Soft Computing
https://doi.org/10.1007/s00500-018-3621-z

FOCUS

Improved Relevance Vector Machine (IRVM) classifier for Intrusion Detection System

E. M. Roopa Devi[1] · R. C. Suganthe[2]

Abstract

Intrusion detection is the most significant research area in online applications to avoid intrusion activities. The foremost goal of the present research is to use the Relevance Vector Machine which can recognize extract intrusion activities involved in the Intrusion Detection System. Classification and feature selection are implemented by Improved Relevance Vector Machine and Gaussian Firefly Algorithm, respectively. The proposed work contains three phases such as preprocessing, feature selection and classification, and it would increase the classification accuracy. Preprocessing uses the technique Kalman filtering which focuses on missing values in the given Knowledge discovery in databases. Gaussian Firefly Algorithm selects the most relevant and optimal features, thereby increasing overall execution speed. Then Improved Relevance Vector Machine identifies intrusion attacks efficiently by extracting more relevant vectors and thus classifying maximum likelihood values. The experimental result concludes that Improved Relevance Vector Machine algorithm provides greater performance in terms of precision, recall, specificity and accuracy.

Keywords Intrusion Detection System (IDS) · Gaussian Firefly Algorithm (GFA) · Improved Relevance Vector Machine (IRVM)

1 Introduction

In recent days, there is a massive development in utilizing internet for social networking, for instance, video conferencing, instant messaging, and so on. Bank transactions, e-commerce, healthcare, and frequent former services. Internet applications require an acceptable point of security and confidentiality. The computers are rather under attacks in addition to susceptible to various threats. There is a raising accessibility of tools and tricks for attacking and interfering with networks. An intrusion is one sequence of activities, which endangers the protection needs, for instance, confidentiality, integrity and availability of a computer/network resource, for instance, file systems, user accounts and system kernels. Intruders have endorsed themselves and designed innovative tools which aid a number of network attacks. Therefore, efficient approaches for Intrusion Detection (ID) have turned out to be a challenging requirement to safeguard the workstation from intruders. Generally, there are two classes of Intrusion Detection Systems (IDS): abuse detection and abnormality detection. Numerous commercial IDS use the misuse technique where group of intrusions identified and is part of system as signatures as presented in McHugh (2001).

Data mining methods are utilized for misuse in addition to anomaly intrusion detection. Misuse denotes the identified attacks and dangerous actions, which exploit the well-known sensitivities of the system. According to misuse detection, every occurrence in a data set is denoted as simple or malicious and this learning approach is mentioned as the data considered. Anomaly detection is known as a normal action which is denoted as an intrusion. A benefit of misuse detection system methods is their better accuracy in identifying well-known attacks as well as their deviation. IDS are known as the area, in which IDS data are gathered from numerous

Communicated by A.K. Sangaiah, H. Pham, M.-Y. Chen, H. Lu, F. Mercaldo.

✉ E. M. Roopa Devi
roopasen5@gmail.com

R. C. Suganthe
suganthe_rc@kongu.ac.in

[1] Department of Information Technology, Kongu Engineering College, Perundurai, Tamil Nadu, India

[2] Department of Computer Science Engineering, Kongu Engineering College, Perundurai, Tamil Nadu, India

Published online: 16 November 2018

sources such as host data and network log data. The data cannot be examined easily as the network traffic is hefty, which offers to the requirement of utilizing IDS in designing diverse data mining methods in ID as suggested by Han and Kamber (2011).

In data mining, Support Vector Machine (SVM) is an effective classification method, but its lengthy training time restricts its usage. As per Stolfo et al. (2001), it would get through years to train SVM on a data set around one million records. With the aim of improving its training performance, numerous techniques are introduced by Upadhyaya and Jain (2013) arbitrary selection or guesstimate of the trivial classifier. In contrast, these methods are even now not possible with big data sets where even numerous examinations of complete data set are very costly to carry out, or they bring about the loss via distortion of any advantage through the use of SVM as proposed by Yu et al. (2003).

Hu et al. (2014) have presented the AdaBoost classifier; the weak classifiers are built for every own feature component, for continuous and categorical attributes, with the intension that the associations among these features are naturally controlled of any compulsory change amid continuous attributes in addition to categorical attributes. Novel methods are developed for local ID. A Particle Swarm Optimization (PSO) and SVM-based algorithm help to classify the IDS.

In this present research, effective preprocessing and feature selection techniques are proposed to improve the intrusion classification accuracy rates. The proposed system comprises three phases: (1) preprocessing, (2) feature selection and (3) classification. Applying Kalman filtering algorithm replaces the missing values for the Knowledge discovery in databases (KDD) dataset. GFA accomplishes feature selection in the second phase. It generates objective function value by increasing the accuracy and provides more optimal solutions. In third phase, IRVM classification algorithm produces efficient detection results for the KDD dataset.

IDS plays a vital role in various applications such as credit fraudulent detection, anomaly detection in networking, denial-of-service attack detection.

2 Related work

Xiang et al. (2004) have developed a Multiple-level Tree Classifier for IDS and for raising the accuracy rate. Multilevel Tree Classifier is effective in known malicious occurrences while for unknown susceptibility it provides flat detection rate. Peddabachigari et al. (2007) have presented a framework of disturbance discovery system uniting Decision Tree and Support Vector Machine (DTSVM) classification methods which supports the greater detection rate.

