STM JOURNALS

Journal of Artificial Intelligence Research & Advances
ISSN: 2395-6720 (Online)
Volume 6, Issue 2
www.stmjournals.com

Database Intrusion Detection Using Genetic Support Vector Fuzzy Clustering Learning Model

Anitarani Brahma**, SuvasiniPanigrahi
Department of Computer Science and Engineering, Veer Surendra Sai University of Technology, Burla, Odisha, India

Abstract

The rapid development of computer networks and increasing dependency of almost all companies and government agencies on internet and cloud computing lead to the problem of stability and security, like intrusions in several forms which can cause huge loss to these organizations. During recent years, disaster in data due to intrusions has dramatically increased. The hindrance of such intrusions is entirely dependent on their detection part which can be possible through a high-performance based intrusion detection system in database which has higher accuracy rate and negligible false positive rate. As part of funded effort in database security, soft computing has proven to be capable of creating a system capable of detecting and characterizing anomalous behaviour which is composed of evolutionary computing tools with artificial neural networks and/or fuzzy logic. In this progression, here we present a database intrusion detection system, by applying genetic algorithm for feature extraction; and fuzzy clustering and support vector machines are used for detection purpose to efficiently detect insider threat with a reasonable false positive rate.

Keywords: *Genetic algorithm, support vector machine, fuzzy clustering, database intrusion detection system*

****Author for Correspondence*** E-mail: brahmaanita00@gmail.com.

INTRODUCTION

In recent days, it is very important to maintain a high-level security to the data contained in the database and to the trusted communication channels of information. Database management system which represents the ultimate layer in preventing malicious access towards database; in traditional systems, commercial database deploys security mechanism like authorization, access control, inference control, multi-level secure transactions processing, database encryption etc. However, such prevention-based techniques may be fooled by knowledgeable attackers. Hence, Database Intrusion Detection Systems (DIDSs) have become essential component for securing sensitive and confidential data contained in databases. There are various approaches developed till now for database intrusion detection, but unfortunately most of them are able to detect intrusive activities effectively but lead to the generation of large number of false alarms as well. So, the objective in this work is to further improve the detection rate while minimizing the false alarm rate.

Generally, knowledgeable attacker or intruder is defined as a person who illegally tries to and may become successful in breaking into the database systems. Intruders from within the organization pose more danger as they have authorized access to the database and can misuse their privileges. The detection of insider attacks are therefore more challenging than outsider attacks and needs to addressed in the best possible manner [1].

Generally, existing systems on insider intrusions are based on anomaly IDSs and may range from rule based detection to statistical anomaly detection. It is well known that anomaly detection systems need to maintain the past records of users' behaviours and the statistics of normal usages, which is referred to as "profiles". Such type of detection process

identifies and classifies the malicious behaviour from the legitimate ones by measuring deviation of current patterns from the established profile. The most challenging task is to model the normal behaviour or the normal usage pattern of a database user on a database system. In the current study, we have thus emphasized on building user profiles more accurately by initially applying feature selection using Genetic Algorithm (GA) for extracting features relevant for intrusion detection. Once the useful features are determined, normal profile construction of database users is done by utilizing Fuzzy Clustering algorithm. The classification of an incoming transaction is done by applying Support Vector Machines (SVMs) based on a distance measure computed from the cluster centres.

In these years, the research in database intrusion detection field has received much attention. In such progression, Chung *et al*. presented a misuse intrusion detection system which identifies the frequent data items referenced together and exploits them for detection [2]. Lee *et al.* also studied the frequently updated data and utilized them for database intrusion detection [3, 4]. Rakesh *et al.* extended the functionality of DBMS by recording and analysing the accessing habit of every user and compared them in real time with each access toward the database [5]. Hidden Markov Model based DIDS has been proposed by Barbara *et al.* to model the behaviour of a user [6]. Another work by Lee *et al.* used time signatures of data items in detecting the malicious access towards the database [7]. Srivastava *et al*. proposed a data mining based DIDS which considers the sensitivity of attributes while mining the dependency rule among the data item [8]. Panigrahi *et al*. introduced a two stage database intrusion detection system which considers both inter and intra transactional features as well as the sensitivity level of attributes within a database [9, 10].

