http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 10 (3): 648-655, 2011
ISSN 1812-5638 / DOI: 10.3923/itj.2011.648.655

A K-Means and Naive Bayes Learning Approach for Better Intrusion Detection

Z. Muda, W. Yassin, M.N. Sulaiman and N.I. Udzir
Faculty of Computer Science and Information Technology, University Putra Malaysia,
43400 UPM Serdang, Selangor, Darul Ehsan, Malaysia

Abstract: Intrusion Detection Systems (IDS) have become an important building block of any sound defense network infrastructure. Malicious attacks have brought more adverse impacts on the networks than before, increasing the need for an effective approach to detect and identify such attacks more effectively. In this study two learning approaches, K-Means Clustering and Naïve Bayes classifier (KMNB) are used to perform intrusion detection. K-Means is used to identify groups of samples that behave similarly and dissimilarly such as malicious and non-malicious activity in the first stage while Naïve Bayes is used in the second stage to classify all data into correct class category. Experiments were performed with KDD Cup '99 data sets. The experimental results show that KMNB significantly improved and increased the accuracy, detection rate and false alarm of single Naïve Bayes classifier up to 99.6, 99.8 and 0.5%.

Key words: Intrusion detection system, K-Means clustering, Naïve Bayes classifier, accuracy, detection rate, false alarm

INTRODUCTION

The major key role of Intrusion Detection System (IDS) is for detecting various kinds of attacks and securing the applications and networks in the pervasively connected network environment. Any attempt for file modification, malicious activity and unauthorized entrance can be monitored through the IDS. There are two common techniques used to detect intruders: the misuse detection or signature based detection and anomaly detection. A signature based IDS monitors packets on the network and compares it against signatures stored inside the database from known malicious threats. They operate in such a way like virus scanner, seeking for known attacks or signatures which match the intrusion event. Signature based IDS, however, needs to be regularly updated with new signature (attacks), as it is unable to detect unknown attacks (Gandhi and Srivatsa, 2008).

Anomaly-based IDS is more established and use normal usage pattern as a baseline. It will deduce anything that widely deviates from its normal profile as a possible intrusion (Leung and Leckie, 2005). Anomaly detection techniques normally investigate user patterns, such as ordinary user access or execute process which is not privileged to them (Feng *et al*., 2003; Sekar *et al*., 2001). The ability to detect attempts to exploit new and unforeseen vulnerabilities is a major advantage for anomaly-based IDS. However, the inability to cover the entire scope of an information system behavior during the learning phase and the high false positive rate, become the major disadvantage of an anomaly-based IDS (John McHugh *et al*., 2000).

The limitations of signature and anomaly detection have lead many researches applied the technology of data mining and machine learning, which can own a better detection capacity for unknown attacks and reduce false alarm rate. Thus, most researchers shift in designing and developing this kind of anomaly based IDS to obtain high detection rate with low false alarm rates for their proposed approach and it has been the most challenging task to achieve (Gang *et al*., 2010; Su-Yun and Yen, 2009).

In this study, two learning approaches for intrusion detection are used which are K-Means method for clustering and Naïve Bayes method for classification. We name this combination as K-Means Naïve Bayes (KMNB). Clustering is useful in intrusion detection for separating malicious activity against non-malicious activity. Clustering provides significant advantages over the classification techniques which help identifies group of data that behave similarly or show similar characteristics in earlier. When compared with Naïve Bayes classifier experiment, KMNB shows a significant improvement in terms of accuracy, detection rate and false alarm.

Over the last decade, various approaches have been developed and proposed in order to detect an intrusion. Signature based detection are not capable to detect unforeseen attacks as anomaly detection (Upadhyaya *et al*., 2001). However, anomaly detection

Corresponding Author: Zaiton Muda, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400 UPM Serdang, Selangor, Darul Ehsan, Malaysia Tel: +60-3-8947-1706 Fax: +60-3-8946-6577

raises false alarm when some normal connection is falsely identified as an attack. Several methodologies have been applied to resolve the problem of false alarm, accuracy and detection capabilities overall. Furthermore various machine learning algorithms have been proposed in the intrusion detection field to date.

Clustering techniques have increased in popularity over the past couple of decades, example those proposed by Anderberg (1973) and Jain and Dubes (1988). The basic and widely used algorithm for cluster analysis is K-means (Halkidi *et al*., 2001). Eric *et al*. (2001) look for outliers in the connection of log for anomalies in the network traffic through k-means clustering. Eskin *et al*. (2002) and Chan *et al*. (2006) used a similar approach by applied fixed width and k-nearest clustering techniques.