Panda and Patra (2008) have matched diverse data mining methods for IDS and identified that accuracy as well as execution of Naïve Bayes classifier for the entire classes is superior compared to the accuracy obtained in diverse Decision Tree algorithm; however, on the other hand Decision Tree is capable of identifying unfamiliar intrusions in contrast to Naïve Bayes classification algorithm.

Ektefa et al. (2010) have presented SVM and Decision Tree data mining method for intrusion detection in network. They have combined C4.5 algorithm and SVM by using investigational outcomes. C4.5 algorithm contains improved performance with regard to true positive rate as well as false alarm rate compared to SVM, while SVM does well for U2R attack.

Hu et al. (2008) have proposed rapid machine learning-based intrusion detection techniques containing greater accuracy rates and lesser false alarm rates. Adaptable preliminary weights and a modest technique for evading over fitting are accepted for enhancing the attainment of the AdaBoost algorithm.

Gao et al. (2009) have presented scattered IDS framework. The above-mentioned framework comprises the individual and global representation. Particularly, the typical unit is derived from Gaussian Mixture Model emanated from the online AdaBoost algorithm, and this would be skilled, connected and take minimum traffic to communicate amid local units.

Yang (2009) have presented a novel Firefly Algorithm (FA) for multimodal optimization applications. Replications and outcome show that the FA offers greater performance compared to the already available PSO algorithm.

Bishop and Tipping (2000) have dealt with the Relevance Vector Machine for increasing the classification accuracy. RVM attains equivalent identification accuracy to the SVM. It also offers a complete predictive distribution and needs some kernel functions to a certain extent.

Li et al. (2018a) have introduced the current problems of IoT in network security and pointed out need for ID. Several kinds of ID technologies are presented, and their application on IoT architecture is analysed.

Li et al. (2018b) have proposed a CNN-based technique with the help of synergetic neural networks. The technique at the initial stages sets in the form of a watermark signals into the block Discrete Cosine Transform (DCT) component.

Mabu et al. (2011) have introduced the history and current situation of IDS, expounded types of IDS and the framework of general intrusion detection and conversed the types of ID technology entirely.

Sangaiah et al. (2018b) have proposed dimensionality reduction using clustering with the help of K-means algorithm.

Sangaiah et al. (2018a) have presented Hybrid Fuzzy multicriteria for decision-making purpose.

Sangaiah et al. (2015) have presented genetic learning algorithm for global software development. Medhane and

Sangaiah (2017) have proposed search space-based multiobjective evolutionary algorithm (SSMOEA) for multiobjective optimization problems.

3 Proposed methodology

3.1 Preprocessing using Kalman filtering

The proposed IRVM with GFA IDS is categorized into three modules: preprocessing, feature selection and classification. Figure 1 describes architecture of the IRVM with GFA ID network system. In this section, KDD dataset is taken into account along with Kalman filtering for preprocessing.

In this proposed system, data preprocessing is the first module which is an analysis of data by using Kalman filtering approach. The numerical values and categorical data are included in KDD dataset. Generally, the numerical values miss any tuple, and hence, the proposed Kalman filtering algorithm mainly focuses on to handling the missing values in the given KDD dataset.

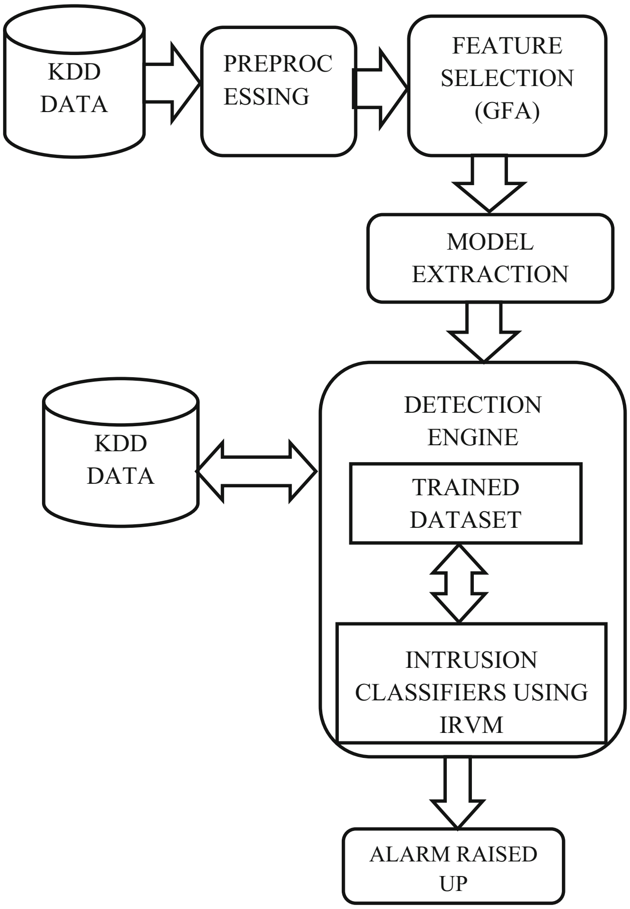

Fig. 1 Proposed system of hybrid IDS

Kalman filter is to discover the missing values by deriving the equation and covariance error. Hence, classification accuracy of the given KDD dataset will get improved.