This study addresses the issue of identifying essential database transactional features which are useful in capturing user's normal profile accurately so that each user's behavioural pattern is identified uniquely. The deviation from the normal patterns gives an indication of intrusive behaviour, which can be easily detected by the classifier model used in our proposed DIDS.

The remainder of the paper is organized as follows: Next part of the paper gives an idea about the details of the proposed database intrusion detection system along with preliminaries of the techniques used. Then it describes the implementation procedure, performance evaluation and comparative analysis with existing work. And finally, it concludes with a summary of the work done with directions for further research.

MATERIALS AND METHODS

The detection methodology of the proposed DIDS has been discussed later in this paper. Initially, the transactional dataset is pre-processed and a normalized dataset is generated, which is divided into a train and test set by applying 10-fold cross validation. Feature extraction using GA is applied on the train set for determining the important features and thereby resulting in dimensionality reduction of the dataset. The reduced training data set is clustered by fuzzy clustering algorithm for forming clusters of normal behaviour. During the training phase, a trained SVM classifier model is generated by training it with the output of the fuzzy cluster module. This training helps the Support Vector Machine classifier to accurately discriminate the attack patterns from the normal ones. Once the training procedure is over, the test data is feed in to trained SVM model for classification.

Pre-processing of Data

Generally, the real world data are incomplete, noisy and inconsistent; for this reason it needs to be passing through some pre-processing steps to produce quality data which are eligible for quality knowledge extraction process [11]. The input dataset is pre-processed so as to normalize the data required as input for the proposed model. We have mainly focused on

data cleaning and data transformation phases in this step which are described below:

Data Cleaning

The received data is not the complete one where some of the data are missing, inconsistent with other recorded data, and hence the data set is cleaned through binning and made it complete for transformation process.

Data Transformation

The cleaned data is transformed to normalized, aggregated data through normalization process which involves scaling of all values in order to fall within a small specified range for given attribute (Here we took the range from 0 to 1) through min-max normalization process.

Feature Extraction using GA

In this proposed DIDS, the applied dataset holds several features which contain false correlations, which may be redundant and which also increase computation time and can impact the accuracy of DIDS. It is very hard to know which of these features are redundant or irrelevant for DIDS and which ones are relevant or essential for DIDS. In this study, we use genetic algorithm as depicted in Figure 1 to select useful features for the detection purpose. This dimensionability reduced selected subset features data are then used to detect malicious transactions.

In the field of soft computing, a genetic algorithm is a meta-heuristic searching algorithm which mimics the process of natural selection and evolution. The evolution usually starts from a population of randomly generated individuals and moved through several iterations; in each iteration or generation, the fitness of each individual is evaluated; the fitness is usually the value of the objective function in the optimization problem being solved. The more fit individuals are selected from the current population and combined or crossed to produce new generation or population which are again fed as candidate solutions in the next iteration of the algorithm; the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population [12]. However, from the literature survey, it is observed that genetic algorithm has proven to be one of the best algorithms for feature selection in image processing and also in intrusion detection process [13].

Feature selection and extraction plays very crucial role in optimizing performance of a classifier and is also a critical problem. The goal of feature selection is the task of finding best possible features y that satisfies some optimal performance characterised by probability of classification as in Eq. (1) from initial N features which are redundant and/or noise in the dataset.

$$y=M(x) \quad (1)$$

In the proposed approach, as shown in Figure 1, the normalized dataset (We assume it as ‘x’) are fed to the feature extraction process which is done through GA. It defines a population of weight vectors w*, where the size of each w* is the size of the data pattern x. Each w* from the GA is multiplied with every sample’s data pattern vector x, yielding new vector y for the given data. The K-Nearest Neighbourhood (KNN) algorithm then classifies the transformed dataset y [14]. The classification result for data set is fed back as the fitness function, to guide the searching algorithm for the best feature extraction. GA produces the best w* vector; for example, the best w* might have fewest features and gives misclassification results.