Xiang *et al*. (2004) designed and proposed a model which contains three-level of decision tree classification to increase detection rate. This model is more efficient in detecting known attacks but a serious shortcoming of this approach is due to its low detection rate for unknown attacks as well as high false alarm rate generated. Jeffrey *et al*. (2006) used cluster analysis to identify a group of traffics which is similar with each other by using K-Means and DBSCAN.

Modeling intrusion detection system using a hierarchical hybrid intelligent system combining decision tree and support vector machine (DT-SVM) was proposed by Peddabachigari *et al*. (2007). DT-SVM produces high detection rate while reduces differentiate attacks from normal behavior. Chen *et al*. (2007) introduced a Flexible Neural Tree Model (FNT) based on the combination of genetic algorithm and neural network. Cluster analysis will find similarities between data according to the characteristics found in the data and grouping of similar data objects into clusters. Several clusters formed and characterized by high similarity and high difference among data as suggested by Wu *et al*. (2007).

Meanwhile (Panda and Patra, 2008) compared various data mining algorithm in order to detect intrusion in network. Panda and Patra (2008) concluded that their proposed approaches increased the detection rate while reducing the false alarm rate but can still be improved. Ali *et al*. (2009) use clustering approach in order to minimize irrelevant tagging of response classes for response classes re-tagging. In Tsai *et al*. (2010), the data of each category was assigned into k clusters through K-means clustering and train the SVM by using new dataset which consist of only the centers of cluster. Anyhow, the false alarm values still can be reducing.

Although an effective machine learning algorithms have been proposed in the intrusion detection fields and related work, generally there are still room to improve the accuracy and detection rate with low false alarm. The new KMNB learning approach offers high detection and accuracy with low detection rates compared to others in detecting anomaly based network intrusions.

K-MEANS NAÏVE BAYES (KMNB) LEARNING APPROACH

Learning approaches promises high detection rate in detecting new attacks, but also yields very high false alarm rates. KMNB learning approach is formed by combining clustering and classification techniques. K-means clustering technique is used as a pre-classification component for grouping similar data in earlier stage. Next, for the second stage clustering the data will be classified by category of attack using Naïve Bayes classifier. Thus, data which are misclassified during the first stage will be classified accordingly by its category in the second stage.

The iterative K-Means minimizes an objective function, in this case an algorithm for clustering N input data points $x_1, x_2, \ldots, x_N$ into k disjoint subsets C_i, $i = 1, \ldots, k$, each containing n_i data points, $0 < n_i < N$, minimizes the following mean-square-error (MSE) cost-function:

$$J_{MSE} = \sum_{i=1}^{k} \sum_{x_t \in C_i} \| x_t - c_i \|^2 \quad (1)$$

where, X_t is a vector representing the t-th data point in the cluster C_i and c_i is the geometric centroid of the cluster C_i. Finally, this algorithm aims at minimizing an objective function, in this case a squared error-function, where $\|X_t - C_i\|^2$ is a chosen distance measurement between data point x_t and the cluster centre c_i (Zalik, 2008).

Figure 1a-d show the steps involved in K-Means clustering process for stage 1. As a standard classification for network intrusion, network attacks can be divided into four main categories: DoS, Probe, U2R and R2L, as stated by Lippmann *et al*. (2000). In this stage, the value of k is 3 representing 3 clusters (C1, C2 and C3). C1 is used to group an attack data such as Probe, U2R and R2L while C2 is used to group DoS attack data. In order to separate normal data from an attack, C3 has been used. K-Means clustering will partition the input dataset into k clusters as initial values named seeds-points by cluster's centroids (also called cluster centre). A centroid is the mean value of the numerical data contained within a cluster.

From Fig. 1b each input will be assigned to the closest centroid by squared distances between the inputs (also called input data points) and centroids. New

