Received: February 4, 2021. Revised: April 8, 2021.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

IDS-MIU: An Intrusion Detection System Based on Machine Learning Techniques for Mixed type, Incomplete, and Uncertain Data Set

Musaab Riyadh[1]* **Basim Jamil Ali**[1] **Dina Riadh Alshibani**[1]

[1]*Collage of Science, Computer Department, Mustansiriyah University, Baghdad, Iraq*
* Corresponding author's Email: m.shaibani@uomustansiriyah.edu.iq

Abstract: The rapid growth of computer networks has led to massive flow of data every second. Some of these data flow is a malicious activity and cannot detect by anti-malware and firewalls. Therefore, an intrusion detection system is an urgent issue aims to distinguish between non relevant and relevant data in order to maintain data availability and integrity. Due to this, a hybrid intrusion detection system is proposed in this study based on machine learning techniques to tackle various challenging issues in data set such as mixed type data, incomplete, and uncertain data. The proposed system has achieved its objectives by supporting: Firstly, the density based clustering approach due to its robustness to noise removal. Secondly, the K means and K-nearest neighbour algorithms to transform n dimensional data into one dimensional data in order to deal with mixed type data and minimize the running time. Finally, a special type of dissimilarity measure has been supported to tackle the problem of missing data. The experimental results illustrate that the proposed classifier has better or similar classification accuracy (92.9%) as compared with support vector machine (91.5%), CANN(92.2%), Random forest(93.3%), and EIDS-ACC- OD(91.9%) classifiers in KDD Cup99 data set. However, the proposed classifier has the best performance (92.2%) and (91.2%) when randomly removing 5% and 10% of the KDD Cup99 data set in spite of decreasing the overall accuracy classification for all the classifiers.

Keywords: Intrusion detection system, DBSCAN, KNN, Uncertain data, Incomplete data.

1. Introduction

Recently, the cost of cyber crimes (e.g. services denial, web based attack, and malicious insiders) are exponentially increasing. These cyber crimes are caused by malicious software which sneak over the networks and lead to steal the intellectual property of organization and disruptions of a company infra-structure [1]. In response to this, various approaches (e.g. anti-malware, firewalls, and Intrusion detection system IDS) have been developed to effectively tackle the challenges of cyber-crimes over computer networks. IDS the focusing of this study can be defined as a security mechanism which is used to monitor and detect unauthorized and malicious activities such as worms, viruses, DDoS attacks over networks traffic [2, 3, 4]. The IDS mechanism can be categorized into two basic approaches: Signature-based intrusion detection system (SIDS) and Anomaly-based intrusion detection system (AIDS). The key idea behind the SIDS is to compare the signature of current activity with a list of previously stored intrusion signatures and an alarm is triggered when a match is found. Therefore, the SIDS approach is hardly detecting a new attack which has no previous pattern in the database which represents the main weak point of this approach [5]. In the AIDS which is the focusing of this work, a model for the normal behaviour of a computer system is build based on machine learning techniques, any remarkable deviation between the model and the observed behaviour can be considered as an intrusion. Due to this, the data updating is not required in the AIDS approach to detect new attacks [6]. Many researchers suggested AIDS system based on single machine learning techniques (SLT) such as K-nearest neighbour (KNN) algorithm [7, 8], Support vector machine SVM [9, 10], Naïve Bayesian [11], decision trees [12, 13]. Other researchers suggested AIDS based on hybrid learning techniques (HLT), Ahmad et al. [14] proposed an accurate hybrid technique based on the K means clustering algorithm and Gaussian Mixture Model (GMM) and Random Forest classification technique. Saleh et al. [15] suggested a hybrid IDS based on optimized SVM and prioritized K-nearest neighbors classifiers, however this system is not convenient for massive data with high dimensions. A hybrid real time IDS in [16] was proposed depending on two layers of neural networks, the first one performs as an outliers-based detection for anonymous attacks and the second one performs as a misuse-based detection for anonymous attacks. A more sophisticated multi-level IDS was suggested by Al-Yaseen et al. [17] based on SVM and extreme learning machine. This system remarkably increases the detection accuracy for various types of attacks; however, the system was built for specific data set (KDD Cup 99) and it is difficult to apply it to other data set. Kaja et al. [18] adopted the K means algorithm in the first stage for clustering the data and tested the classifiers J48, random forest, Naıve Bayes, and Adaptive Boosting in the second stage. The results show that the random forest classifier has the best classification accuracy. It is clear that the HLT techniques are more accurate than the SLT but they are time consuming techniques. Many works have enhanced the classification accuracy by adopting an effective data pre-processing techniques such as normalization and data reduction. Setiawan et al. [19] suggested IDS based on the modified ranked based features selection, log normalization, and Support Vector Machine of multi-class to enhance the overall accuracy and maximize the discovering of the minority classes such as R2L and U2R. Kumar et al. [20] proposed Multi-Linear Dimensionality Reduction to extract features and SVM of multi-class classifier to decrease the classification accuracy and reduce running time of training phase. However, most of the aforementioned studies have focused only on enhancing the classification accuracy of the IDS and did not pay attention to the challenging issues in data set such as incomplete and noisy data.