10-fold Cross Validation

The dataset with reduced features are fed to the cross validation process which is a model validation technique to generate a generalized independent dataset for training and testing phase of the model to limit problems like over fitting. Ten numbers of folds or rounds of cross-validation are performed using different partitions and the validation results are averaged over the folds to reduce the problem of variability. In addition, this phase is proven to be useful estimator of model performance as there is not enough data available or there is not a good distribution of data to partition it into separate training and test sets in the conventional validation method [15].

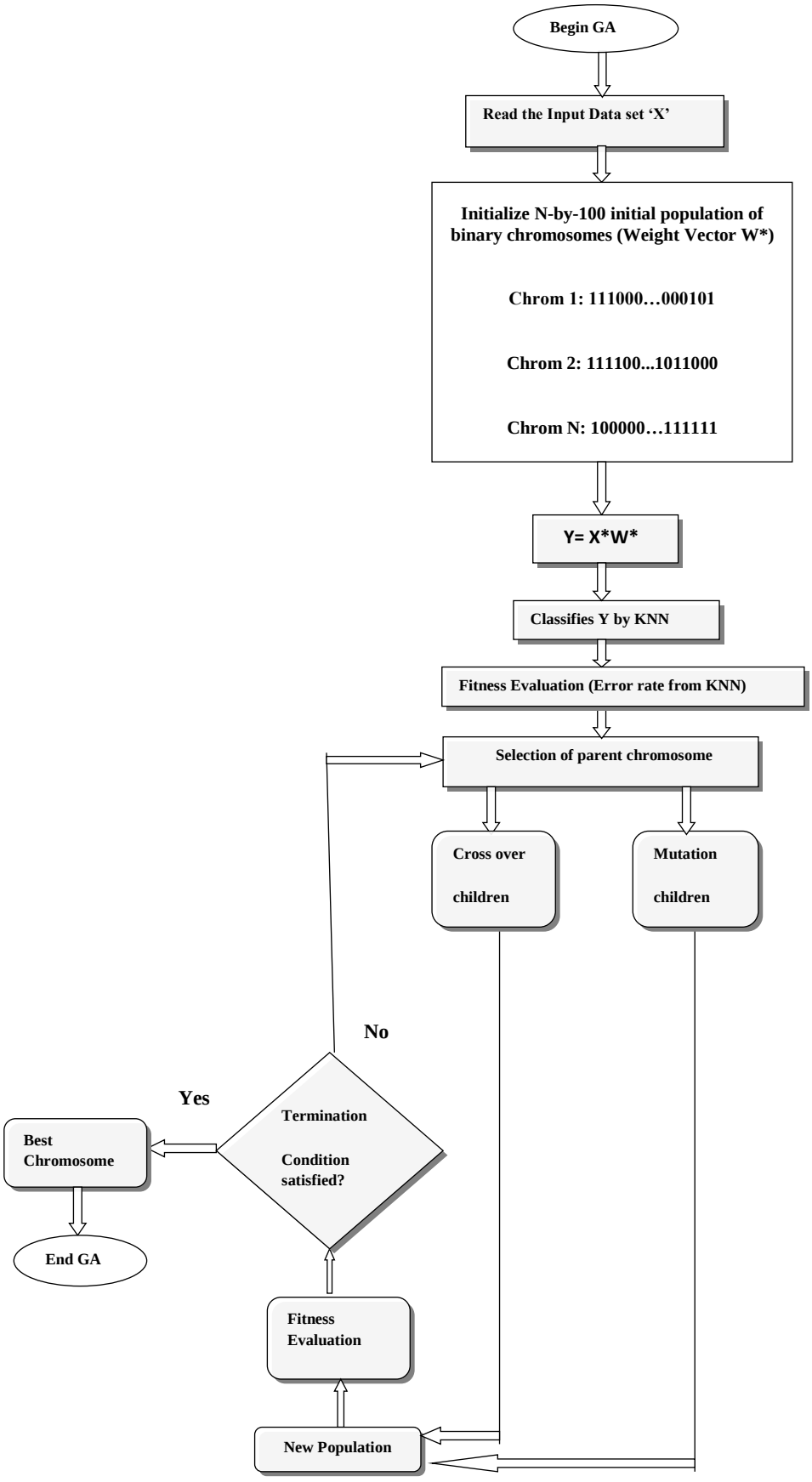

Fig. 1: *Flow Chart of Genetic Algorithm in Feature Selection.*

Fuzzy Clustering

The final pre-processed training data set are applied to fuzzy clustering process which groups the dataset that populate some multidimensional space; represent the normal profile of the database user and also it helps the classification process to produce the accurate result. The clustering process produces a concise representation of normal behaviour of users by identifying natural groupings of data from a large multidimensional dataset. We have applied Fuzzy C_Means (FCM) clustering which clusters the data points where each data point belongs to a cluster with some membership value and outputs a list of cluster centres representing normal profile and several membership grades for each data point [12].

FCM is a data clustering algorithm that starts with an initial guess for the cluster centres that are identified to be the mean location of each cluster and FCM assigns membership grade to each of the data point for each cluster. However, these random cluster centres and the assigned membership grades are likely to be change in each iteration based on an objective function equated in Eq. (2) that represents the distance from any given data point to a cluster weighted by that data point's membership grade.

$$\arg\min c \sum_{i=1}^{n} \sum_{j=1}^{c} w_{ij} |x_i - c_i|^2 \quad (2)$$

Where, $w_{ij} = \frac{1}{\sum_{k=1}^{c} (\frac{|x_i - c_j|}{|x_i - c_k|})^{\frac{2}{m-1}}}$ 1<i<n, 1<j<c (3)