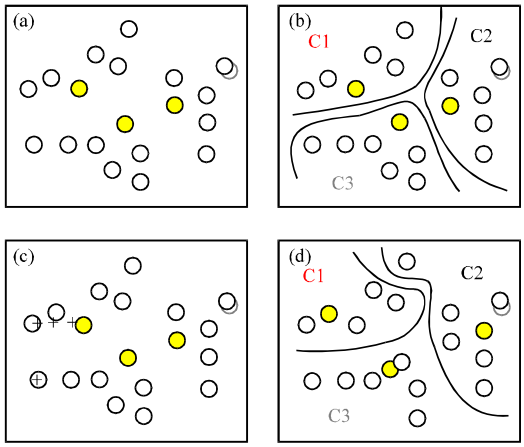

Fig. 1: Clustering process using K-Means clustering, (a) Initial centroid, (b) assigns instances, (c) finds centroid and (d) new clusters

centroids are generated for each cluster by calculating the mean values of the input set assigned to each cluster as in Fig. 1c. Steps in Fig. 1b and c are repeated until convergence has been reached. Once the coverage has been reached, there are three clusters performed completely which contain an attack and normal instances in the correct group. Thus the K-means algorithm is as follows:

- Select initial centers of the K clusters. Repeat step 2 through 3 until the cluster membership stabilizes
- Generate a new partition by assigning each data to its closest cluster centers
- Compute new clusters as the centroids of the clusters

In data mining and machine learning, Naïve Bayes becomes one of the popular learning algorithms. It analyzes the relationship between independent variable and the dependent variable to derive a conditional probability for each relationship. Thus, Naïve Bayes is based on a very strong independence assumption and the construction of Naïve Bayes is very simple. Using Bayes theorem:

$$P(H|X) = P(X|H)\ P(H) / P(X)$$

Let X be the data record and H be some hypothesis represent data record X which belongs to a specified class C. For classification, we want to determine P(H|X) the probability that the hypothesis H holds, given the observed data record X. P(H|X) is the posterior probability of H conditioned on X. In contrast, P(H) is the prior probability, or apriority probability. The posterior probability, P(H|X), is based on more information (such as background knowledge) than the prior probability, P(H), which is independent of X. Similarly, P(X|H) is the posterior probability of X conditioned on H. Bayes theorem is useful in that it provides a way of calculating the posterior probability, P(H|X), from P(H), P(X) and P(X|H).

For stage 2 in KMNB, Naïve Bayes classifier has been used to group all data from stage 1 into more specific groups. Five categories of classes (K1 = Normal, K2 = DoS, K3 = Probe, K4 = R2L and K5 = U2R) are considered. Given x, predict K1, K2, K3, K4 and K5. By Bayes rule:

$$P(K_i \mid X) = \frac{P(X \mid K_i)P(K_i)}{P(X)} \tag{2}$$

where, K_i represents the category of classes and X is the data record. X can be divided into pieces of instances, say x_1, x_2...x_n which are relative to the attributes X1, X2... Xn, respectively. Probability is obtained as follows:

$$P(K_i \mid X) = \frac{P(x_1 \mid K_i)P(x_2 \mid K_i)...P(x_n \mid K_i)}{P(X)} \tag{3}$$

However, having strong dependencies among attributes may result in poor performance. Improving the constraint of Naïve Bayes classifier in terms of accuracy, detection rate and false alarm have led us to group the data using K-Means clustering technique.

The combination of K-Means clustering and Naïve Bayes classifier shows an improvement compared to the Naïve Bayes single classifier, as it increases accuracy, detection rate and reduces false alarms.

EXPERIMENTAL OPERATIONS

The data set used to perform the experiment was taken from KDD Cup '99, which is widely accepted as a benchmark dataset and referred by many researchers (Tsai *et al*., 2010; Gang *et al*., 2010; Su-Yun and Yen, 2009). "10% of KDD Cup'99" and "corrected (test)" from KDD Cup '99 data set was chosen to evaluate KMNB as training and testing data sets to detect intrusion. The entire KDD Cup '99 data set contains 41 features. Description for the available features and intrusion instances can be found in (Breiman *et al*., 1984).

KDD Cup '99 data set covers four major categories of attacks. In order to demonstrate the abilities of detecting different kinds of intrusions, the training and testing data set cover all intrusion categories as below:

- **Denial of Service (DoS):** Where an attacker usually occupies all system sources, disables system resources and makes some computing or memory resources too busy or too full to handle legitimate requests, or deny legitimate users access to a machine. Examples are Smurf, Mailbomb, SYN Flooding, Ping Flooding, Process table, Teardrop, Apache2, Back, Land
- **Remote to User (R2L):** Where an attacker sends packets to remote machine over a network and exploits some vulnerability to gain local access as a user of that machine. Examples are Ftp_write, Imap, Named, Phf, Sendmail and SQL Injection
- **User to Root (U2R):** An attacker takes advantage of the system leak by accessing a normal user account on the system and able to exploit system vulnerabilities to get legitimate administrator access to the system. Examples are Loadmodule, Perl, Fdformat
- **Probing:** Where an attacker does some preparation step before launching attacks by scanning a network of computers to gather information or to find known vulnerabilities. An attacker will use this information to look for exploits by determining the targets and the type of operating system. Examples are Nmap, Satan, Ipsweep, Mscan

"10% of KDD Cup'99" distribution records as training dataset by class type is summarized in Table 1. Meanwhile Table 2 shows the testing dataset information obtained from "Corrected (Test)". The behavior of data for intrusion detection system can be categorized as in Table 3.

