BILINGUAL PUBLISHING CO.
Pioneer of Global Academics Since 1984

Journal of Computer Science Research
https://ojs.bilpublishing.com/index.php/jcsr

ARTICLE

Integration of Expectation Maximization using Gaussian Mixture Models and Naïve Bayes for Intrusion Detection

Loka Raj Ghimire Roshan Chitrakar*

Department of graduate study, Nepal College of Information Technology, Nepal

ARTICLE INFO

Article history
Received: 27 February 2021
Accepted: 17 March 2021
Published Online: 20 April 2021

Keywords:
Anomaly detection
Clustering
EM classification
Expectation maximization (EM)
Gaussian mixture model (GMM)
GMM classification
Intrusion detection
Naïve Bayes classification

ABSTRACT

Intrusion detection is the investigation process of information about the system activities or its data to detect any malicious behavior or unauthorized activity. Most of the IDS implement K-means clustering technique due to its linear complexity and fast computing ability. Nonetheless, it is Naïve use of the mean data value for the cluster core that presents a major drawback. The chances of two circular clusters having different radius and centering at the same mean will occur. This condition cannot be addressed by the K-means algorithm because the mean value of the various clusters is very similar together. However, if the clusters are not spherical, it fails. To overcome this issue, a new integrated hybrid model by integrating expectation maximizing (EM) clustering using a Gaussian mixture model (GMM) and naïve Bays classifier have been proposed. In this model, GMM give more flexibility than K-Means in terms of cluster covariance. Also, they use probabilities function and soft clustering, that's why they can have multiple cluster for a single data. In GMM, we can define the cluster form in GMM by two parameters: the mean and the standard deviation. This means that by using these two parameters, the cluster can take any kind of elliptical shape. EM-GMM will be used to cluster data based on data activity into the corresponding category.

1. Introduction

Recently, through their networks, many organizations have encountered heavy network use. The large technological expansion that followed these networks, however, gave them different threads. Such threads include many types of malicious programs that affect network efficiency or unauthorized network access to data. This has encouraged work to strengthen and develop new ways of addressing and mitigating these threats. Any unauthorized operation on a computer network constitutes a network intrusion [1].

Intrusion detection is a "species of security technology that can collect information from some of the network or computer system's key points and attempt to analyze it to assess whether there is a violation of the security policy or a suspicion of the computer system's network attack." Intrusion detection methods are classified into two groups according to the different objects for intrusion detection. One is called the identification of anomaly that is used to detect the unknown intrusion. And the other is called detection of abuse, which is used to detect the identified intrusion.

Mixed intrusion detection techniques have been fo-

**Corresponding Author:*
Roshan Chitrakar,
Department of graduate study, Nepal College of Information Technology, Nepal;
Email: roshanchi@gmail.com

 DOI: https://doi.org/10.30564/jcsr.v3i2.2922

cused on to resolve shortcomings in anomaly detection and misuse detection methods. The anomaly detection model and signature detection system can be paired with three different strategies: anomaly detection followed by misuse recognition, identify anomalies and misuse concurrently, and misuse identification accompanied by anomaly detection [1].

While new technologies in intrusion detection and research have been suggested, the accuracy and detection rate as well as the false alarm rate have still to be improved. The proposed method provides high detection and precision compared to previous attack detection with low false alarm rate by using a hybrid model.

2. Related Work

Dorothy Denning first described intrusion detection in 1987 [2]. According to him, "network intrusion can be detecting by monitoring network activity in terms of data and then the system can generate alerts and responses before the infringement". Instantaneity is one of the key features of intrusion. Snort IDS applied the rule-based intrusion detection method [3, 4]. Rule-based detection system has quick detection characteristics, but it has a big problem. It cannot detect other than pre-defined types of attack. Since intruders will frequently change their technique of attack, which is often riskier. For this case, this approach cannot adopt itself so that it has not been suitable in new types of attacks. It also has a higher false alarm rate.

Intrusion detection using data mining technique requires extensive data collection in advance. Large quantities of data limit the rate of online detection [5]. Conventional intrusion detection methods are being developed using data mining [6–7] and common file analyzed [8]. In differential analyzes performed by Fisher, an et al. [9] used the approach of combining the minimum scatter class with a traditional support vector (SVM) analysis and then implemented a minimum scatter support class vector (WCS-SVM) analysis, which is better than traditional SVM. Kabir et al [10] suggested a vector based intrusion detection method (LS-SVM) that supports the least squares, called (LS-SVM) method. The new method of improved decision mapping for intrusion detection was introduced by M. Gudadhe, Al. [11] to develop an intermediate classifier for multiple decision makers. Sufyan et al. [12] used backpropagation models for artificial neural networks to detect intrusion, encouraging intrusion detection system to adapt more effectively respond to new environments and new attack types. The vast scale of the network data set takes time and effort for manual tagging. The classification of the dataset is therefore subject to clustering methods [13]. The Ymeans clustering algorithm [14] surmounts two disadvantages of K-means clustering. This is dependency and deterioration of k-means by splitting the set of data automatically into a correct number of clusters. The k-means clustering algorithm is a simple algorithm that solves the complexity of previous clustering algorithms. Traditional SOM algorithm has some disadvantages like, not providing accurate result while clustering. This has been overcome by integration of SOM and k-means [15]. One of the major problems in clustering is to determine the cluster center and number of clusters. High speed, high detection can be achieved by the parallel clustering integration algorithm [16] for IDS. The ANN classifier [17] has a good performance in the detection of intrusion. Research in [18–20] uses a mixed learning approach to have a higher detection. Shah et al. [21] compared directly to the Snort intrusion detection system and the machine learning detection performance and found the better performance in machine learning detection system.