Depending the known Kalman filter approach, the lost data are projected. Prediction is carried out with the aid of the available data for the period of updating methods for approximation of coefficients of the Kalman filter when no update is carried out while the data are lost for Kalman filter as proposed by Arai (2013). In the meantime, update is carried out utilizing the mean of the data in firm even for the missing data.

Algorithm

1. Prediction

$$A_{s|s-1} = FA_{s-1|s-1} \tag{1}$$

$$P_{s|s-1} = FP_{s-1|s-1}F^T + Q \tag{2}$$

2. If the data is missing

Update filter

$$\begin{cases} K_s = P_{s|s-1}H^T(HP_{s|s-1}H^T + R)^{-1} \\ A_{s|s} = A_{s|s-1} + K_s(B_s - HA_{s|s-1}) \\ P_{s|s} = (I - K_sH)P_{s|s-1} \end{cases}$$

3. If the data is existing

$$\begin{cases} K_s = P_{s|s-1}H^T(HP_{s|s-1}H^T + R)^{-1} \\ A_{s|s} = A_{s|s-1} + K_s(B_s - HA_{s|s-1}) \\ P_{s|s} = (I - K_sH)P_{s|s-1} \end{cases}$$

4. Filter

$$\begin{cases} L_s = P_{s|s}F^Tp^{p-1}_{s+1|s} \\ A_{s|s} = A_{s|s} + L_s(A_{s+1|T} - A_{s+1|s} \\ P_{s|T} = P_{s|s} + L_s(P_{s+1|T} - P_{s+1|s})L_s^T \end{cases}$$

5. The formula given below calculates

Covariance matrix

$$\begin{cases} A_{s|j} = E(A_s|A_j) \\ P_{s|j} = E[(A_s - A_{s|j})^T] \end{cases}$$

where the A_s refers to the state vector comprising of the terms of concern for the system (for example, position, velocity and heading). F stands for the state transition matrix that uses the outcome of every system state parameter. P_s is covariance matrix, H^T refers to the transformation matrix which maps the state vector parameters onto the measurement domain, and K_t is Kalman filter. The information from the predictions as well as measurements is united to offer the finest probable approximation location of train. It substitutes the missing values by anticipated values in the given KDD dataset. Recovering technique of losing data dependent on the Kalman filter presents the formation of fine-resolution intrusion data with the aforesaid technique based on a Kalman filter.

3.2 Feature selection using GFA

The purpose of the GFA is to improve the optimal feature selection process. The Firefly Algorithm produces slow process for high-dimensional dataset, and it is not efficient in finding the global optimal solution. Hence GFA implements Gaussian Distribution (GD) in increasing the optimal solutions as proposed by Farahani et al. (2011). This Gaussian algorithm uses three behaviours for enhancing the performance of FA.

- The initial behaviour an adaptive step length, which modifies random step length by the time.
- Secondly, the individual behaviour or guided movement, which guides the random movement in the direction of global best.
- Finally, a social behaviour which alters the position of every firefly dependent on a Gaussian Distribution (GD).

Adaptive step length

Firefly movement step length is a permanent value in standard Firefly Algorithm (FA). Due to the permanent step length, the algorithm would ignore improved local search abilities, and sometimes, it traps into numerous local optima. It is healthier that FA searches the space all over in the first iteration and in the final iteration and uses the specific place to excerpt better solutions.

Directed movement

In standard FA, firefly evolves is based on light intensity and matching it amid each pair of fireflies. When there is no confined best in every firefly's neighbourhood, it moves in the direction of the finest solution and makes improved position for every firefly for the subsequent repetition and it would get more near to the global best.

Social behaviour

Random walk is known as a random method that encompasses taking a sequence of successive random steps. Due to these advantages, the present research uses GFA for clustering anonymized data.

3.2.1 Feature selection using Gaussian Firefly Algorithm (GFA)

The three most significant classification processes for features are given below:

(1) The entire fireflies are unisex. Therefore, one attribute data matrix would be drawn to other attribute data matrix, not considering their sex as suggested by Nayak et al. (2015).

(2) Attractiveness is relative to the firefly brightness. Hence, for any two flashing attribute data matrices (fireflies), a smaller amount of bright one would travel towards the brighter one. It would check if it has no brighter one compared to a certain firefly, and would travel arbitrarily.

(3) The brightness of an attribute in the data matrix (firefly) influenced by the landscape of the similarity and dissimilarity represents the objective function.

Table 1 Simulation parameters for FA algorithm

Parameters	Values
α (alpha)	0.2
β_0 (beta$_0$)	0.3
γ (gamma)	0.2
Iterations	20

Gaussian Distribution (GD) enhances the performance of Firefly Algorithm, and this algorithm uses three behaviours. The initial behaviour is basically an adaptive step length, which modifies random step length by time, and the second one is personal behaviour or directed movement, which directs random movement for the overall best. The final behaviour is known as a social behaviour, which alters the locations of every firefly based on a Gaussian distribution.