Table 1: Sample distribution of the training data set

Class	No. of samples	Sample percentage
Normal	97277	19.69
Probe	4107	0.83
DoS	391458	79.24
U2R	52	0.01
R2L	1126	0.23
Total	494020	100.00

Table 2: Sample distribution of the testing data set

Class	No. of samples	Sample percentage
Normal	60593	19.40
Probe	4166	1.33
DoS	231455	74.40
U2R	88	0.028
R2L	14727	4.73
Total	311029	100.00

Table 3: Behavior of data

Actual	Predicted normal	Predicted attack
Normal	TN	FP
Intrusions (attacks)	FN	TP

IDS requires high detection rate and low false alarm rate, thus the performance of an IDS can usually be evaluated in terms of accuracy, detection rate and false alarm as below:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

$$\text{Detection rate} = \frac{TP}{TP+FP}$$

$$\text{False alarm} = \frac{FP}{FP+TN}$$

A series of experiments was conducted using Naïve Bayes as a single classifier and KMNB approach with the benchmark dataset, KDD-Cup '99. The experiments were carried out at the Faculty of Computer Science and Information Technology, University Putra Malaysia on 2010. All data was normalized and some features have been changed before the implementation to obtain a better output. Cross-validation is one of the most commonly used methods. In 10 cross-validations the whole dataset will be divided into 10 subsets, which 9 subsets count in as the training subsets and the rest as the testing subset. The results are implemented by five category classes (Normal, Probe, DoS, U2R, R2L) and the binary category classes (Normal and Attacks), respectively.

RESULTS

As mentioned earlier, KDD Cup '99 dataset is used to evaluate the proposed approach to compare with Naïve Bayes classifier. There are two components of dataset used, which are training data and testing data. The test data contain an unforeseen attack which has not been covered in the training set. Classification process using Naïve Bayes classifier is shown in Fig. 2.

Training: Table 4 and 5 show the classification results using Naïve Bayes classifier. In short, the rates of accuracy, detection and false alarm are 97.39, 97.95 and

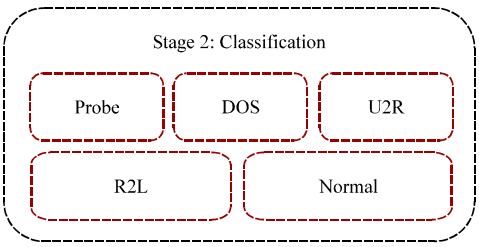

Fig. 2: Classification process using Naïve Bayes classifier

Table 4: Classification result for naïve bayes classifier using training data set

Actual	Predicted normal	Predicted probe	Predicted DoS	Predicted U2R	Predicted R2L	Accuracy (%)
Normal	8909	138	8	570	102	91.6
Probe	0	410	0	0	1	99.8
DoS	444	16	36921	1757	8	94.3
U2R	0	0	0	4	1	80.0
R2L	27	3	0	9	74	65.5

Table 5: Result of naïve bayes classifier for the normal and attacks classes using training data set

Actual	Predicted normal	Predicted attack
Normal	8909	818
Intrusions (attacks)	471	39204

Table 6: Classification result for KMNB using training data set

Actual	Predicted normal	Predicted probe	Predicted DoS	Predicted U2R	Predicted R2L	Accuracy (%)
Normal	9687	23	3	5	9	99.6
Probe	0	411	0	0	0	100.0
DoS	3	0	38936	0	207	99.5
U2R	1	0	0	2	2	40.0
R2L	35	3	2	4	69	61.6

Table 7: Result of KMNB for the normal and attacks classes using training data set

Actual	Predicted normal	Predicted attack
Normal	9687	40
Intrusions (attacks)	39	39636

Table 8: Summary of overall measurement using training data set

Measurement	Single classifier NB	Hybrid approach KMNB
Accuracy	97.39	99.84
Detection rate	97.95	99.89
False alarm	8.40	0.41
Increment-accuracy rate		+2.45
Increment-detection rate		+1.94
Reduced false alarm rate		-6.99

8.4%, respectively. Table 6 and 7 show the classification results using the proposed KMNB. KMNB outperforms Naïve Bayes classifier with 99.84% accuracy, 99.89% detection and 0.41% false alarms, respectively.