The FCM algorithm groups the dataset X={x_1, x_2,……., x_n} into a collection of c clusters having cluster center {c_1, c_2,…….., c_n}. W_{ij} is the partition matrix which describes the degree to which element x_i belongs to a cluster c_j and is updated in each iteration as in Eq. (3). After determining the clusters, their centroid can be easily found out by applying Eq. (4), where each centroid is a representative of each cluster and are input to the SVM classifier to learn the normal behaviour.

Centroid= No. of data points within a cluster/No. of Clusters (4)

Classification by Applying SVM

The fuzzy clustered training dataset representing normal profile are applied to Support Vector Machines (SVM) classifier which is taught to classify the malicious transactions from genuine. The support vector machine is a non-probabilistic binary linear supervised learning classifier that analyzes the training data and assigns them in to one category or the other; and test data are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on [16].

Generally, SVM is given with a training dataset X={(x_1, y_1), (x_2, y_2)……, (x_n, y_n)} where y_i (is 0 for normal or 1 for malicious) indicates the class to which the x_i belongs. Normally, SVMs are a set of hyper plane as equated in Eq. (5) in a high dimensional space, and can be used for classification, regression or other tasks as shown in Figure 2.

$$(w^T x) + b = 0 \quad (5)$$
$$(w^T x_i) + b > 0 \text{ if } y_i=1 \quad (6)$$
$$(w^T x_i) + b < 0 \text{ if } y_i=0 \quad (7)$$
$$f(x) = sin((w^T x)13 + b) \quad (8)$$

Where,
X={x_1, x_2,……,x_n} are the data points or the training vector,
W= Normal vector to the hyper plane, and
b= Offset of the hyper plane from the origin.

The above constraints defined in Eqs. (6) and (7) state that each data point must lie on the correct side of the margin or we can say whether the given points belong to the support of the distribution. However the test dataset are being classified through a decision function as defined in Eq. (8).

However, the response time of SVM classifiers is high due to dependence on number of input variables and that of support vectors. Therefore, the proposed approach uses FCM clustering process to cluster the dataset which represents the normal behaviour and also helps the SVM classifier to discriminate between the malicious and the normal transactions by reducing the support vector so as to reduce the response time of classifier.