The steps involved in GFA are described in Table 1. Simulation parameters for FA algorithm are given in Table 1. These parameter values are chosen before start of GFA algorithm.

Random walk is an arbitrary process which comprises a sequence of continuous random steps. At this point, it computes the step range or length in a random walk. When the step extent adopts the GD, the random walk turns out to be the Brownian motion. In regard to all the attribute data matrix values, it exploits random walk conceptions to shift every one of the agents depending on a GD as in Eq. (3):

$$p = f(\mathrm{dm}|\mu, \delta) = \frac{1}{\delta\sqrt{2\pi}} \mathrm{e}^{-(\mathrm{dm}-\mu)^2/2\delta^2} \tag{3}$$

where x represents an error amid most excellent solution and fitness value of multiple data matrix (firefly) i is represented in Eq. (4).

$$x = f(g_{\text{best}}) - f(\mathrm{dm}_i) \tag{4}$$

In Eq. (3), μ represents Mean and δ indicates Standard Deviation (SD). As a result of utilizing Standard Normal Distribution (SND), Mean is fixed to $\mu = 0$ and $\delta = 1$. The social behaviour of fireflies is given by

$$\mathrm{dm}_i = \mathrm{dm}_i + \alpha * (1 - p) * \mathrm{rand}() \tag{5}$$

where α in Eq. (5) represents firefly parameter which is fine-tuned by adaptive parameter approach as suggested by Nasiri and Meybodi (2016). In this proposed GFA, data clustering process depends on the random walk, all the fireflies from their own data holder matrix i are moved to finest result striking (brighter) depending on the location for every firefly for subsequent repetition, and they find added close to overall best as indicated in Eq. (4). The fitness value gives accurate value that shows the optimal detection of intrusion features. The optimal features are used to discover the accuracy between all features. The intrusion detection accuracy enhances by increasing the brightness values. This accuracy value estimates the intrusion class for the given KDD cup dataset effectively.

$$\text{Accuracy} = \frac{(\text{True positive} + \text{true negaive})}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{6}$$

Equation (6) represents the features as normal or intrusion attacks. When attack is predicted as attack, it is true positive (TP), when attack is predicted as normal, it is false negative (FN), and when normal is predicted as attack, then it is false positive (FP) and it gives result which is not both normal and intrusion features.

```
1. Initialize algorithm parameters: Objective function of $f(x)$, where $x = (x_1, \ldots, x_d)^{\mathrm{T}}$
2. Create the initial population of fireflies or $x_i (i = 1, 2, \ldots, n)$
3. Characterize light intensity of $I_i$ at $x_i$ via $f(x_i)$ from Eq. (4)
4. While (t < MaxGen)
    4.1 For $i = 1$ to $n$(all $n$ fireflies);
    4.2 For $j = 1$ to $n$(all $n$ fireflies)
    4.3 If $(I_j > I_i)$ move firefly $i$ towards $j$; end if
    4.4 Attractiveness changes with distance '$r$' via Exp $[-r2]$
    4.5 Estimate new solutions and revise light intensity;
    4.6 End for $j$;
    4.7 End for $i$;
5. Rank the fireflies and identify the current best cluster
6. Describe normal distribution
7. For $k = 1, \ldots n$ all $n$ fireflies
8. Obtain random data records from GD implemented $(o)$ for the chosen data matrix value
9. Estimate new solution (new solution $(k)$)
10. If new (new solution($k$ < solution($i$))&&(new solution($k$) < last solution($k$))
11. Shift features towards current best
12. End if
13. End for $k$;
14. End while;
15. Post-process results and visualization
16. End procedure;
```

3.3 Classification using IRVM algorithm

In mathematical field, a Relevance Vector Machine (RVM), a machine-based learning method, uses Bayesian inference to get parsimonious solutions for regression and probabilistic classification as proposed by An (2016). The RVM contains a similar functional form to the SVM, and it also gives probabilistic classification. The Gaussian parameter identifies the intrusion features and categorizes features using IRVM.

Posterior distributions of the major amount of weights are projected around zero. Training vectors related to the nonzero weights are the 'relevance vectors'.

For the provided data set of input point pairs $\{X_n, t_n\}_{n=1}^{N}$

$$t_n = y(x_n; W) + \epsilon_n(m) \tag{7}$$

where ϵ_n are models from some missing method that is presumed to be mean zero Gaussian with variance σ^2. So, $p(t_n|x) = N(t_n|y(x_n)\sigma^2$.

In classification process, a training sample set $\{x_n, t_n\}_{n=1}^{N}$ where $x \in R^D$ is the training sample and D is featured observations, and $t \in \{1 \ldots C\}$ where C as class labels. It can be expressed as $X \in R^{N \times D}$ from which the training kernel can be derived as $K \in R^{N \times N}$ based on a dataset dependent on kernel function. t_i denotes the testing sample label and $t_i = y_i + \varepsilon_i$, where $y_i = w^{\mathrm{T}}\varphi(x_i) = \sum_{j=1}^{N} w_j K(x_i, x_j) + w_0$ is the model of classification prediction.