The experimental results for a single classifier Naïve Bayes and KMNB are summarized in Table 8, Fig. 3-5. The Table 8 and Fig. 3-5 representing measurement in terms of accuracy, detection rate and false alarm on the training set. From Table 8, the Naïve Bayes classifier has produced a slightly high accuracy and detection rate but with high false alarm rates. In contrast, KMNB records high accuracy and detection rate with low false alarm rates. KMNB is much better in term of misclassification with 0.41% false alarm and the accuracy and detection rates are more than 99%. The false alarm for the single Naïve Bayes classifier increased up to 8.4% with moderate accuracy and detection rates which are less than 98%. The clustering techniques used as a pre-classification component for grouping similar data by classes in the earlier stage helps KMNB produces a better result compared to Naïve Bayes classifier. The data which misclassified during the first stage was classified

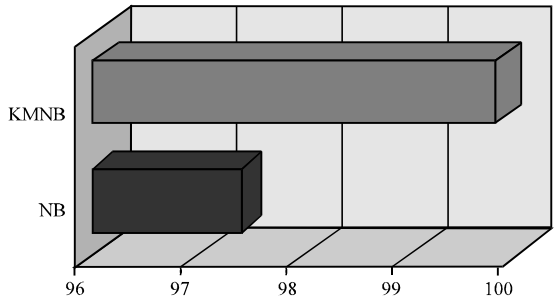

Fig. 3: Accuracy for training data set

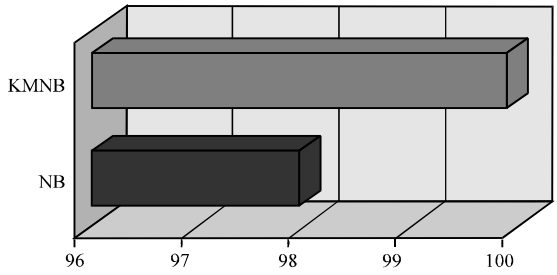

Fig. 4: Detection rate for training data set

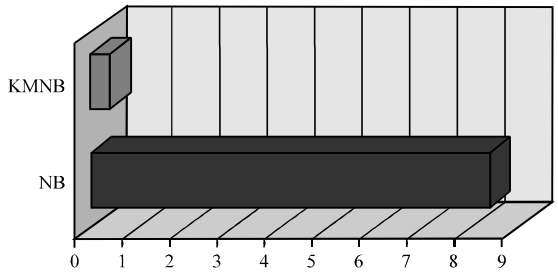

Fig. 5: False alarm for training data set

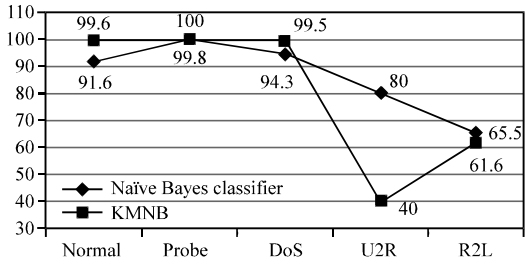

Fig. 6: Accuracy for all category classes using training data set

accordingly in the second stage, hence making KMNB outperforms Naïve Bayes classifier in term of false alarm.

Figure 6 describes the obtained results considering the overall category classes in the training set. KMNB performs better than Naïve Bayes classifier in detecting normal, probe and DoS instances. Since normal, U2R and

Table 9: Classification result for naïve bayes classifier using testing data set

Actual	Predicted normal	Predicted DoS	Predicted U2R	Predicted R2L	Predicted probe	Accuracy (%)
Normal	7875	14	1664	43	131	81.0
DoS	6431	32298	0	0	417	82.5
U2R	1	0	4	0	0	80.0
R2L	10	0	0	102	1	90.3
Probe	6	12	0	0	393	95.6

Table 10: Result of Naïve Bayes classifier for the normal and attacks classes using testing data set

Actual	Predicted normal	Predicted attack
Normal	7875	1852
Intrusions (attacks)	6448	33227