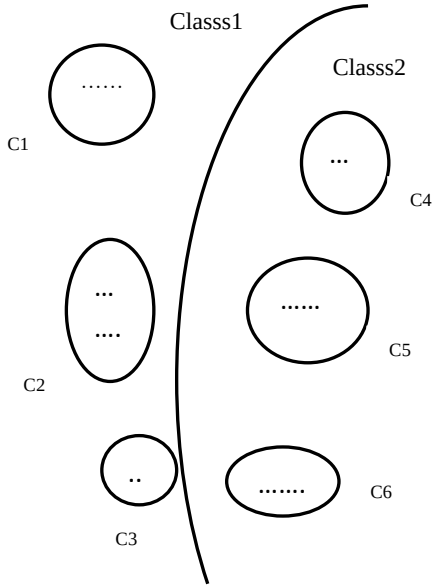

Fig. 2: *Separating Hyperplane in SVM.*

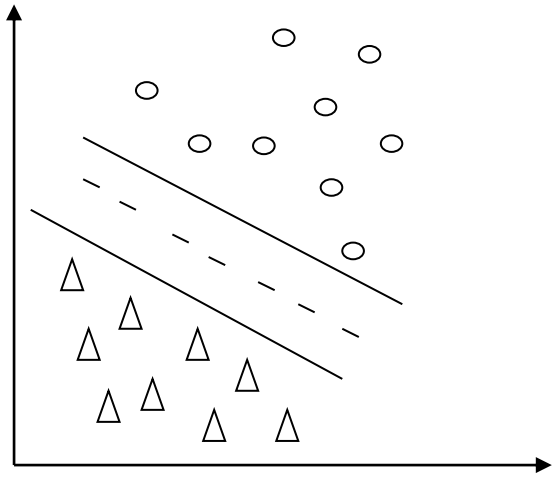

Fig. 3: *Classification of Clustered Data through SVM.*

Figure 3 illustrates the classification of clustered data where each cluster is represented by their centroid; number of centroids is equal to the number of clusters formed. The centroid is represented as normal profile and can be calculated by the Eq. (4). The centroids are fed as input to the SVM classifier that classifies the data by applying Eq. (8).

IMPLEMENTATION PROCEDURE AND PERFORMANCE EVALUATION

The proposed GA_Fuzzy_SVM for Database Intrusion Detection is implemented with a synthetically generated dataset in Matlab13 tool [17]. The applied dataset undergoes through the above discussed pre-processes and the meaningful or the wanted features are extracted from the processed dataset by GA which is embedded with KNN algorithm. The dimensionally reduced data are 10-fold cross validated to produce a generalized train and test dataset. The trained data is clustered by FCM where each cluster represents normal profile of user accesses towards the database and is trained by SVM classifier. The trained SVM classifier is then tested against test dataset which can distinguish the malicious transactions from the normal ones with low FPR and higher accuracy.

The correctness and efficiency of a classifier can be determined through its Accuracy, True Positive Rate (TPR), False Positive Rate (FPR), Precision and Recall. Table 1 describes the procedure for determining these performance measures. TPR indicates the percentage of intrusive tuples that are correctly labelled by the model and FPR refers to rate of the genuine samples that are mislabelled as intrusive. The accuracy of a classifier on a given test set is the percentage of test set tuples that are correctly classified by the classifier. Precision, otherwise called positive predictive value which is the fraction of retrieved tuples are normal or relevant; whereas, recall is the fraction of normal tuples that are retrieved. So, precision can be termed as a measure of exactness or quality, whereas recall is a measure of completeness or quantity. After the model is trained to distinguish between malicious and genuine transactions, the test dataset is used to find out the accuracy, TPR, FPR, Precision and Recall of the proposed DIDS.