Sheng Yi Jang et al. proposed a clustering-based intrusion detection method [22] wherein clusters consist of unlabeled datasets and have been classified as normal or abnormal by their external factors. This method's time complexity is linear with the dataset size and number of attributes. A method for anomaly detection by clustering regular user behavior is proposed by Sang Hyum Oh et al. [23] to model a user's typical behavior using the clustering algorithm. Clustering prevents statistical analysis causing inaccuracy. Therefore, the user's daily habits are more reliable than the statistical analysis.

Tasi and Lin use K-Means clustering in K-clusters to cluster data instances [24]. Next the study trains the latest dataset consisting only of cluster centers with support vector machine (SVM). They managed to achieve a high precision rate for nearly all types of attacks. This approach provides a high rate of detection but comes with a high false alarm rate.

The new approach of the IDS based on the Artificial Neural Network (ANN) with the clusters ANN and Fuzzy FC-AN Network, is suggested by Gang, Jin Xing and Jian [25]. Before a similar ANN model is trained, fuzzy clustering is carried out to formulate different models to produce different training subsets. A fuzzy module of aggregation is then used to sum the result. The subset of the training set is less complex with the use of fuzzy clusters that help the ANN learn from each subset more effectively and to detect low frequency attacks such as U2R and R2L attacks. Nevertheless, in contrast with the Naïve Bayes approach, this approach results in a lower detection rate for probing attacks.

DOI: https://doi.org/10.30564/jcsr.v3i2.2922

Shaohua et al. [26] suggested detection of intrusion based on Fuzzy SVMs (FSVM) to improve classification accuracy. The clustering algorithm's aim is to build a new training set using cluster centers. This new set will then be trained to get a support vector with FSVM. Although their findings have shown that the accuracy rate has been improved by this approach, it is not an adequate percentage.

Amiri et al. [27] used a feature selection method to improve the performance of existing classifiers by eliminating unimportant features like SVM with heavy computational challenges for large datasets. The authors have recently introduced the support vector machine of an improved least squire called PLSSVM. PLSSVM performs well in the classification of normal records and probes but misses many dynamic attacks that are very similar to normal behavior, such as DOS and U2R.

Horng [28] suggested hierarchical clustering of SVM-based IDS BIRCH as a pre-processing step and a basic feature selection method to remove unimportant features. The hierarchical clustering algorithm enhances SVM's efficiency while the simple selection of features allows the SVM model to properly classify some data. As this method was unable to differentiate between R2L and Normal data, the percentage of predictions for this class dropped dramatically.

In terms of classification accuracy and AUC, Huang, Lu and Ling [29] performed a comparative study of Naïve Bayes, Decision tree and SVM. They found that both Naïve Bayes and SVM have very similar predictive accuracy as well as similar AUC scores are produced.

Roshan Chitrakar and Huang Chauanhe proposed a hybrid anomaly detection approach using K-medoids clustering and support vector machine classification [30]. Since there may be too many support vectors in the case of using a high dimensional kernel, this also reduces the training speed, KMeans / Medoids needs a large sample and can only handle spherical shape.

S. Varuna and Dr. P. Natesan proposed an integrated model of K- Means clustering and Naïve Bayes classification for intrusion detection [1]. The integrated algorithm improved the detection rate for the normal, Probe, R2L and U2R attacks, but it does not meet the requirements for DOS.

This paper is organized as follows: Section 3 describes the proposed work and the implementation details. Section 4 contains the results and discussion.

3. Proposed Model

In this research an integrated model has been proposed. This is the integration of Expectation Maximization using Gaussian Mixture Model clustering and Naïve Bayes classifier. Data are clustered and formed five clusters with outlier. The purpose of clustering is to label the data with enhancing the accuracy and performance of model by improving capacity of parallel processing of the model. Thus, clustered data with outlier are then classified using Naïve Bayes classifier.

