World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:8, No:9, 2014

Feature Based Unsupervised Intrusion Detection

Deeman Yousif Mahmood, Mohammed Abdullah Hussein

***Abstract*—**The goal of a network-based intrusion detection system is to classify activities of network traffics into two major categories: normal and attack (intrusive) activities. Nowadays, data mining and machine learning plays an important role in many sciences; including intrusion detection system (IDS) using both supervised and unsupervised techniques. However, one of the essential steps of data mining is feature selection that helps in improving the efficiency, performance and prediction rate of proposed approach. This paper applies unsupervised K-means clustering algorithm with information gain (IG) for feature selection and reduction to build a network intrusion detection system. For our experimental analysis, we have used the new NSL-KDD dataset, which is a modified dataset for KDDCup 1999 intrusion detection benchmark dataset. With a split of 60.0% for the training set and the remainder for the testing set, a 2 class classifications have been implemented (Normal, Attack). Weka framework which is a java based open source software consists of a collection of machine learning algorithms for data mining tasks has been used in the testing process. The experimental results show that the proposed approach is very accurate with low false positive rate and high true positive rate and it takes less learning time in comparison with using the full features of the dataset with the same algorithm.

***Keywords*—**Information Gain (IG), Intrusion Detection System (IDS), K-means Clustering, Weka.

I. Introduction

WITH the growth of the internet network, a huge increase in the number of attacks has been witnessed. Intrusion detection system has become the main topic and research area of information security. Traditionally, intrusion detection techniques come into two categories: Signature detection and Anomaly detection [1], [2]. Signature or misuse detection searches for well-known patterns of attacks, and it can only detect an attack if there an accurate matching behaviour against an already stored patterns (known as signatures). Anomaly detection on the other hand is based on establishing a normal activity profile for a system. This technique evolves itself by understanding and gathering the information about the system and determines the behaviour of the system based on it [2]. There are two primary types of IDS: host-based (HIDS) and network-based (NIDS), HIDS resides on a particular host and looks for indications of attacks on that host while NIDS resides on a separate system to watch network traffic and looking for indications of attacks that traverse that portion of the network [3]. The choice of which one to use depends on the overall risks to the organization and the resources available. The main issue in standard classification problems lies in minimizing the probability of error while performing the classification decision. Hence, the key point is how to choose an effective classification approach to build accurate intrusion detection systems in terms of high detection rate while keeping a low false alarm rate [4]. Our proposed approach combines K-means clustering algorithm with Information Gain as a filtering approach for feature selection and it produces better classification accuracy with other existing approaches. We have performed two class (attack or normal) clustering to verify the effectiveness of the proposed IDS system using NSL-KDD dataset. The NSL-KDD is a new version of KDDcup99 dataset, which is considered as a standard benchmark for intrusion detection evaluation [5]. The training dataset of NSL-KDD is similar to KDDcup99 and consists of approximately 4,900,000 single connection vectors, each of which contains 41 features and is labelled as either normal or attack type [4]. Every instance in the dataset has 42 features or attributes including target class as shown in Table I.

TABLE I
FEATURES OF NSL-KDD DATASET

Sr. No	Feature Name	Sr. No	Feature Name
1	Duration	22	s_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	s_host_login	42	Normal or Attack

II. Related Works

This section summarizes some of the techniques and algorithms that were used in designing and developing intrusion detection systems. In [4] the authors proposed an intrusion detection system model based on K-star and Information gain for feature set reduction. The key idea of the

Deeman Yousif Mahmood is with the Computer and Internet Center, University of Kirkuk, Kirkuk 36001, Iraq (e-mail: deeman-yousif@uokirkuk.edu.iq).

Dr. Mohammed Abdullah Hussein is the Dean of Informatics Technical College, Sulaimani Polytechnic University, Sulaimani 46001, Iraq (e-mail: engmohamed.hussain@gmail.com).

paper is to take advantage of instance-based classifier and dataset features reduction for intrusion detection system, the model has the ability to recognize attacks with high detection rate and low false negative. In [6] Stein and Chen applied the genetic algorithm and the decision tree algorithm for intrusion detection. They used the genetic algorithm technique for the feature reduction. In [7] an Intrusion detection system has been effectively introduced by using Principal Component Analysis (PCA) as an approach to select the optimum feature subset with Support Vector Machines (SVMs) as the system classifier. In [8] Horeis used self-organizing maps (SOM) and radial basis function (RBF) networks. The system offers better results than IDS based on RBF or SOM networks alone. [9] Shows that the dimension reduction and identification of effective network features for category-based selection can reduce the processing time in an intrusion detection system while maintaining the detection accuracy within an acceptable range.