The following figures show the comparative results of the proposed DIDS by applying SVM, SVM with fuzzy clustering, GA_Fuzzyclustering_SVM based on the five parameters: Accuracy, TPR, FPR, Precision and Recall. From the results shown in Figure 4, it is clearly understood that GA_Fuzzy_SVM is showing the superiority in terms of all the three parameters. The performance of the proposed DIDS is giving better performance than SVM without

clustering and SVM with FCM clustering as shown in Figure 5. The precision graph as shown in Figure 6 justifies the performance measure of positive predictions of the proposed algorithm. The Recall graph as shown in Figure 7 describes about the positive part of the database of the dataset and the values are same as sensitivity of the developed model. The elapsed time of the proposed approach as shown in Figure 8 is very low compared to other approaches because of the reduced number of input vector to SVM by applying GA and FCM.

Table 1: *Determination of the Performance Measure.*

Performance Measures	Procedure	Notations Used
Accuracy	(TP+TN)/TP+FP +FN+TN)	TP: Correctly predicted positive values
TPR	TP/P	TN: Correctly predicted negative values
FPR	1-(TN/N)	
Precision	TP/(TP+FP)	
Recall	TP/(TP+FN)	FP: Incorrectly classified positive samples
		FN: Incorrectly classified negative samples
		P: Total number of positive samples
		N: Total Number of negative samples