3.1 Description of Dataset

Each dataset record reflects a 41-feature network connection. Among them, 7 are nominal features, 34 are continuous features and a label. Label indicates that the data is either in normal status or in one of the 39 identified attack status. The NSL-KDD data can be categorized as either a standard class or one of four attack classes, i.e. remote to local, denial of service, Users to root and Probe classes.

Table 3.1 lists the number of instances in the training and testing data set of every type of attack group and the total number of instances in each data set.

Table 3.1 Size and Distribution of Training and Test Data Based on Attack Class

Attack Class	Training data size	Test data size
Normal	67343	9711
Prbe	11656	2421
Remote to local	995	2754
Denial of services	45927	7456
User to root	52	200
Total	125973	22542

3.2 Feature Scaling and Selection

There are 41 attributes in the NSL-KDD dataset. In this analysis, 14 common and basic characteristics, also known as traditional characteristics, are used.

3.3 Conceptual Model Diagram

The proposed model consists of three sub modules. These are data preprocessing module, clustering and classification module with outlier detector and decision module. In the first module all the functionalities of data preprocessing such as feature selection, feature scaling, data encoding is performed. In the second module, data are cluster to the appropriate number of clusters with outlier detection. Thus, clustered data with outlier are then classified using Naïve Bayes classifier. The third module is a decision-making module.

 DOI: https://doi.org/10.30564/jcsr.v3i2.2922

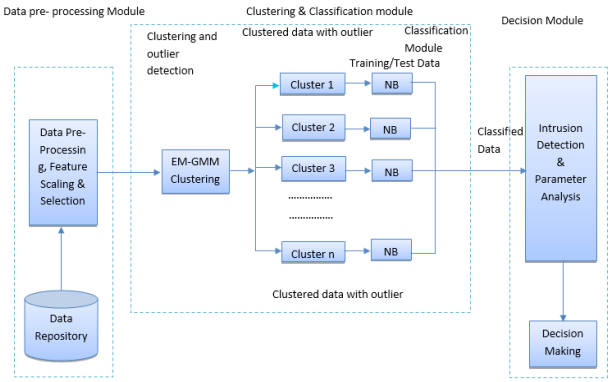

Figure 1. Conceptual Model Diagram of Proposed method

3.4 Algorithm

Algorithm 1 Data Clustering

Input: Dataset
Output: K number of clusters with outlier
Initialization:
1: Randomly choose μ_k, Σ_k, π_k
2: Specify k
3: Choose an initial random gaussian parameter θ
4: E step
5: Estimate the value of the latent variables Υ_k
6: Compute P (Z_i = k |X_i, θ);
7: M step
8: Update gaussian parameters μ_k, Σ_k, π_k
9: if
10: log-likelihood value converges
11: Stop
12: Else
13: Compute Υ_k and update μ_k, Σ_k, π_k
14: Assign data to appropriate cluster
15: End

3.5 Outlier Detection

Outer detection is the method of detecting the pattern in data that did not expect property. Following is the process of outlier detection:

• Randomly choose data in the dataset and measure the distance of the data to all other data. If the distance between the data and certain data is below the radius that we already set, assign that certain data as a neighbor, then assign the data and its neighbors as 1 cluster.

• Do as in previous step but the data is replaced by its neighbors. Neighbors of the neighbor are in the same cluster with previous data. Do this step until all detected neighbor is chosen.

• When all detected neighbor is chosen, construct a new cluster using data that has not been chosen. The new clusters are formed as in steps first and second.

The data that are not part of any cluster considered as an outlier.

Table 3.5 Outlier Statistics

Outlier = TRUE	32194	25.55%
Outlier = FALSE	125973	74.45%

3.6 Clustering

Clustering is a non-supervised approach to machine learning, but it can be used to maximize the precision of the supervised machine learning algorithm and cluster the data point into similar groups.

The purpose of clustering is to create dataset sub-population based on clustering results and to develop separate cluster classification models. Clustered membership can be considered as a feature in the classification and may have more details from these features. That improves the parallel processing capability of the model and manages data skews. That increases the accuracy of the classification.

3.6.1 EM Clustering

The expectation maximization (EM) clustering algorithm measures probabilities of cluster membership based on one or more distributions of probabilities. The goal of the clustering algorithm is then, given the (final) clusters, to maximize the overall likelihood or probability of the results.

Each gaussian j (j=1,2…..k) is defined in the EM clustering by its own μ and σ^2 as:

$$P_X\left(\frac{x}{\mu^j}, \sigma_j^2\right) = N(X; \mu^j, \sigma_j^2) = \frac{1}{\left(2\pi\sigma_j^2\right)^{\frac{d}{2}}} e^{\frac{-\left|x-\mu^j\right|^2}{2\sigma_j^2}} \quad (1)$$

Where,
μ is mean
σ is standard variable
x- μ is the distance between two points.
Each gaussian component has a mixture weight that indicates the likelihood.