The training kernel takes the past knowledge over the dataset. Each row k_n of the kernel K states how associated, based on the chosen kernel function, is the observation n to the others of the training set. The learning process comprises with the knowledge of the model parameter $W \in R^{N \times C}$, which by the quantity $W^{\mathrm{T}}K$ performs as a voting system to state which associations of the data are significant for the present model to contain suitable discriminative properties.

Let the training sample sets be autonomous and symmetrically distributed; the observation of vector t follows the subsequent distribution

$$p\left(t|, w, \sigma^2\right) = (2\pi\sigma^2)^{-\frac{N}{2}} \exp\left[-\frac{1}{2\sigma^2}||t - \varphi w||^2\right] \tag{8}$$

where φ is the vector expressed as

$$\varphi = \begin{pmatrix} 1 \; k(x_{1,}x_1) & \ldots\ldots & k(x_{1,}x_N) \\ \vdots \; \vdots & \ldots\ldots & \vdots \\ 1 \; k(x_{N,}x_1) & & k(x_{N,}x_N) \end{pmatrix}$$

The RVM utilizes sample label t to foresee the testing sample label t_* provided by

$$p(t_*|t) = \int p\left(t_*|w, \sigma^2\right) p(w, \sigma^2|t) \mathrm{d}w \mathrm{d}\sigma^2$$

To create the value of main components of the weight vector w zero and to decrease the computing work of the kernel function, the weight vector w is related to extra ordering. If w_i follows a distribution with a mean value of zero and a variance of α_i^{-1}, the mean $w_i \sim N\left(0, \alpha_i^{-1}\right)$, $w|a = \prod_{i=0}^{N} p(w_i|a_i)$, in which a is a hyperparameter vector of the prior distribution of the weight vector w.

$$p(t_*|t) = \int p\left(t_*|w, a, \sigma^2\right)p(w, a, \sigma^2|t)\mathrm{d}w\mathrm{d}a\mathrm{d}\sigma^2 \tag{9}$$

$$p\left(t_*|w, a, \sigma^2\right) = N(t_*|y(x_*; w)), \sigma^2 \tag{10}$$

As $p\left(w, a, \sigma^2|t\right)$ could not be attained by an integral, it should be solved with the help of a Bayesian formula, provided by

$$p\left(w, a, \sigma^2|t\right) = p\left(w|a, \sigma^2, t\right)p(a, \sigma^2|t) \tag{11}$$

$$p\left(w|a, \sigma^2, t = \frac{p\left(t|w, \sigma^2\right)p(w|a)}{p(t|a, \sigma^2)}\right) \tag{12}$$

The integral of the product of $p\left(t|a, \sigma^2\right)$ and $p(w|a)$ is given by

$$p(t|a,) = (2\pi)^{-N/2}|\Omega|^{-1/2}\exp\left(-\frac{t^{\mathrm{T}}\Omega^{-1}t}{2}\right) \tag{13}$$

$$\Omega = \sigma^2 I + \varphi A^{-1}\varphi^{\mathrm{T}}, A = \mathrm{diag}\left(a_{0,}a_1, \ldots a_N\right), \tag{14}$$

$$p\left(w|a, \sigma^2, t\right) = (2\pi)^{-\frac{N+1}{2}}|\Sigma|^{-\frac{1}{2}}\exp\left(-\frac{(w-u)^{\mathrm{T}}(w-u)}{2}\right) \tag{15}$$

$$\Sigma = \left(\sigma^{-2}\varphi^{\mathrm{T}}\varphi + A\right)^{-1} \tag{16}$$

$$u = \sigma^{-2}\Sigma\varphi^{\mathrm{T}}t \tag{17}$$

The solution is estimated by utilizing the maximum likelihood technique, denoted by

$$(a_{\mathrm{MP}}, \sigma_{\mathrm{MP}}^2) = \mathrm{argmax}_{a,\sigma^2}\, p(t|a, \sigma^2) \tag{18}$$

The iterative process of a_{MP} and σ_{MP}^2 is as follows

$$\begin{aligned} a_i^{\mathrm{new}} &= \frac{\gamma_i}{\mu_i^2} \\ (\sigma^2)^{\mathrm{new}} &= \frac{||t - \varphi\mu||^2}{N - \Sigma_{i=0}^N \mu_i} \\ \gamma_i &= 1 - a_i \Sigma i, i \end{aligned} \tag{19}$$

where $\Sigma i, i$ refers to ith component on the diagonal of Σ and the primary value of a and σ^2 determines the means of the approximation of a_{MP} and σ_{MP}^2 by constantly bringing up to date with the help of formula (19). Subsequently adequate repetitions, most of a_i is a_i are close to perpetuity, the value of the equivalent parameters in w_i would be zero, and other a_i values are close to set. The ensuing specifications *xi* of *ai* are then denoted as the relevance vector. Thus, the improved RVM to classify the intrusion types for the given KDD cup dataset accurately. The RVM classifier is used to match most the relevant features, providing a different classifier based only on a subset of the original features. IRVM increases the classification accuracy by using kernel function. By measuring the maximum likelihood samples, it results in accuracy of intrusion detection.