III. Proposed Framework for IDS

In this section we present the whole framework used in this paper, and then we will discuss the main models used, which are: information gain (IG) for feature selection and K-means clustering algorithm. The proposed architecture initially filters the given dataset using information gain for feature selection. Features with the highest information gain are the criteria for the selection of the attributes. After features reduction the clustering algorithm is implemented.

Experimental results show that learning time of the algorithm is obviously decreased without compromising the accuracy of the algorithm, which is desirable feature in any IDS. After selecting the features the data set is passed to the K-means clustering algorithm with k=2 for training and testing. A test mode with splitting by 60.0% for training set and the remainder for testing set has been used. The block diagram of the proposed method is shown in Fig. 1.

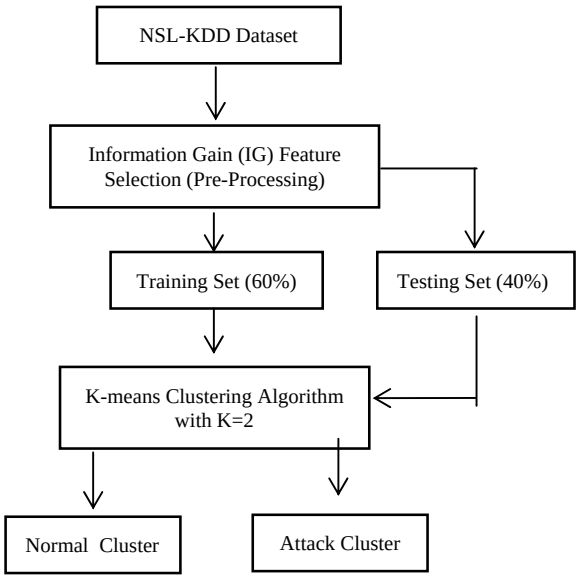

Fig. 1 Block diagram of the proposed system

A. Information Gain

In order to make the IDS more efficient, the dimensions and the complexity of the data have been reduced by feature selection process. As feature selection can reduce both the data and the computational complexity it can make the process more efficient and be used to select more useful feature subsets. It is the process of choosing a subset from the original features so that the feature space is optimally reduced to the evaluation criterion [10].

Choosing a good subset of features proves to be significant in improving the performance of the system. In Information Gain the features are filtered to create the most prominent feature subset before the start of the learning process [4]. Information gain *IG(A)* is the measure of the difference in entropy from before to after, if the set *S* is to get split on attribute *A*. In other words, how much uncertainty in *S* was reduced after splitting set *S* on attribute *A*.

$$IG(A) = H(S) - \sum_{t \in T} p(t) H(t) \quad (1)$$

where:

- *H(S)* - Entropy of set *S*
- *T* - The subsets created from splitting set *S* by attribute *A*.
- *p(t)* - The proportion of the number of elements in t to the number of elements in set *S*
- *H(t)* - Entropy of subset *t*

While entropy *H(S)* is a measure of the amount of uncertainty in the (data) set *S*.

$$H(S) = -\sum_{x \in X} p(x) log_2 p(x) \quad (2)$$

where,

- *S* - The current (data) set for which entropy is being calculated.
- *X*- Set of classes in *S*.
- *p(x)* - The proportion of the number of elements in class *x* to the number of elements in set *S*.

Weka implementation of the Information gain attribute selector (called Info Gain Attribute Eval) [4] is used to determine the effectiveness of the attributes. The attributes are ranked in decreasing order by the information gain values and as shown in Table II. According to attribute ranks only 23 attributes of the dataset with highest ranks will be chosen for training and testing of the algorithm.

B. K-means Clustering Algorithm

Clustering algorithms are used to group unlabelled data. K-means is one of the simplest unsupervised clustering algorithms.