3.7 Maximum Likelihood Estimation

In construction of a Bayesian classifier the class-conditional probability density functions need to be determined. The initial model selection can be done for example by visualizing the training data, but the adjustment of the model parameters requires some measure of goodness, i.e., how well the distribution fits the observed data. Data likelihood is such a goodness value. Assume that there is a set of independent samples X = {X_1, ... …X_N} drawn from

DOI: https://doi.org/10.30564/jcsr.v3i2.2922

a single distribution described by a probability density function P (x; θ) where θ is the PDF parameter list. The likelihood function can be written as:

$$L(X; \theta) = \prod_{n=1}^{N} p(x_n; \theta) \quad (2)$$

Equation (2) indicates the probability of X due to its distribution parameters θ. The goal is to calculate $\hat{\theta}$ which optimize the likelihood.

$$\hat{\theta} = \arg\max_{\theta} L\ (X; \theta) \quad (3)$$

This function is generally not explicitly maximized, but rather the logarithm as:

$$L(X; \theta) = \ln L(X; \theta) = \sum_{n=1}^{N} \ln p(x_n; \theta) \quad (4)$$

This is due to easier to handle logarithm function analytically. The limit can be identified analytically according to P (x; θ) by setting the derivatives of the log-like function to zero and θ resolution. A gaussian PDF can be used which leads to the estimation of intuitive mean and variance but the research approach is generally intractable. In this case, the iterative method, such as the EM algorithm, is used in practice [1]. Maximizing the likelihood in certain situations will lead to unique estimates, which is the main issue of highest probability methods. The function of classifying vector in K classes is recalled by Gaussian mixture model. If different classes are treated as distinct (i.e., class samples don't say anything about other courses), the k class-conditional PDF estimation problem can be divided into K separate estimation problems.

3.8 Gaussian Mixture Probability Density Function

In a single dimensional bell-shaped curve, the Gaussian probability density function is defined by two parameters; mean (μ) and variance (σ^2). But, for D dimensional space it is in matrix form as:

$$N(x; \mu, \Sigma) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} e^{[-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)]} \quad (5)$$

Where,

Σ is a matrix of covariance

μ is the mean vector.

Gaussian surfaces are μ-centered hyperellipsoids.

The gaussian mixture model (GMM) consists of a mixture of several Gaussian distributors, thus representing different subclasses within a class. The probability density fction is also known as the weighted sum of Gaussian.

$$P(x; \theta) = \sum_{c=1}^{C} \propto_c N(x; \mu_c, \Sigma_c) \quad (6)$$

were

α_c is the component weight c, $0 < \alpha_c < 1$ for all components, and PC c=1

α_c is the list of parameters whose value is equal to 1.

$$\theta = \{\alpha_1, \mu_1, \Sigma_1, ..., \alpha_C, \mu_C, \Sigma_C\} \quad (7)$$

defines a fundamental Gaussian density.

3.9 Basic EM Estimation

Suppose, X is all good features of sample and Y is all unknown features of sample, then the expectation (E) step of the EM algorithm is

$$Q(\theta; \theta^i) \equiv E_Y[\ln L(X, Y; \theta) \mid X; \theta^i] \quad (8)$$

Where θ^i is the previous distribution parameter estimate and θ is the distribution-descriptive estimation variable for the new estimate. L is the probability function which determines the likelihood of the data, including the unknown attribute Y marginalized in relation to the current distribution estimate defined by θ^i. Maximization step (M) is to optimize Q for θ and set steps are repeated until the conditions of convergence have been met.

$$\theta^{i+1} \leftarrow \arg\max_{\theta} Q\ (\theta; \theta^i) \quad (9)$$

It is proposed in [14] that the convergence parameters

$$Q(\theta^{i+1}; \theta^i) - Q(\theta^i; \theta^{i-1}) \leq T \quad (10)$$

with a correctly chosen T and in [18] that

$$||\theta^{i+1} - \theta^i|| \leq e \quad (11)$$

The EM algorithm begins with an initial distribution parameter guess, which ensures that the log-likelihood will increase on each iteration up to converge. Convergence results in a local or global limit, but it can also lead to specific estimates, especially for Gaussian mixture distributions with arbitrary matrices. The definition and implementation of the general EM algorithm for the Gaussian mixture model can be found in [6,1,4]. One of the main problems of EM algorithm is to initialize. The selection of theta (θ) defines where the algorithm converges or reaches the space parameter boundary that generates singular, insignificant results. Many solutions use random multiple starts or a clustering initialization algorithm [7]. The Gaussian mixtures implementation of the EM algo-

 DOI: https://doi.org/10.30564/jcsr.v3i2.2922

rithm as follows:

Let,

X is incomplete data

Y is knowledge of component that produced each sample X_n

For each X_n, a binary vector is assigned as:

$y_n = \{y_{n,1}, \ldots, y_{n,c}\}$

where,

$y_{n,c} = 1$, if component c or zero otherwise was generated in the sample.