4 Experimental result

The KDD dataset as presented by UCI KDD Archive (1999) is used for testing the algorithms. This dataset is the most trustable and also plausibly benchmarked dataset which is used to assess network intrusion detection algorithms. MATLAB simulation tool evaluates set-up of KDD dataset. In KDD dataset, there are 41 features comprising nine categorical features and 32 continuous features extracts for every network connection. Attacks in the dataset belong to the subsequent four key types.

(1) *DOS* denial of service.
(2) *R2L* unauthenticated access from a remote machine, e.g. guessing password.
(3) *U2R* unauthenticated access from local to remote machine (root) privileges.
(4) *Probe* surveillance and other probing, e.g. port scanning.

All the four types of attack encompass certain low-quality attack kinds. The test dataset comprises certain abnormal patterns which do not survive in the training dataset. The statistics of normal connections and every kind of abnormal patterns in the training and test datasets are stated in KDD dataset. GFA algorithm has been selected with 18 important features in Table 2.

4.1 Accuracy

Accuracy is the general precision of the model and computed as the summation of the original classification parameters $\left(T_p + T_n\right)$ isolated by the overall number of classification parameters $\left(T_p + T_n + F_p + F_n\right)$

$$\mathrm{Accuracy} = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \tag{20}$$

Table 2 Result of feature selection

Serial number	Feature number	Type
1.	2	Duration: continuous
2.	3	protocol_type: symbolic
3.	4	service: symbolic
4.	6	src_bytes: continuous
5.	7	dst_bytes: continuous
6.	12	num_failed_logins: continuous
7.	14	num_compromised: continuous
8.	24	count: continuous
9.	25	srv_count: continuous
10.	31	diff_srv_rate: continuous
11.	32	srv_diff_host_rate: continuous
12.	33	dst_host_count: continuous
13.	34	dst_host_srv_count: continuous
14.	35	dst_host_same_srv_rate: continuous
15.	36	dst_host_diff_srv_rate: continuous
16.	39	dst_host_serror_rate: continuous
17.	40	dst_host_srv_serror_rate: continuous
18.	42	dst_host_srv_rerror_rate: continuous

Figure 2 illustrates the of accuracy comparison for the KDD intrusion dataset. The number of samples is engaged along the x-axis, and the parameter accuracy is plot along the y-axis. Experimental result proves that the AdaBoost and AdaBoost-based SVM + PSO reveal lower accuracy results while IRVM + GFA reveal higher classification accuracy results.

4.2 Precision

Precision is the proportion of the true positives in contradiction to true and false positive results for intrusion and real features. It is defined as

$$\text{Precision} = \frac{T_p}{T_p + F_p} \tag{21}$$

In Fig 3, the graph reveals the precision metric comparison for the KDD cup intrusion dataset. The number of samples is plotted along the x-axis, and the parameter precision value is plotted along the y-axis. Experimental result shows that the AdaBoost and AdaBoost SVM + PSO have lower precision results. The proposed IRVM + GFA gives the higher precision metric results. Thus, the result shows that introduced IRVM + GFA accomplishes better result compared to other algorithm.

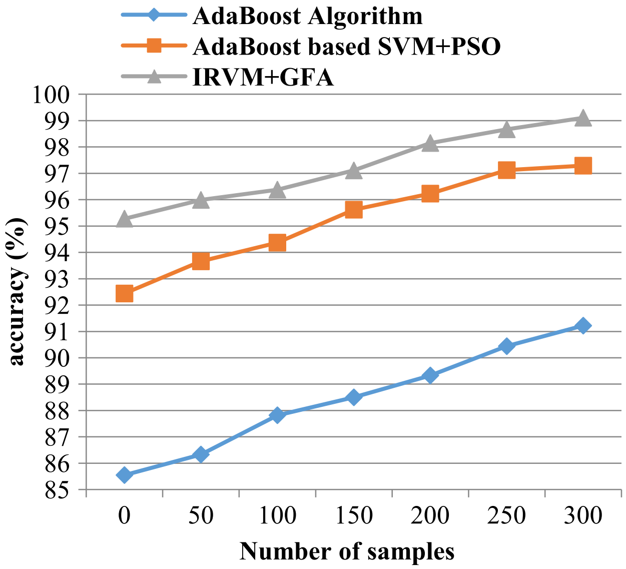

Fig. 2 Accuracy

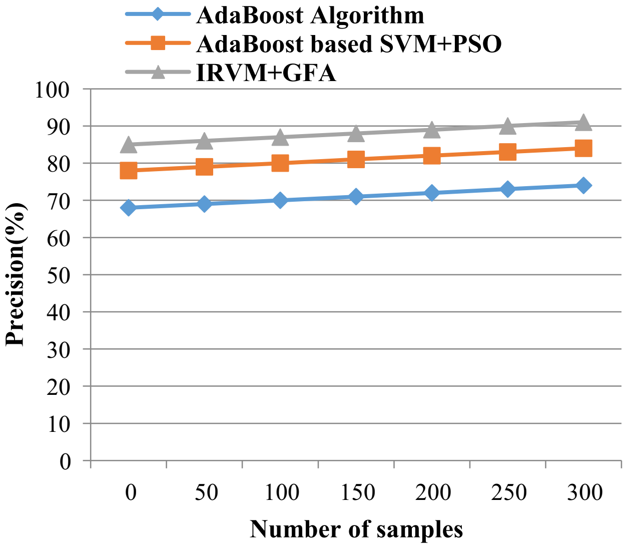

Fig. 3 Precision

4.3 Recall

Recall value is the proportion of the true positive by true positive prediction and false negative. Generally, it denotes

$$\text{RECALL} = \frac{\text{True positive}}{\text{True positive} + \text{False negative}} \tag{22}$$

Recall is known as sensitivity.