TABLE II
ATTRIBUTES RANKING BY INFORMATION GAIN

Attribute Rank	Sr.No.	Attribute name
0.56585	6	dst_bytes
0.44723	30	diff_srv_rate
0.43943	26	srv_serror_rate
0.43243	25	serror_rate
0.42913	5	src_bytes
0.40199	12	logged_in
0.39734	38	dst_host_serror_rate
0.39719	29	same_srv_rate
0.36677	39	dst_host_srv_serror_rate
0.3516	23	Count
0.34968	4	Flag
0.28421	33	dst_host_srv_count
0.27946	34	dst_host_same_srv_rate
0.23973	35	dst_host_diff_srv_rate
0.22846	37	dst_host_srv_diff_host_rate
0.19532	32	dst_host_count
0.17903	31	srv_diff_host_rate
0.17443	3	Service
0.14591	8	wrong_fragment
0.12908	36	dst_host_same_src_port_rate
0.12082	41	dst_host_srv_rerror_rate
0.11019	19	num_access_files
0.10764	16	num_root
0.09036	28	srv_rerror_rate
0.08697	27	rerror_rate
0.08288	1	Duration
0.07317	40	dst_host_rerror_rate
0.06468	2	protocol_type
0.04755	24	srv_count
0.007	22	is_guest_login
0	9	Urgent
0	7	Land
0	18	num_shells
0	17	num_file_creations
0	21	is_host_login
0	20	num_outbound_cmds
0	11	um_failed_logins
0	10	Hot
0	15	su_attempted
0	13	num_compromised
0	14	root_shell
Target class	42	Normal or Attack

The aim of K-means cluster is to partition a given set of data into clusters, where data belonging to different clusters should be as different as possible. The algorithm is a partitioning prototype-based technique that divides the data set into K clusters, where the integer k needs to be specified, and run for a range of K values. Assignment of the data points to clusters is depending upon the distance between cluster centroid and data point [11]. K-means algorithm uses Euclidean distance, which is a distance function used to compute the distance between two objects, and it's defined as:

$$d(x,y) = \sqrt{\sum_{i=1}^{m}(x_i - y_i)^2} \quad (3)$$

Two input vectors with m quantitative features where $x = (x_1,\ldots,x_m)$ and $y = (y_1,\ldots,y_m)$.