The maximum probability of data log is

$$\ln L(X, Y; \theta) = \sum_{n=1}^{N} \sum_{c=1}^{C} y_{n,c} \ln\left(\propto_c p\left(x_n \mid c; \theta\right)\right) \quad (12)$$

The propose of E step is to calculate conditional expectancy for the whole log-like data, Q-function is produced by X and θ^i is current parameters estimation. As the whole data log-like function in L (X, Y; θ) is straightforward to the missing Y. Conditional expectation W simply needs to be determined and placed in ln L (X, Y; θ). That's why

$$Q(\theta, \theta^i) \equiv E \ln L(X, Y; \theta) | X, \theta^i = \ln L(X, W; \theta) \quad (13)$$

Where:

W elements have been defined as

$$\omega_{n,c} \equiv E [y_{n,c} | X, \theta^i] = Pr[y_{n,c} = 1 | x_n, \theta^i] \quad (14)$$

The estimate is determined using the Bayes law

$$\omega_{n,c} = \frac{\propto_c^i p(x_n \mid c; \theta^i)}{\sum_{j=1}^{C} \propto_j^i p(x_n \mid j; \theta^i)} \quad (15)$$

Where $\propto_c^i$ is the probability of a priori, and $\omega_{n,c}$ is the likelihood of posteriori of $Y_{n,c} = 1$ after observing X_n. In other words, "$\omega_{n,c}$ is the probability that X_n was produced by component c" [21].

If the M-step is used to evaluate the distribution parameters for C-component Gaussian mixture, with Arbitrary covariance matrices the following formulas will be used:

$$\propto_c^{i+1} = \frac{1}{N} \sum_{n=1}^{N} \omega_{n,c} \quad (16)$$

$$\mu_c^{i+1} = \frac{\sum_{n=1}^{N} x_n \omega_{n,c}}{\sum_{n=1}^{N} \omega_{n,c}} \quad (17)$$

$$\sum_{c}^{i+1} = \frac{\sum_{n=1}^{N} \omega_{n,c} \left(x_n - \mu_c^{i+1}\right)\left(x_n - \mu_c^{i+1}\right)^T}{\sum_{n=1}^{N} \omega_{n,c}} \quad (18)$$

previous numbers are now x θ^{i+1}. Unless the convergence criterion (Equations 10 or 11) is met, i ← i + 1 and Equations 15-18 new models are being tested again. [15]

weight α_c of the item is the sample portion of the element. The conditional PDF variable is estimated with the preliminary parameter estimates, and later the likelihood is determined for each sample point of c. The mean μ component is calculated in the same way as a covariant matrix Σ_c. The samples are evaluated according to the probability of the variable and the sample average and covariance matrix are calculated.

Table contains classification statistics, the number of instances transmitted into each cluster, and the proportion of instances from each cluster's total data.

Table 3.9 Clustered Instances

	No. of instances	% of instances
Cluster 1	45108	36%
Cluster 2	34025	27%
Cluster 3	13432	11%
Cluster 4	27394	22%
Cluster 5	6013	5%

3.10 Classifier

A classifier may adjust a number of parameters to the function. This is known as training. The samples in the training are labelled in supervised learning and the training algorithm tries to reduce the training set's classification error. Unsupervised learning does not label samples, but the training algorithm recognizes clusters and classes. The training samples are not also classified in reinforcement learning, but the training algorithm uses input to inform whether or not to identify a sample properly [40].

3.10.1 Bayesian Classification

Bayesian classification and its decisions are based on the probability theory and on the idea that the most likely or lowest risk i.e, expected cost is chosen. Suppose there is a classification task in which to assign functional vectors to K various classes. A vector function is labelled with $x = [X_1, X_2, \ldots\ldots, X_D]$ T. Where, D is the dimension of a vector. Probability that a feature vector x belongs to class ω_k is $p\left(\frac{\omega_k}{x}\right)$, and this is referred to as a posteriori

DOI: https://doi.org/10.30564/jcsr.v3i2.2922

probability. The vector's classification is based on the subsequent probabilities or decision risks determined from the probabilities. The conditional probability can be determined by Bayes formula as

$$P\left(\frac{\omega k}{x}\right) = \frac{P\left(x/\omega k\right)P\left(\omega k\right)}{P\left(x\right)} \quad (19)$$

where $P\left(\frac{X}{\omega_k}\right)$ is the probability density function of class ω_k in the feature space and $P(\omega_k)$ is the a priori probability. That gives the likelihood class before any characteristics are calculated. When previous probabilities are not known, they can be calculated in the training set according to the class proportions.

$$P(x) = \sum_{i=1}^{k} P\left(\frac{x}{\omega i}\right) P(\omega i) \quad (20)$$

It's just a factor in scaling to ensure that later probabilities are actual probabilities, that is, their sum is 1. Choosing the lowest retrograde likelihood class will illustrate the minimum pbability of error [1,4]. However, if the costs of making various types of error are not consistent, a risk function can be used which calculates the expected cost with the following probabilities and selects the lesser-risk class. The main problem in the Bayesian classification is the class-conditional density function $p\left(\frac{x}{\omega K}\right)$. The function defines the dispersion of feature vectors within a specific class, i.e., the class model. It is always unclear in reality, except for certain artificial classification activities. With a variety of methods, the distribution can be calculated in the training set.