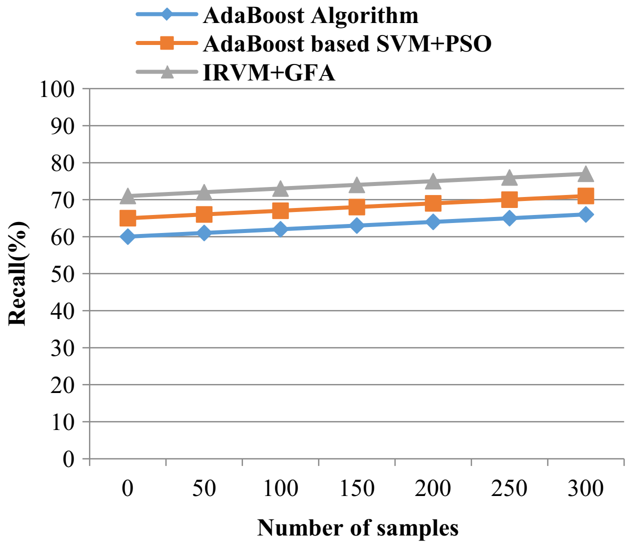

Fig. 4 Recall

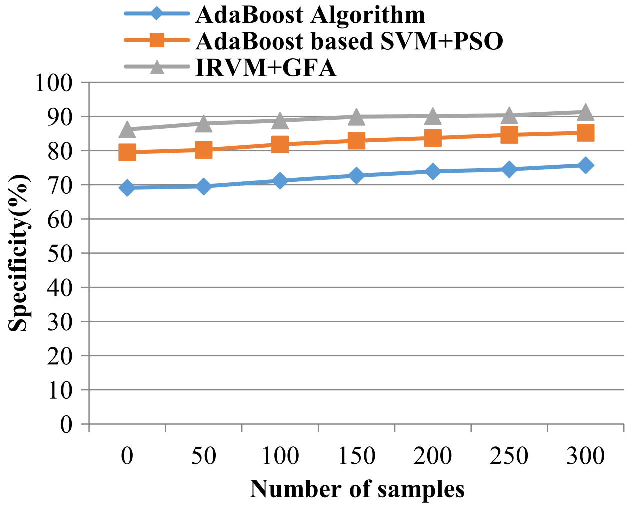

Fig. 5 Specificity

The comparison of recall value between the existing AdaBoost, AdaBoost SVM + PSO and proposed IRVM + GFA is represented in Fig. 4. The recall value gets increased as the figure of samples increases. From the above figure it is proposed that the IRVM+GFA does more willingly than the previous techniques in terms of recall.

4.4 Specificity

Specificity assesses the proportion of the real negatives that relate with negative subject class, which denotes intrusion feature as real feature and real feature as intrusion feature.

$$\text{Specificity} = \frac{T_n}{T_n + F_p} \tag{23}$$

Figure 5 depicts the results of number of samples Vs specificity for existing AdaBoost, AdaBoost SVM + PSO and the proposed IRVM + GFA method. From the simulation results, the proposed IRVM + GFA has higher specificity rather than the existing AdaBoost and AdaBoost-based SVM + PSO methods.

5 Conclusion and future work

In the present research, IRVM with GFA method enhances the intrusion detection accuracy. The proposed system varies from the existing system in using SVM with PSO. To enhance the performance metrics of IDS, the proposed system provides effective techniques which are comprised of three phases such as preprocessing, feature selection and classification. In the preprocessing stage, the covariance matrix and mean value compute the missing values, thereby increasing classification accuracy. Kalman filtering replaces the missing values in the KDD dataset and results in improved classification accuracy. Then, the feature selection is carried out using GFA to choose the optimal features from the dataset. The feature selection algorithm is focused to increase the brightness value which retrieves the global optimal solutions. The objective of best value function increases the optimal intrusion detection rates for the specified KDD dataset. By using IRVM classification method, the intrusion and normal features from the dataset are more effectively extracted. Relevance vectors and the number of samples are executed using kernel random vectors. Performance metrics such as precision, recall, specificity and accuracy have been improved. An existing AdaBoost algorithm and AdaBoost based AdaBoost SVM with PSO algorithm provides lower performance, whereas the proposed IRVM with GFA provides higher performance. Experimental results imply that the proposed IRVM + GFA has superior performance rather than existing approaches. However, for large-scale intrusion dataset, the intrusion detection is not robust; hence, it leads to high attack rates. Hybrid optimization feature selection algorithm with classification approach can handle high-dimensional data, and it would be for future work to yield higher classification accuracy. Future work may tackle with application attacks and determine the signature for remote buffer overflow attacks. In further research, enhancing machine learning techniques can act as classifier to improve detection rate.