Table 11: Classification result for KMNB using testing data set

Actual	Predicted normal	Predicted DoS	Predicted U2R	Predicted R2L	Predicted probe	Accuracy (%)
Normal	9678	9	35	2	3	99.5
DoS	134	38984	0	1	27	99.6
U2R	1	0	4	0	0	80.0
R2L	4	12	3	94	0	83.2
Probe	0	3	4	0	404	98.3

Table 12: Result of KMNB for the normal and attacks classes using testing data set

Actual	Predicted normal	Predicted attack
Normal	9678	49
Intrusions (attacks)	139	39536

Table 13: Summary of overall measurement using testing data set

Measurement	Single classifier NB	Hybrid approach KMNB
Accuracy	83.19	99.60
Detection rate	94.70	99.80
False alarm	19.00	0.50
Increment-accuracy		+16.41
Increment-detection rate		+5.10
Reduced false alarm rate		-18.50

R2L instances are similar to each other; KMNB records a comparable result for R2L except U2R. However, KMNB is more efficient in classifying normal and attack instances accordingly. Table 5 proved that Naïve Bayes classifier is less efficient when it falsely predicts 818 instances as attack and 471 as normal compared to KMNB approach in Table 7 which is 40 and 39, respectively.

Testing: Table 9 and 10 show the results obtained from the experiment using the testing data set for Naïve Bayes classifier. From Table 9, the rates for accuracy, detection and false alarm are 83.19, 94.70 and 19%, respectively. Table 11 and 12 show the result of detection rate, accuracy and false alarm for KMNB using testing dataset. The new approach, KMNB performs a better performance with 99.60, 99.80 and 0.5% as detection rate, accuracy and false alarm respectively.

From Table 13, Fig. 7-9, it is clear that the proposed approach, KMNB is much better than the Naïve Bayes classifier as KMNB shows high improvement in reducing false alarm and maintaining high detection and accuracies

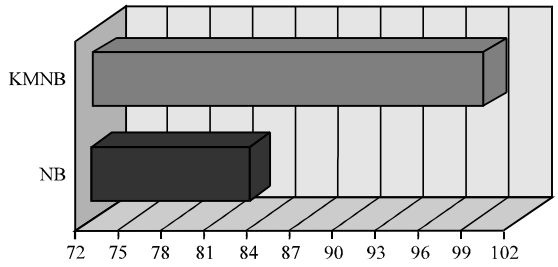

Fig. 7: Accuracy for testing data set

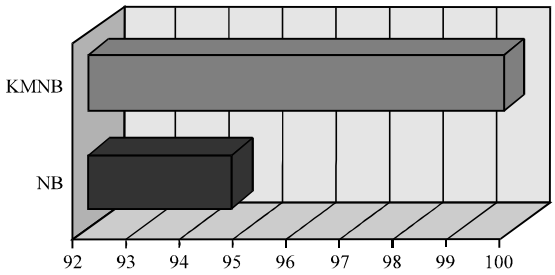

Fig. 8: Detection rate for testing data set

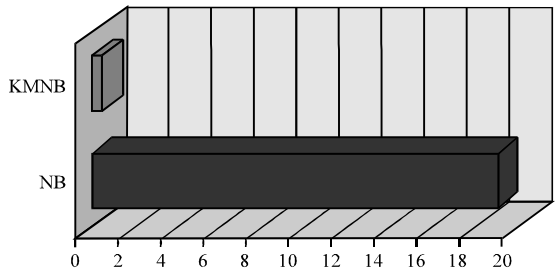

Fig. 9: False alarm for testing data set

with a larger number of unknown attacks. It is also quite evident that from Table 13 that the KMNB enhances the Naïve Bayes classifier's accuracy rate, where the accuracy for KMNB increased +16.41% while keep reducing false alarm rate up to -18.5% against Naïve Bayes' 83.19 and 19%, respectively. It can also be observed that KMNB correctly predicts most normal connection and attack exactly in comparison to Naïve Bayes classifier's failure to correctly predict all the instances. These comparisons show that KMNB is more suitable in building an efficient anomaly based network intrusion detection model.