3.11 Unit of Results

The model performance is calculated based on the following parameters and unit.

3.11.1 Accuracy of Classification

It is the proportion of correctly classified.

$$\text{Classification accuracy} = \frac{\frac{TP}{TN}}{\left(TP+TN+FP+FN\right)}$$

3.11.2 Sensitivity (True Positive Fraction)

It is the percentage of the number of properly identified attack.

$$\text{Sensitivity} = \frac{TP}{\left(TP+FN\right)}$$

3.11.3 Specificity (True Negative Faction)

It is the percentage properly categorized.

$$\text{Specificity} = \frac{TN}{\left(TP+FN\right)}$$

3.11.4 False Alarm Rate (FAR)

It is the percentage of the number of normal connections in correctly classified.

$$\text{False alarm rate (FAR)} = \frac{FP}{\left(TN+FP\right)}$$

3.11.5 Detection Rate (Precision)

It is the rate of detection of total anomaly from the total flow of packets in the network.

$$\text{Detection rate (DR)} = \frac{TP}{\left(TP \quad FP\right)}$$

Where,

True positive (TP) = Attacks that are correctly detected as attack.

True negative (TN) = Normal data that are correctly detected as normal.

False positive (FP) = Normal data that are incorrectly detected as attack.

False negative (FN) = Attack that are incorrectly detected as normal.

4. Results and Discussion

Based on obtained result, the overall accuracy in compared with different algorithms. The obtained result is illustrated in the following table.

Table 4. Result Comparison of Different Algorithm

Attack Class	K-NN	C4.5	SVM	DSSVM	K means with NB	Proposed method
Normal	98.3	97.0	97.7	98.4	74.11	97.48
DoS	97.0	96.8	97.2	97.2	86.05	81.65
Probe	79.4	84.3	86.1	87.5	92.48	97.13
R2L	6.5	3.0	7.2	6.3	32.02	95.17
U2R	11.8	4.4	9.2	3.1	19.0	73.66

From the above discussion, it is cleared that low frequency attack (probe, R2L, U2R) detection rate is improved in the integrated models. In proposed model this rate is significantly improved. Also, the detection rate for

 DOI: https://doi.org/10.30564/jcsr.v3i2.2922

normal class also improved in competitive ratio with the existing algorithms.

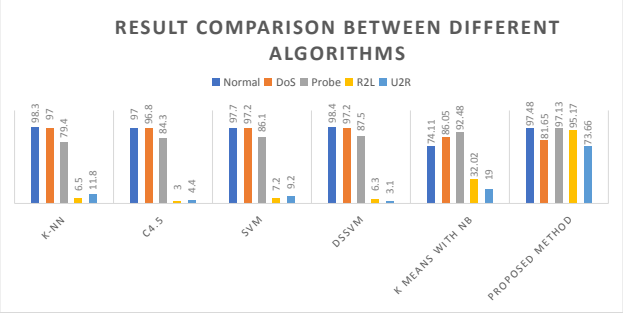

Figure 2. Result comparison between different algorithm with proposed model

From the above comparison chart with various models, overall performance is beaten by EM GMM with naïve Bays (proposed method) for low frequency attack i.e, R2L and U2R. Also, the performance for Prob is better than other models but the performance of DoS class is higher in other intrusion detection systems.

Except DoS, overall performance of proposed model is better than integration of K-means clustering with Naïve Bayes.

In this paper, we tried to simulate proposed model with various parameters with different ratio of training/testing model and calculate different matrices based on the obtained result. These metrics are objective measurements that are calculated mathematically defined algorithms. The comparison table for the experimental result is shown above in the table.

5. Conclusions

The research work observed with overall performance winner as integration of Expectation Maximization clustering with Naïve Bayes classifier for intrusion detection over Integration of K-Means clustering and Naïve Bayes classifier is considered to be best in terms of precision, sensitivity, specificity, and false alarm rate for the different types of attack class such as Probe, R2L, U2R and normal. It is shown that clustering plays a supportive role for classification by parallel computation so that the computation capacity of the model is improved. Since the whole dataset is clustered in a K number of clusters and compute parallelly, it can be used as real time/online computation with full efficiency computation on large data.

