples as malicious of all benign objects, it computes as (10):

$$\text{FAR} = \frac{FP}{FP+TN} \qquad (10)$$

The confusion matrix is a quality measurement of the classifier that shows the number of correct and incorrect predictions made by the classification system compared to the actual outcomes in the data. The matrix is NxN, where N is the number of classes. Table (2) shows the confusion matrix for a two-class classifier [20].

Table 2: Confusion matrix for two classes

Actual Class	Predicted class	
	Negative	Positive
Negative	TP	FN
positive	FP	TN

3.3 Testing environment and results

The experiment was processed within a Microsoft Visual Studio Enterprise Version 4.7.02556 | 2017, which was running on a PC powered by Intel® Core™ i7 CPU M620 @ 2.67GHz 2.67GHz 64-bit operating system and 8 GB RAM.
The proposed model was trained by training database and then evaluated by dedicates test database provided by NSL-KDD. So all the given results in this study are evaluated by this test database. After normalization step for test database, the projection matrix (W) which obtain from training test applied on a test database. Another important issue which appears from the NSL-KDD Dataset is the rare and dangerous attack like R2L are so involved with normal behaviors. But our proposed model can nearly solve this issue by using C4.5 as a second classifier. Figure.3, shows the detection rate of our model for different PCA reduced, at this step 41 iteration experimented. According to a detection rate of this experiment 12 PCA dimensions nominated to applied in the proposed model because of obtaining better detection rate on low-frequency attacks in comparison with the other nominated number. To show the usefulness of the proposed model concept for using two stage of classification, Table 3 shows the detection rate of the first stage which belongs to attack instances is compared to the final decision of the second stage.

Licensed Under Creative Commons Attribution CC BY

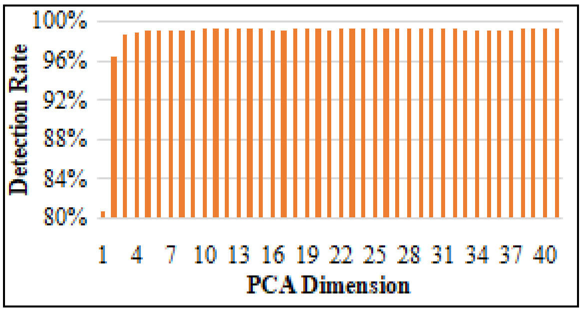

Figure 3: Detection rate experiment over different PCA Dimension Reduction by NSL-KDD

Table3: Comparison between detection rate (%) of the first and refined stage of classification

Level	Probe	DoS	U2R	R2L
First level of classification	71.51	91.71	11.76	30.86
Refined level of classification	99.39	99.88	81.17	91.15

The comparison results in Table 4 and Figure 4 shows that the proposed model gained better detection rate in normal and the low-frequency attacks (U2R, R2L) and also close detection rates to other types of attacks against one of the recent works. In comparison to the two classification models, the proposed model also obtained a desirable result. It should be noted that this model is proposed to address with the lack of other models present in the detection of low-frequency class attacks that which is located in the data set and also obtain promising detection rates of the other types of attack In addition, the model should be compared with multi-layered classifications such as [21] which provided a solution to the same problem, As can it seen in Table 4. The proposed model has exceeded the U2R detection rate by threefold as much, and the same as in the R2L attacks. Let's take a look at other existing models that had an impressive low false alarm and their detection rate against low frequency attacks Table5. In this study two-class (normal or anomaly) the classification problem of anomalies, each object on arrival which gave one of the attack label called anomalies and other so-called normal behavior. Table 6 also presents a comparison between the one-level approach and the proposed model that has exploited two classifiers. As demonstrated the two-stage model outperformed the other models in detection and false alarm rates.

Table 4: Multi-stage classification Detection Rates (%) comparison to existing models.

Method	Normal	Probe	DoS	U2R	R2L
Proposed model	99.43	99.39	99.88	81.17	91.15
HFR-MLR method [21]	93.70	80.2	89.70	29.50	34.20

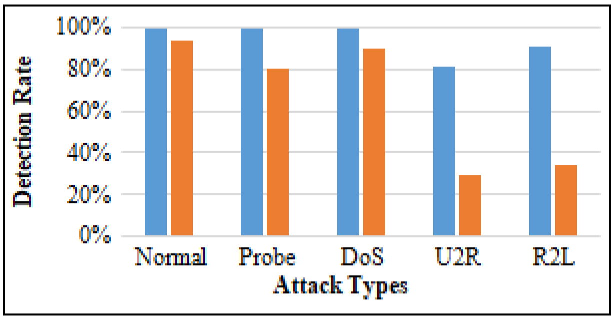

Figure 4: The performances of the proposed model and HFR-MLR method

Table 5: Confusion matrix of existing models which had a low false alarm and undesirable detection rate (%) against the low-frequency attacks versus proposed model

Method	Normal	Probe	DoS	U2R	R2L
Proposed model	99.43	99.39	99.88	81.17	91.15
Association rule [22]	99.5	96.8	74.9	0.79	0.38
SVM with BIRCH clustering [23]	99.0	99.5	97.5	28.8	19.7
ESC-IDS [24]	98.2	99.5	84.1	31.5	14.1

Table 6: Single-layer and multi-layer classification comparison (%) result

Method	Detection Rate	False alarm Rate
Proposed model	99.351	0.002
Naïve Bayes [17]	76.56	N/A
Random forest [17]	80.67	N/A
SVM [17]	69.52	N/A
Decision trees [17]	81.05	N/A
SOM IDS [25]	75.49	N/A
Feature selection with SVM IDS [26]	82	15
Fuzzy classification by Evolutionary algorithms [27]	82.74	3.92

4. Conclusion

This paper is proposed a network anomaly detection model which used a data preprocessing, PCA feature extraction model and also two-stage classifier. The proposed model works with only 12 mapped feature out of 41 distinguished attributes of NSL-KDD database. Applying two stage of classification by Naïve Bayes and C4.5 which drive to earn higher detection rate on the critical and low-frequency type of attacks in comparison to existing models.