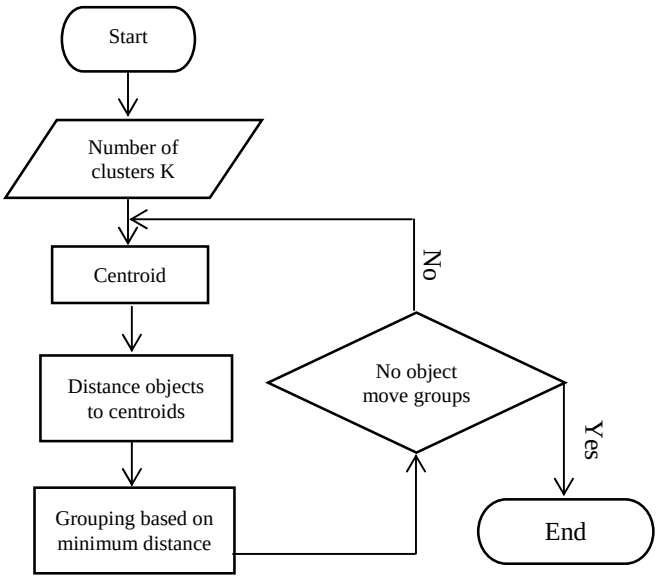

Fig. 2 K-means clustering flowchart

TABLE III
CLUSTER CENTROIDS FOR EACH SELECTED FEATURES

Attribute name	Cluster 0 (Attack)	Cluster 1 (Normal)
dst_bytes	0	9645.8296
diff_srv_rate	0.0736	0.0597
srv_serror_rate	1	0.0114
serror_rate	0.9972	0.0149
src_bytes	0	7692.53
logged_in	0	1
dst_host_serror_rate	0.0012	0.1745
same_srv_rate	0.1211	0.8709
dst_host_srv_serror_rate	0.9907	0.0039
Count	167.77	49.65
Flag	S0	SF
dst_host_srv_count	15.9	146.992
dst_host_same_srv_rate	0.0719	0.6799
dst_host_diff_srv_rate	0.0714	0.082
dst_host_srv_diff_host_rate	0.0011	0.0428
dst_host_count	246.185	153.848
srv_diff_host_rate	0	0.1379
Service	private	http
wrong_fragment	0	0.0545
dst_host_same_src_port_rate	0.0116	0.1958
dst_host_srv_rerror_rate	0.0032	0.1844
num_access_files	0	0.014
num_root	0	1.4469

The general steps of K-means algorithm are as written below and shown in the flowchart of Fig. 2:

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2. Assign each object to the group that has the closest centroid.

3. When all objects have been assigned, recalculate the positions of the K centroids.
4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

The above is done using K= 2 and with 23 attributes selected by IG. Table III shows the cluster centroids calculated for the selected features.

IV. Results and Discussion

A. Evaluation

To evaluate the classifier used in this work, we applied the evaluation indices as follows:

True positive (TP) for correctly identified, true negative (TN) for correctly rejected, false positive (FP) for incorrectly identified, Precision, Recall, F-Measure, and Accuracy. Precision and Recall are not dependent on the size of training and test samples. These metrics are derived from a basic data structure known as the confusion matrix [4]. A sample confusion matrix for a two class case can be represented as shown in Table IV.

TABLE IV
Confusion Matrix

	Predicted Class		
Actual Class	Activity	Attack	Normal
	Attack	TP	FN
	Normal	FP	TN

These metrics are defined as follows:

$$Precision = \frac{tp}{tp+fp} \quad (4)$$

$$Recall = \frac{tp}{tp+fn} \quad (5)$$

Recall in this context is also referred to as the True Positive Rate or Sensitivity, and **Precision** is also referred to as Positive predictive value (PPV); other related measures used in classification include True Negative Rate and Accuracy. True Negative Rate is also called **Specificity**.

$$True\ negative\ rate = \frac{tn}{tn+fp} \quad (6)$$

Accuracy is the most basic measure of the performance of a learning method. This measure determines the percentage of correctly classified instances. From the confusion matrix, we can state that:

$$Accuracy = \frac{tp+tn}{tp+tn+fp+fn} \quad (7)$$

F-measure is a measure of test accuracy. It considers both the precision and the recall of the test to the F-measure. The F-measure can be interpreted as a weighted average of the precision and recall, where F-measure reaches its best value at 1 and worst score at 0.

The traditional F-measure is the harmonic mean of precision and recall:

$$F - measure = \frac{2*Precision*Recall}{Precision+Recall} \quad (8)$$

B. Results

The K-means clustering algorithm is used in two ways. First, the clustering model is implemented by using all the features of the dataset. The results of this evaluation are summarized in Table V.

TABLE V
Results of Clustering Model (K-Means) with All Attributes

Parameter	Value
Accuracy	92.0635 %
Error Rate	7.9365 %
Average True Positive Rate	92.1%
Average False Positive Rate	7.6%
Average Precision	92.3%
Average Recall	92.1%
Average F-Measure	92.1%
Learning Time	18.07 sec.

Then the clustering algorithm K-means is evaluated on the dataset by using feature reduction using the Information Gain measure. The results of this test are summarized in Table VI.

TABLE VI
Results of Clustering Model (K-Means) with Feature Reduction (IG) Only 23 Features

Parameter	Value
Accuracy	97.22 %
Error Rate	2.778 %
Average True Positive Rate	97.2%
Average False Positive Rate	2.9%
Average Precision	97.2%
Average Recall	97.2%
Average F-Measure	97.2%
Learning Time	7.93 Sec.

Fig. 3 is showing in a graphical way a comparison between the two methods using K-means.