As the overall result of this model is significantly improved in different attack classes such normal, probe, R2L and U2R. But the other intrusion detection system has a higher detection rate for DoS attack.

Further improvement can be done in a number of ways. Firstly, the overall accuracy of DoS can be improve. Next improvement can be done in reducing the computation time at outlier detection.

References

[1] S. Varuna, Dr. P. Natesan "An Integration of K-Means Clustering and Naïve Bayes Classifier for Intrusion Detection." 2015 3rd international conference on signal processing, communication and networking " ICSCN. 978-1-4673-6823-0/15. 2015 IEEE.

[2] D. E. Denning, "An intrusion-detection model," IEEE Transactions on Software Engineering, vol. SE-13, no. 2, pp. 222-232, 1987.

[3] W. Parkand S. Ahn, "Performance Comparison and Detection Analysis in Snortand Suricata Environment," Wireless Personal Communications, vol.94, no.2, pp.241-252, 2016.

[4] R. T. Gaddam and M. Nandhini, "An analysis of various snort based techniques to detect and prevent intrusions in networks: Proposal with code refactoring snort tool in Kali Linux environment," in Proceedings of the 2017 International Conference on Inventive Communication and Computational Technologies, ICICCT2017, pp.10-15, India, March 2017.

[5] C.-T. Huang, R. K. C. Chang, and P. Huang, "Signal Processing Applications in Network Intrusion Detection Systems," EURASIP Journal on Advances in Signal Processing, vol. 2009, Article ID 527689, 2 pages, 2009.

[6] U. Adhikari, T. H. Morris, and S. Pan, "Applying Non-Nested Generalized Exemplars Classification for Cyber-Power Event and Intrusion Detection," IEEE Transactions on Smart Grid, vol. 9, no. 5, pp. 3928-3941, 2018.

[7] R. Taormina and S. Galelli, "A Deep Learning approach for the detection and localization of cyber-physical attacks on water distribution systems," Journal of Water Resources Planning & Management, vol.144, no.10, Article ID 04018065, 2018.

[8] F. Raynal, Y. Berthier, P. Biondi, and D. Kaminsky, "Honeypot forensics,"in Proceedings of the Proceedings from the Fifth Annual IEEE System, Man and Cybernetics Information Assurance Workshop, SMC, pp.22-29, USA, June 2004.

[9] W. J. Anand M. G. Liang, "A new intrusion detection method based on SVM with minimum within-class scatter," Security and Communication Networks, vol.6, no. 9, pp. 1064-1074, 2013.

[10] E. Kabir, J. Hu, H. Wang, and G. Zhuo, "A novel statistical technique for intrusion detection systems," Future Generation Computer Systems, vol. 79, pp. 303-318, 2018.

DOI: https://doi.org/10.30564/jcsr.v3i2.2922

[11] M. Gudadhe, P. Prasad, and K. Wankhade, "A new data mining based network intrusion detection model," in Proceedings of the 2010 International Conference on Computer and Communication Technology, ICCCT-2010, pp. 731-735, India, September 2010.

[12] S. T. Al-Janabi and H. A. Saeed, "A Neural Network Based Anomaly Intrusion Detection System," in Proceedings of the 2011 Developments in E-systems Engineering (DeSE), pp. 221-226, Dubai, United Arab Emirates, December 2011.

[13] K. D. Denatious and A. John, "Survey on data mining techniques to enhance intrusion detection," in Proceedings of the International Conference on Computer Communication and Informatics, pp. 1-5, 2012.

[14] Y.Guan, A. A. Ghorbani, and N. Belacel, "Y-means: A clustering method for intrusion detection," in Proceedings of the CCECE 2003 Canadian Conference on Electrical and Computer Engineering: Toward a Caring and Humane Technology, pp. 1083-1086, Canada, May 2003.

[15] H.-B. Wang, H.-L. Yang, Z.-J. Xu, and Z. Yuan, "A clustering algorithm use SOM and K-means in intrusion detection," in Proceedings of the 1st International Conference on E-Business and E-Government (ICEE'10), pp. 1281-1284, May2010.

[16] H. Gao, D. Zhu, and X. Wang, "A Parallel Clustering Ensemble Algorithm for Intrusion Detection System," in Proceedings of the 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES), pp. 450–453, Hong Kong, China, August 2010.

[17] Akashdeep, I. Manzoor, and N. Kumar, "A Feature Reduced Intrusion Detection System Using ANN Classifier," Expert Systems with Applications, vol. 88, pp. 249-257, 2017.

[18] Z. Muda, W. Yassin, M.N. Sulaiman, and N.I. Udzir, "Intrusion detection based on K-Means clustering and Naïve Bayes classification," in Proceedings of the 7th International Conference on Information Technology in Asia (CITA '11), pp. 1-6, IEEE, July 2011.