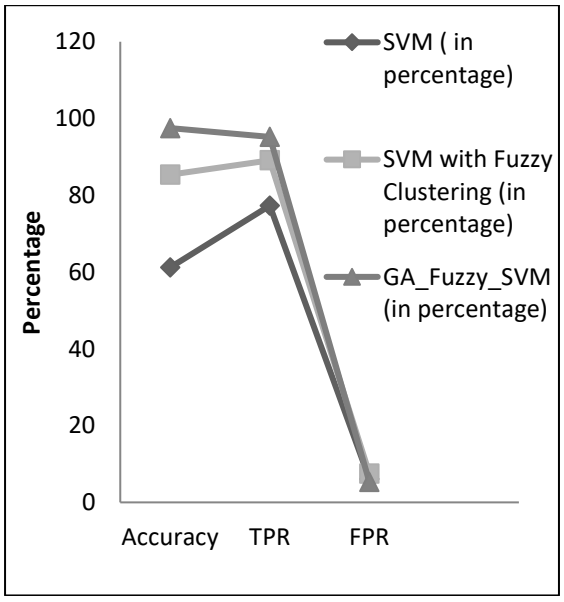

Fig. 4: *Performance Analysis of GA_Fuzzy_SVM DIDS.*

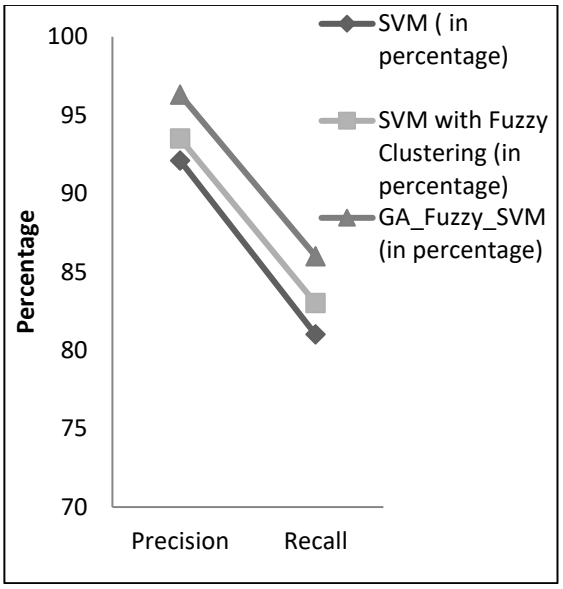

Fig. 5: *Performance Analysis of GA_Fuzzy_SVM DIDS.*

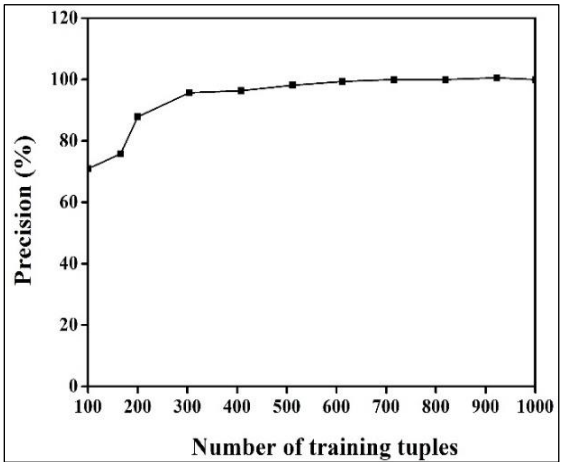

Fig. 6: *Precision Graph of GA_Fuzzy_SVM DIDS.*

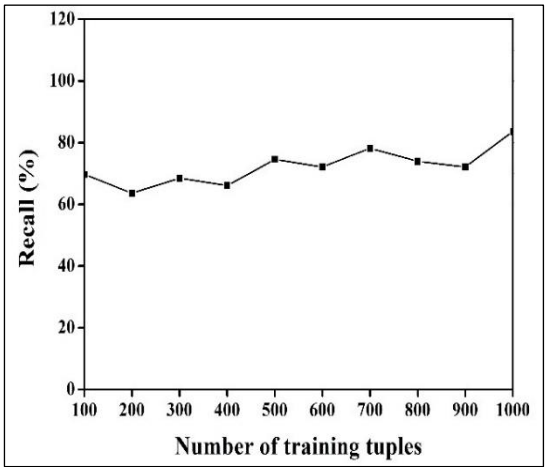

Fig. 7: *Recall Ananlysis Graph of GA_Fuzzy_SVM DIDS.*

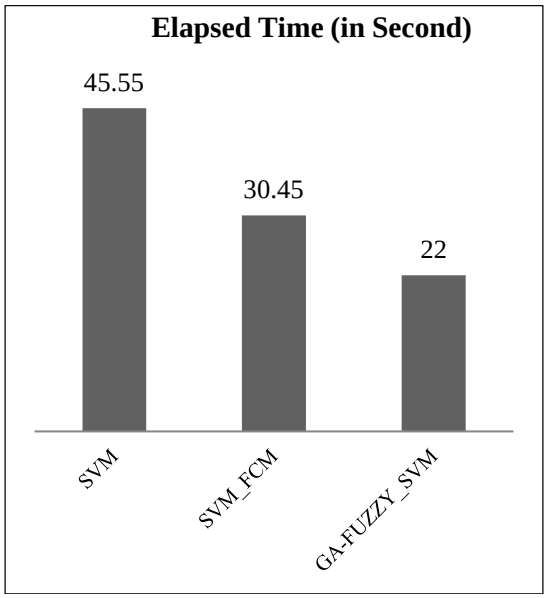

Fig. 8: *Elapsed Time of the Proposed Model.*

CONCLUSION

Security is the primary concern in the field of data and information. Since prevention techniques cannot be sufficient and new intrusions are continuously emerging, DIDS is an indispensable part of a security system. We investigated with three widely used soft computing paradigms such as genetic algorithm, fuzzy clustering and SVM to efficiently detect intrusions in database. To implement and measure the performance of the proposed model, we used the synthetically generated dataset. The GA_Fuzzy_SVM shows higher percentage of detection than those of its counterpart SVM and Fuzzy_SVM and enhances the testing time due to data dimension reduction and feature selection. In future research, other soft computing techniques such as Radial Basis Function Network and other optimization techniques can be applied, and their comparisons will guide to build best model for database intrusion detection.

Abbreviations Used in the Proposed System
DIDS: Database Intrusion Detection Systems
GA: Genetic Algorithm
SVM: Support Vector Machine
FCM: Fuzzy C-Mean Algorithm
FPR: False Positive Rate
TPR: True Positive Rate