From Fig. 10, Naïve Bayes classifier obtained low accuracies for each category classes except U2R and DoS. Although the number of atack instances are greater in the testing set, KMNB still shows a better performance compared to Naïve Bayes classifier. KMNB significantly perform well when considering the accuracies for normal and probe instances. Out of 100% normal and probe

Table 14: Comparison of KMNB with previous findings

Approaches	Normal	DoS	U2R	R2L	Probe	AC	DR	FP	FA
KMNB	99.50	99.60	80.00	83.20	98.30	99.60	99.80	0.09	0.50
TANN (Tsai *et al*., 2010)	N/A	N/A	N/A	N/A	N/A	96.91	98.95	0.80	3.83
KM-KNN (Tsai *et al*., 2010)	N/A	N/A	N/A	N/A	N/A	93.55	98.68	0.98	4.79
Hierarchical Clustering and SVM (Horng *et al*., 2011)	99.30	99.50	19.70	28.80	97.50	95.70	N/A	0.70	N/A
Hybrid artificial immune system and SOM (Powers and He, 2008)	99.40	96.80	34.60	5.20	64.70	N/A	N/A	N/A	N/A
Hybrid Classifier (Xiang *et al*., 2008)	96.80	98.66	71.43	46.97	83.40	96.78	99.21	3.20	3.20
ESC-IDS (Toosi *et al*., 2007)	98.20	99.50	14.10	31.50	84.10	95.30	N/A	1.90	N/A
Three level tree (Xiang *et al*., 2004)	42.73	97.35	61.43	23.69	93.23	N/A	N/A	N/A	N/A
Winning entry KDD Cup 1999 (Pfahringer *et al*., 1999)	99.50	83.30	13.20	8.40	97.10	N/A	N/A	N/A	N/A

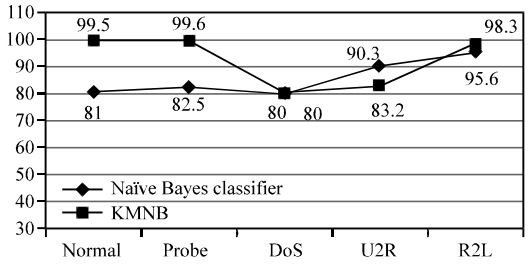

Fig. 10: Accuracy for all category classes using testing data set

instances, only 0.5 and 0.4% became false positive. This is outperformed by the Naïve Bayes classifier which earns 19 and 17.5% as false positive. DoS and R2L records a comparable result while have tendencies for U2R instances.

Further comparisons were made with other methods and the results are shown in Table 14. All previous researchers also tested their methods on KDD Cup '99 dataset. It is clear that KMNB perform high improvement in reducing false alarm while maintaining high accuracies and detection rates. KMNB detects a better percentage of attacks than the rest as proven in Table 14 where it obtains 99.6 and 99.8% as an overall accuracy and detection rate. This is because K-Means clustering technique that used as a pre-classification component in the first stage groups similar data respectively and instances which were misclassified during the first stage of clustering were classified correctly by Naïve Bayes classifier in the second stage.

CONCLUSION

The suggested approach called KMNB is evaluated and compared with the single Naïve Bayes classifier using KDD Cup '99 data set. The experimental results show that the KMNB approach achieves better accuracy and detection rates while reducing the false alarm by detecting novel intrusions accurately. The performance of Naïve Bayes classifier has been improved by applying KMNB. However, KMNB have limitation to detect intrusions that are very similar with each other such as U2R and R2L. Since U2R and R2L attacks are primary attack strategies used by attackers, honey net like techniques can be considered for the future work.

REFERENCES

Ali, S.A., N. Sulaiman, A. Mustapha and N. Mustapha, 2009. K-means clustering to improve the accuracy of decision tree response classification. Inform. Technol. J., 8: 1256-1262.

Anderberg, M.R., 1973. Cluster Analysis for Applications. Academic Press, New York, ISBN: 0120576503, pp: 359.

Breiman, L., J.H. Friedman, R.A. Olshen and J. Stone, 1984. Classification and Regression Trees. 1st Edn., Wadsworth International Group, Belmont, CA, ISBN: 978-0412048418, pp: 102-116.

Chan, P.K., M.V. Mahoney and M.H. Arshad, 2006. Managing Cyber Threats: Issues, Approaches and Challenges. Kluwer Academic Publishers, Boston.

Chen, Y., A. Abraham and B. Yang, 2007. Hybrid flexible neural-tree-based intrusion detection systems. Int. J. Intell. Syst., 22: 337-352.

Eric, B., A.D. Christiansen, W. Hill, C. Skorupka, L.M. Talbot and J. Tivel, 2001. Data mining for network intrusion detection: How to get started. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.102.8556

Eskin, E., A. Arnold, M. Preraua, L. Portnoy and S.J. Stolfo, 2002. A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data. Data Mining for Security Applications. Kluwer Academic Publishers, Boston.