[19] M. Ishida, H. Takakura, and Y. Okabe, "High-performance intrusion detection using OptiGrid clustering and grid-based labelling," in Proceedings of the 11th IEEE/IPSJ International Symposium on Applications and the Internet, SAINT 2011, pp. 11-19, Germany, July 2011.

[20] H. Om and A. Kundu, "A hybrid system for reducing the false alarm rate of anomaly intrusion detection system," in Proceedings of the 2012 1st International Conference on Recent Advances in Information Technology, RAIT-2012, pp. 131-136, India, March 2012.

[21] S. A.R. Shah and B. Issac, "Performance comparison of intrusion detection systems and application of machine learning to Snort system," Future Generation Computer Systems, vol. 80, pp. 157-170, 2018.

[22] J. S. Yi., X. song, H. Wang, J.-J. Han and Q.-H. Li, "A clustering-based method for unsupervised intrusion detections." Pattern recognition letters 27, no. 7 (2006): 802-810.

[23] Oh, S. Hyum, and W. S. Lee. "An anomaly intrusion detection method by clustering normal user behavior." Computer and security 22, no.7 (2003): 596-612.

[24] C.F. Tasi and C.Y. Lin 2010. "A triangle area-based nearest neighbors approach to intrusion detection." Pattern recognition, 43(1): p.222-229.

[25] W. Gang, H. Jinxing and M. Jian 2011. "A new approach to intrusion detection using artificial neural networks and fuzzy clustering. Expert systems with applications, 376: p.6255-6232.

[26] Shaohua, D. Hongle, W. Naiqi, Z. Wej and S. Jiangyi, 2010. "A cooperative network intrusion detection based on fuzzy SVMs. Journals of networks, 5: p. 475-483.

[27] F. Amiri, F. Mohammad, R. Y. Caro, L. Azadeh, S. and Y. Nasser 2011. "Mutual information-based feature selection for intrusion detection system." Journal of network and computer applications, 34: p.1184-1199.

[28] S.J. Horng 2011 "A novel intrusion detection system based on hierarchical clustering and support vector machines. Expert systems with applications. 38(1) :P.399-408.

[29] J. Huang, J. Lu, C. X. Ling, "Comparing Naïve Bayes, Decision trees, and SVM with AUC and accuracy." The third international conference on data mining 2003.

[30] R. Chitrakar and H. Chauanhe "Anomaly detection using support vector machine classification with K-medoids clustering". 978-1-4673-2590-5/12. 2012 IEEE.

[31] F. Kelly. "The mathematics of traffic in networks." The Princeton companion to mathematics, 1(1):862-870, 2008.

[32] Z.Muda, W. Yassin, M.N. Sulaiman, N.I. Udzir "K-Means clustering and Naïve Bayes classification for intrusion detection." Journal of IT in Asia Vol 4 (2014).

[33] V.-E. Neagoe, V.C.-Berbentea "Improved Gaussian mixture model with Expectation Maximization for clustering of remote sensing imagery." 978-1-5090-

 DOI: https://doi.org/10.30564/jcsr.v3i2.2922

3332-4/4/16. 2016 IEEE.

[34] A. Reddy, M. Ordaway-West, M. Lee, M. Dugan, J. Whitney, R. Kahan, B. Ford, J. Muedsam, A. Henslee, & M. Rao "Using Gaussian Mixture models to detect outliers in seasonal univariate network traffic." DOI 10.1109/SPW.2017.9 IEEE computer society 2017.

[35] E. A. Shams and A. Rizaner, "A novel support vector machine based intrusion detection system for mobile adhoc networks," Wireless Networks, pp.1-9, 2017.

[36] W. Shang, L. Li, M. Wan, and P. Zeng, "Industrial communication intrusion detection algorithm based on improved one-class SVM," in Proceedings of the World Congress on Industrial Control Systems Security, WCICSS 2015, pp. 21-25, UK, December 2015.

[37] T. Jan, "Ada-Boosted Locally Enhanced Probabilistic Neural Network for IoT Intrusion Detection," in Proceedings of the Conference on Complex, Intelligent, and Software Intensive Systems, pp. 583-589, Springer, 2018.

[38] O. Osanaiye, K.-K. R. Choo, and M. Dlodlo, "Distributed denial of service (DDoS) resilience in cloud: review and conceptual cloud DDoS mitigation framework," Journal of Network and Computer Applications, vol.67, pp.147-165, 2016.

[39] H. Li, "Research and Implementation of an Anomaly Detection Model Based on Clustering Analysis," Journal of Beijing Information Science & Technology University, pp. 458-462, 2010.

[40] R. O. Duda, P.E. Hart, and D.G. Stork. Pattern Classification. John Wiley & Sons, Inc., 2nd edition, 2001.

DOI: https://doi.org/10.30564/jcsr.v3i2.2922