References

[1] C.I.A., ed. https://www.cia.gov/library/publications/the-worldfactbook/geos/cia2010.html.: The world factbook., 2010.
[2] K. A. Scarfone and P. M. Mell, "Sp 800-94. guide to intrusion detection and prevention systems (idps)," 2007.
[3] D. E. Denning, "An intrusion-detection model," *IEEE Transactions on software engineering*, pp. 222-232, 1987.
[4] M. B. Lodhi, V. Richhariya, and M. Parmar, "A survey on Data Mining based Intrusion Detection Systems," *International Journal of Computer Networks and Communications Security*, vol. 2, p. 485490, 2014.
[5] R. Tewatia and A. Mishra, "Introduction To Intrusion Detection System," 2015.
[6] K. K. Abhaya, R. Jha, and S. Afroz, "Data mining techniques for intrusion detection: A review," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 3, pp. 6938-6942, 2014.
[7] A. Balogun and R. Jimoh, "Anomaly Intrusion Detection Using An Hybrid Of Decision Tree And K-Nearest Neighbor," *Journal of Advances in Scientific Research & Applications (JASRA)*, vol. 2, pp. 67-74, 2015.
[8] J. Han, J. Pei, and M. Kamber, *Data mining: concepts and techniques*: Elsevier, 2011.

Licensed Under Creative Commons Attribution CC BY

[9] I. Jolliffe, "Principal component analysis: Wiley Online Library," 2005.

[10] J. K. Bains, K. K. Kaki, and K. Sharma, "Intrusion Detection System with Multi Layer using Bayesian Networks," *International Journal of Computer Applications*, vol. 67, 2013.

[11] M. Amiri, M. Eftekhari, and F. Keynia, "Using naïve bayes classifier to accelerate constructing fuzzy intrusion detection systems," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 2, 2013.

[12] J. R. Quinlan, "C4. 5: Programming for machine learning," *Morgan Kauffmann*, vol. 38, 1993.

[13] K. Adhatrao, A. Gaykar, A. Dhawan, R. Jha, and V. Honrao, "Predicting students' performance using ID3 and C4. 5 classification algorithms," *arXiv preprint arXiv:1310.2071*, 2013.

[14] A. Ibáñez, C. Bielza, and P. Larrañaga, "Cost-sensitive selective naive Bayes classifiers for predicting the increase of the h-index for scientific journals," *Neurocomputing*, vol. 135, pp. 42-52, 2014.

[15] W. Yassin, N. I. Udzir, Z. Muda, and M. N. Sulaiman, "Anomaly-based intrusion detection through k-means clustering and naives bayes classification," in *Proc. 4th Int. Conf. Comput. Informatics, ICOCI*, 2013, pp. 298-303.

[16] B. Hssina, A. Merbouha, H. Ezzikouri, and M. Erritali, "A comparative study of decision tree ID3 and C4. 5," *International Journal of Advanced Computer Science and Applications*, vol. 4, pp. 13-19, 2014.

[17] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on*, 2009, pp. 1-6.

[18] KDDCup, "Data," invol. Accessed 23 September 2017, ed. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html: Accessed 23 September 2017, 1999.

[19] L. Dhanabal and S. Shantharajah, "A study on NSL-KDD dataset for intrusion detection system based on classification algorithms," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, pp. 446-452, 2015.

[20] A. Santra and C. J. Christy, "Genetic algorithm and confusion matrix for document clustering," *International Journal of Computer Science*, vol. 9, pp. 322-328, 2012.

[21] E. Kim and S. Kim, "A Novel Anomaly Detection System Based on HFR-MLR Method," in *Mobile, Ubiquitous, and Intelligent Computing*, ed: Springer, 2014, pp. 279-286.

[22] W. Xuren, H. Famei, and X. Rongsheng, "Modeling intrusion detection system by discovering association rule in rough set theory framework," in *Computational Intelligence for Modelling, Control and Automation, 2006 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on*, 2006, pp. 24-24.

[23] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: an efficient data clustering method for very large databases," in *ACM Sigmod Record*, 1996, pp. 103-114.

[24] A. N. Toosi and M. Kahani, "A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers," *Computer communications*, vol. 30, pp. 2201-2212, 2007.

[25] L. M. Ibrahim, D. T. Basheer, and M. S. Mahmod, "A comparison study for intrusion database (Kdd99, Nsl-Kdd) based on self organization map (SOM) artificial neural network," *Journal of Engineering Science and Technology*, vol. 8, pp. 107-119, 2013.

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

[27] N. D. W. Cahyani, B. Martini, K. K. R. Choo, and A. Al-Azhar, "Forensic data acquisition from cloud-of-things devices: windows Smartphones as a case study," *Concurrency and Computation: Practice and Experience*, vol. 29, 2017.

Licensed Under Creative Commons Attribution CC BY