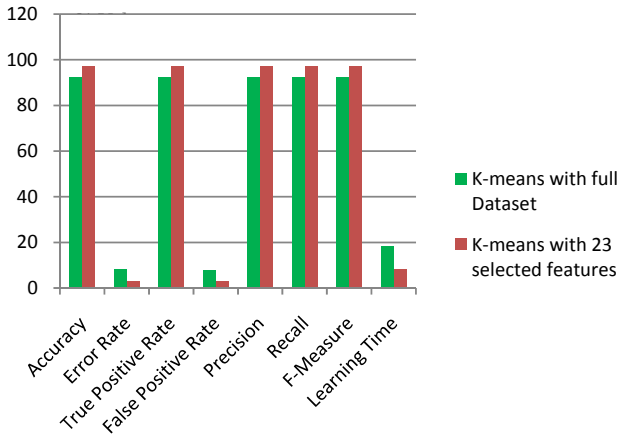

Fig. 3 Comparison chart between two K-means clustering methods

V. Conclusion

In this work, the K-means clustering (known by its high accuracy for clustering network traffics) for Intrusion Detection System has been implemented. The NSL-KDD dataset [12] has been used in two ways (with the same clustering algorithm), first, using all the dataset features and then in a reduced form. In the reduced form only 23 features are selected from the 41 features using Information Gain of the attributes. The results show that there is a significant decrease in learning time of the algorithm and an increase in the accuracy.

The results of experiments done in this work using Weka framework [13] emphasized that information gain is a suitable technique for feature reduction, and the K-means clustering algorithm is convenient and effective methodology to be used in the field of intrusion detection (as an unsupervised technique). The technique could be used as a robust base in any intrusion detection system for detecting new and unknown types of attacks.

References

[1] Bhavin Shah and Bhushan H Trivedi, "Artificial Neural Network based Intrusion Detection System: A Survey" International Journal of Computer Applications (0975 – 8887) Volume 39– No.6, February 2012.

[2] Gaikwad, Sonali Jagtap, Kunal Thakare, and Vaishali Budhawant "Anomaly Based Intrusion Detection System Using Artificial Neural Network and Fuzzy Clustering" International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 9, November- 2012, ISSN: 2278-0181.

[3] Sandip Sonawane , Shailendra Pardeshi, and Ganesh Prasad "A survey on intrusion detection techniques" World Journal of Science and Technology 2012, 2(3):127-133, ISSN: 2231 – 2587.

[4] Deeman Y. Mahmood, Mohammed A. Hussein "Intrusion Detection System Based on K-Star Classifier and Feature Set Reduction" International Organization of Scientific Research Journal of Computer Engineering (IOSR-JCE) Vol.15, Issue 5, PP. 107-112, Dec. 2013.

[5] Chunhua Gu and Xueqin Zhang," A Rough Set and SVM Based Intrusion Detection Classifier", Second International Workshop on Computer Science and Engineering, 2009.

[6] Gary Stein, Bing Chen, "Decision Tree Classifier for network intrusion detection with GA based feature selection", University of Central Florida. ACM-SE 43, proceedings of 43rd annual Southeast regional Conference. Volume 2, 2005, ACM, New York, USA.

[7] Heba F. Eid, Ashraf Darwish, Aboul Ella Hassanien, and Ajith Abraham" Principle Components Analysis and Support Vector Machine" based Intrusion Detection System", IEEE 2010.

[8] Horeis, T, "Intrusion detection with neural network - Combination of self-organizing maps and redial basis function networks for human expert integration", a Research report 2003. Available in hap://ieee-cis.org/Jiles/ EA C-Research-2003-Report-Horeis.pdf

[9] Zargar, G. R. "Category Based Intrusion Detection Using PCA", International Journal of Information Security (October 2012), 3, 259-271.

[10] Yogendra Kumar Jain, Upendra "Intrusion Detection using Supervised Learning with Feature Set Reduction", International Journal of Computer Applications (0975 – 8887) Volume 33– No.6, November 2011.

[11] A. M. Riad, Ibrahim Elhenawy ,Ahmed Hassan and Nancy Awadallah, "Visualize Network Anomaly Detection by Using K-Means Clustering Algorithm", International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.5, September 2013.

[12] The Knowledge Discovery in Databases, NSL-KDD dataset, http://nsl.cs.unb.ca/NSL-KDD/

[13] University of Waikato, WEKA: Waikato environment for knowledge analysis. Data Mining Software in Java. http://www.cs.waikato.ac.nz/ml/weka/.