Compliance with ethical standards

Conflict of interest The author declares that they have no conflict of interest.

References

An J-Y et al (2016) Using the relevance vector machine model combined with local phase quantization to predict protein–protein interactions from protein sequences. BioMed Res Int 2016

Arai K (2013) Recovering method of missing data based on proposed modified Kalman filter when time series of mean data is known. Int J Adv Res Artif Intell 7(2):18–23

Bishop CM, Tipping ME (2000) Variational relevance vector machines. In: Proceedings of the sixteenth conference on uncertainty in artificial intelligence, Morgan Kaufmann Publishers

Ektefa M, Memar S, Sidi F, Affendey LS (2010) Intrusion detection using data mining techniques. IEEE, Shah Alam, Selangor, Malaysia, pp 200–203

Farahani SM, Abshouri AA, Nasiri B, Meybodi MR (2011) A Gaussian firefly algorithm. Int J Mach Learn Comput 1:448–453

Gao J et al (2009) Adaptive distributed intrusion detection using parametric model. In: IEEE/WIC/ACM international joint conferences on web intelligence and intelligent agent technologies, 2009. WI-IAT'09, vol 1. IET

Han J, Kamber M (2011) Data mining: concepts and techniques, 3rd edn. Morgan Kufmann, San Mateo **(2nd edition 2006)**

Hu W, Hu W, Maybank S (2008) Adaboost-based algorithm for network intrusion detection. IEEE Trans Syst Man Cybern B Cybern 38(2):577–583

Hu W, Gao J, Wang Y, Wu O, Maybank S (2014) Online adaboost-based parameterized methods for dynamic distributed network intrusion detection. IEEE Transactions on Cybernetics 44(1):66–82

Li D, Cai Z, Deng L, Yao X, Wang HH (2018a) Information security model of block chain based on intrusion sensing in the IoT environment. Clust Comput 1–18. https://doi.org/10.1007/s10586-018-2516-1

Li D, Deng L, Gupta BB, Wang H, Choi C (2018b) A novel CNN based security guaranteed image watermarking generation scenario for smart city applications. Inf Sci. https://doi.org/10.1016/j.ins.2018.02.060

Mabu S, Chen C, Lu N, Shimada K, Hirasawa K (2011) An intrusion-detection model based on fuzzy class-association-rule mining using genetic network programming. IEEE Trans Syst Man Cybern C Appl Rev 41(1):130–139

McHugh J (2001) Intrusion and intrusion detection. Int J Inf Secur 1(1):14–35

Medhane DV, Sangaiah AK (2017) Search space-based multi-objective optimization evolutionary algorithm. Comput Electr Eng 58:126–143

Nasiri B, Meybodi MR (2016) History-driven firefly algorithm for optimisation in dynamic and uncertain environments. Int J Bio-Inspired Comput 8(5):326–339

Nayak J, Naik B, Behera HS (2015) A novel nature inspired firefly algorithm with higher order neural network: performance analysis. Int J Eng Sci Technol 19:197–211

Panda M, Patra MR (2008) A comparative study of data mining algorithms for network intrusion detection. In: First international conference on emerging trends in engineering and technology, pp 504–507

Peddabachigari S, Abraham A, Grosan C, Thomas J (2007) Modeling of intrusion detection system using hybrid intelligent systems. J Netw Comput Appl 30:114–132

Sangaiah AK, Thangavelu AK, Gao XZ, Anbazhagan N, Durai MS (2015) An ANFIS approach for evaluation of team-level service climate in GSD projects using Taguchi-genetic learning algorithm. Appl Soft Comput 30:628–635

Sangaiah AK, Samuel OW, Li X, Abdel-Basset M, Wang H (2018a) Towards an efficient risk assessment in software projects—fuzzy reinforcement paradigm. Comput Electr Eng 71:833–846

Sangaiah AK, Fakhry AE, Abdel-Basset M, El-henawy I (2018b) Arabic text clustering using improved clustering algorithms with dimensionality reduction. Clust Comput 1–15. https://doi.org/10.1007/s10586-018-2084-4

Stolfo SJ, Lee W, Chan PK, Fan W, Eskin E (2001) Data mining-based intrusion detectors: an overview of the columbia IDS project. ACM SIGMOD Rec 30(4):5–14

The UCI KDD Archive (1999) Information and Computer Science, University of California, Irvine. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. Accessed 2 Feb 2014

Upadhyaya D, Jain S (2013) Hybrid approach for network intrusion detection system using k-medoid clustering and Naïve Bayes classification. Int J Comput Sci Issues (IJCSI) 10(3):231–236

Xiang MY, Chong, Zhu HL (2004) Design of multiple-level tree classifiers for intrusion detection system. In: IEEE conference on cybernetics and intelligent system

Yang X-S (2009) Firefly algorithms for multimodal optimization. Stochastic algorithms: foundations and applications. Springer, Berlin, pp 169–178

Yu H, Yang J, Han J (2003) Classifying large data sets using SVM with hierarchical clusters. In: Proceedings of the SIGKDD 2003, Washington, DC, pp 306–315

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.