Feng, H.H., O.M. Kolesnikov, P. Fogla, W. Lee and W. Gong, 2003. Anomaly detection using call stack information. Proceedings of the IEEE Symposium on Security and Privacy, May 11-14, Berkeley, CA., pp: 62-76.

Gandhi, M. and S.K. Srivatsa, 2008. Detecting and preventing attacks using network Intrusion detection systems. Int. J. Comput. Sci. Sec., 2: 49-60.

Gang, W., J. Hao, J. Ma and L. Huang, 2010. A new approach to intrusion detection using artificial neural networks and fuzzy clustering. Expert Syst. Appl., 37: 6225-6232.

Halkidi, M., Y. Batistakis and M. Vazirgiannis, 2001. On clustering validation techniques. J. Intell. Inform. Syst., 17: 107-145.

Horng, S.J., M.Y. Su, Y.H. Chen, T.W. Kao, R.J. Chen, J.L. Lai and C.D. Perkasa, 2011. A novel intrusion detection system based on hierarchical clustering and support vector machines. Expert Syst. Appli., 38: 306-313.

Jain, A.K. and R. Dubes, 1988. Algorithms for Clustering Data. Prentice-Hall, New Jersey, ISBN: 0-13-022278-X.

Jeffrey, E., M. Arlitt and A. Mahanti, 2006. Traffic classification using clustering algorithms. Proceedings of the ACM SIGCOMM 2006 Conference on Applications, Technologies, Architectures and Protocols for Computer Communications, Sept. 11-15, ACM Press, Pisa, Italy, pp: 281-286.

John McHugh, A.C. and Julia Allen, 2000. Defending yourself: The role of intrusion detection systems. IEEE Software, 17: 42-51.

Leung, K. and C. Leckie, 2005. Unsupervised anomaly detection in network intrusion detection using clusters. Proceedings of the 28th Australasian Conference on Computer Science, January 2005, Newcastle, Australia, pp: 333-342.

Lippmann, R.P., D.J. Fried, I. Graf, J.W. Haines and K.R. Kendall *et al*., 2000. Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation. Proceedings of the 2000 DARPA Information Survivability Conference and Exposition (DISCEX), Jan. 25-27, IEEE Computer Society Press, Los Alamitos, CA, pp: 12-26.

Panda, M. and M.R. Patra, 2008. A comparative study of data mining algorithms for network intrusion detection. Proceedings of the International Conference and Workshop on Emerging Trends in Technology, (ICETET'08), Nagpur, India, 504-507.

Peddabachigari, S., A. Abraham, C. Grosan and J. Thomas, 2007. Modeling intrusion detection system using hybrid intelligent systems. J. Network Comput. Appl., 30: 114-132.

Powers, S.T. and J. He, 2008. A hybrid artificial immune system and self organising map for network intrusion detection. J. Inform. Sci., 178: 3024-3042.

Sekar, R., M. Bendre, P. Dhurjati and D. Bullineni, 2001. A fast automaton-based method for detecting anomalous program behaviors. Proceedings of the IEEE Symposium on Security and Privacy, S and P, May 14-16, Oakland, CA. USA., pp: 144-155.

Su-Yun, W. and E. Yen, 2009. Data mining-based intrusion detectors. Expert Syst. Appl., 36: 5605-5612.

Toosi, A.N. and M. Kahani, 2007. A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers. Comput. Commun., 30: 2201-2212.

Tsai, C.F. and C.Y. Lin, 2010. A triangle area based nearest neighbors approach to intrusion detection. Pattern Recognition, 43: 222-229.

Upadhyaya, S., R. Chinchani and K. Kwiat, 2001. An analytical framework for reasoning about intrusions. Proceedings of the 20th IEEE Symposium on Reliable Distributed Systems, Oct. 28-31 New Orleans, Louisiana, pp: 99-108.

Wu, W.L., Y.S. Liu and J.H. Zhao, 2007. New mixed clustering algorithm. J. Syst. Simulation, 19: 16-18.

Xiang, C., M.Y. Chong and H.L. Zhu, 2004. Design of multiple-level tree classifiers for intrusion detection system. Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems, (CCIS'04), Singapore, pp: 873-878.

Xiang, C., P.C. Yong and L.S. Meng, 2008. Design of multiple level hybrid classifier for intrusion detection system using bayesian clustering and decision tree. Pattern Recognit. Lett., 29: 918-924.

Zalik, K.R., 2008. An efficient k-means clustering algorithm. Pattern Recognition Lett., 29: 1385-1391.