Declaration of Interest: None

REFERENCES

1. Mathew S, Petropoulos M, Ngo HQ, *et al*. A Data-Centric Approach to Insider Attack Detection in Database Systems. *Proceedings of the 13th International Conference on Recent Advances in Intrusion Detection*; 2010 Sept 15–17; Ottawa, Ontario, Canada. Verlag Berlin, Heidelberg:Springer; 2010.
2. Chung CY, Gertz M, Levitt K. DEMIDS: A Misuse Detection System for Database Systems. In: van Biene-Hershey ME, Strous L, editors. *Integrity and Internal Control in Information Systems*. IICIS 1999. IFIP, The International Federation for Information Processing, vol 37. Springer, Boston, MA, 159–78p.
3. Lee W, Stolfo S. A Framework for Constructing Features and Models for Intrusion Detection Systems. *ACM T Inform Syst Se*. 2000; 3(4): 227–61p.
4. Lee W, Xiang D. Information-Theoretic Measures for Anomaly Detection. *Proceedings 2001 IEEE Symposium on Security and Privacy*; 2000 May 14–16; Oakland, CA, USA. USA: IEEE; 2002.
5. Rakesh A, Ramakrishnan S. Fast Algorithms for Mining Association Rules. *Proceeding of 20th International Conference on Very Large Data Bases.* Berlin: Morgan Kaufmann; 1994; 487–99p.
6. Barbara D, Goel R, Jajodia S. Mining Malicious Data Corruption with Hidden Markov Models. In: Gudes E, Shenoi S, editors. Research Directions in Data and Applications Security. *IFIP, The International Federation for Information Processing*, Vol. 128. Springer, Boston, MA. 175–89p.
7. Lee VCS, Stankovic J, Son S. Intrusion Detection in Real Time Databases via Time Signatures. *Proceedings of the 6th IEEE Real-Time Technology and Applications Symposium (RTAS)*; 2000 31 May–2 Jun; Washington, DC, USA. USA: IEEE; 2000.
8. Srivastava A, Sural S, Majumdar AK. Weighted Intra-Transactional Rule Mining for Database Intrusion Detection. In: Ng WK, Kitsuregawa M, Li J, *et al.*, editors. Advances in Knowledge Discovery and Data Mining. PAKDD 2006. *Lecture Notes in Computer Science*, Vol. 3918. Springer, Berlin, Heidelberg.

9. Panigrahi S, Sural S, Majumdar AK. Two-Stage Database Intrusion Detection by Combining Multiple Evidence and Belief Update. *Inf Syst Front*. 2013; 15(1): 35–53p.
10. Panigrahi S, Sural S, Majumdar AK. Detection of Intrusive Activity in Databases by Combining Multiple Evidences and Belief Update. In *2009 IEEE Symposium on Computational Intelligence in Cyber Security*; 2009 30 Mar–2 Apr; Nashville, TN, USA. USA: IEEE; 2009.
11. Han J, Kamber M. *Data Mining, Concept and Techniques*. 2nd Edn. San Francisco: Morgan Kaufmann; 2006.
12. Jang JSR. *Neuro-Fuzzy and Soft Computing*. New Delhi: PHI Publication, Pearson Education; 2004.
13. Wu SX, Banzhaf W. The Use of Computational Intelligence in Intrusion Detection Systems: A Review. *Appl Soft Comput*. 2010; 10(1): 1–35p.
14. Wenchao L, Ping Y, Yue W, *et al*. A New Intrusion Detection System Based on KNN Classification Algorithm in Wireless Sensor Network. *Journal of Electrical and Computer Engineering*. Article ID: 240217. 2014; 8p.
15. Sharma RC, Hara K, Hirayama H. A Machine Learning and Cross-Validation Approach for the Discrimination of Vegetation Physiognomic Types Using Satellite Based Multispectral and Multitemporal Data. *Scientifica*. Article ID: 9806479. 2017; 8p.
16. Chen W, Hsu S, Shen H. Application of SVM and ANN for Intrusion Detection. *Comput Oper Res*. 2006; 32(10): 2617–34p.
17. Transaction Processing Performance Council. TPC Benchmark™ W (Web Commerce), Specification, version 1.2.0. 2008. http://www.tpc.org/tpcw/default.asp.

Cite this Article
Anitarani Brahma, SuvasiniPanigrahi. Database Intrusion Detection Using Genetic Support Vector Fuzzy Clustering Learning Model. *Journal of Artificial Intelligence Research & Advances*. 2019; 6(2): 32–40p.