On the other hand, IDS datasets have various challenges such as high dimensionality, mixed-type, and noisy data that remarkably affect the detection accuracy. These challenges must be taken into consideration when designing an efficient IDS [21, 22]. Various studies have been conducted to tackle these challenges: the systems DSSVM and CANN in [23,24] transformed n dimensional data of mixed-type into one dimensional data and classified these data based on KNN and SVM classifiers in order to minimized the memory storage and the running time of IDS. Manjunatha et al. [25] proposed an efficient algorithm based on enhancing the Canberra method and Minimum Threshold Support Count to detect intrusions in high-dimensionality data set that consists of numerical and categorical features. Other studies have focused on the effects of noise in the performance of IDS. The works in [21, 26] eliminate the noisy patterns based on the DBSCAN clustering algorithm in order to enhance the classification accuracy of IDS. Bhosale et al. [27] suggested a noise removal algorithm to enhance the classification accuracy of Naive Bayes classifier however, it is a time consuming classifier. Hussain et al. [28] proved that Self Organization Map has better intrusion detection accuracy in noise data than widespread classifiers (JRip, J48, RF, NBTree) despite of the low performance in normal data. Sandosh et al. [29] suggested an IDS (EIDS-ACC-OD) depend on the clustering agent and KNN classifier. The classification accuracy was enhanced by removing white space noise based on outlier detection techniques. These studies focused on the importance of eliminating noise to enhance the classification accuracy. However, these studies supported similarity measures such as Euclidean distance which are significantly affected when using incomplete data. Table 1 shows a comparison between the related works.

In response to this, a hybrid Intrusion Detection System for Mixed type, Incomplete, and Uncertain data set (IDS-MIU) has been proposed using KDD Cup99 , the suggested system adopts firstly; the DBSCAN clustering algorithm to remove noise from data set. Secondly, a special kind of similarity measure to tackle the problems of mixed type and incomplete data, thirdly, transforms N dimensional data to one dimension data to improve the running time. To the best of our knowledge there are no studies tackling all these data set challenges together especially the problem of incomplete data set. The rest of the article has been organized as follows: section 2 explains the concepts of DBSCAN clustering algorithm. The similarity measure that supported in this study is explained in section 3. Section 4 describes the proposed system. The results are described in the fifth section. Ultimately, the conclusion has been provided in section 6.

2. DBSCAN algorithm

The Density-based spatial clustering of applications with noise (DBSCAN) clustering algorithm has been supported in the pre-processing phase of IDS-MIU to remove noise data.

Table 1. Related works comparison

Work	Techniques	Data Set	Mixed type data	Noisy data	Incomplet	Evaluation method
Saleh [15]	GMM, K-means	KDD Cup99	✓	x	x	AC[1], FAR[2], DR[3]
Al-Yaseen [17]	SVM, extreme learning machine	Only KDD Cup99	✓	x	x	DR, AC, FAR
Kaja [18]	K-means, Random Forest	KDD Cup99	✓	x	x	AC
Setiawan [19]	Multi-Class SVM	NSL KDD	✓	x	x	AC
Kumar [20]	Multi-Class SVM	NSL KDD	✓	✓	x	AC
Dong [21]	K-means, DBSCAN	NSL KDD	✓	✓	x	AC, Precision
Chen [22]	DBSCAN	DARPA	✓	✓	x	TDR[4], FDR[5]
Guo [23] DSSVM	SVM	KDD Cup99	✓		x	DR, ROC[6]
Lin [24] CANN	K-means, KNN	KDD Cup99	✓	x	x	AC
Manjunatha [25]	Canberra method, MTSC[7]	KDD Cup99	✓	x	x	AC
Shakya [26]	K-means, DBSCAN, SMO[8]	KDD Cup99	✓	✓	x	AC
Bhosale [27]	Naive Bayes	KDD Cup99	✓	✓	x	AC, Precision
Hussain [28]	NN(SOM[9])	KDD Cup99& NSL-KDD3	✓	✓	x	AC, TPR[10], FPR[11], ROC
Sandosh [29] EIDS-ACC-OD	K-means, KNN	KDD Cup99	✓	✓	x	AC, Precision

[1]Accuracy [2]False Alarm Rate [3]Detection Rate [4]True Detection Rate [5]False Detection Rate [6]Receiver Operating Characteristic [7]Minimum Threshold Support Count [8]Sequential Minimal Optimization [9] Self-Organizing Map [10]True Positive Rate [11]False Positive Rate

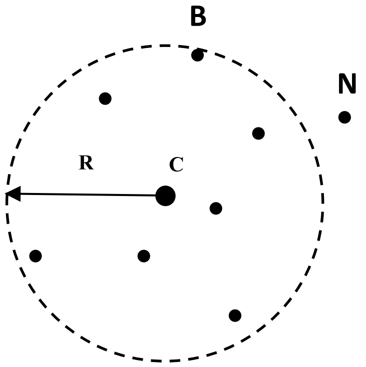

Figure. 1 DBSCAN Concept; C: core point, B: boarder point, and N: noise point

It is unsupervised machine learning technique capable to classify data points into three categories: core points, boarder points, and outlier/noise points [30] as illustrated in Fig. 1. According to [31], noise pattern can be defined as the one that not belongs to any cluster or belong to a cluster with small number of items. A special kind of dissimilarity measure has been employed for DBSCAN algorithm in order to find the distance between two patterns that have mixed type and incomplete data. the DBSCAN algorithm has been chosen for this phase not only for its ability to remove noise data but it has low computational cost O(n) where n is the number of patterns in data set which led to significantly minimize the running time of pre-processing phase [26].

3. Incomplete and mixed type data Dissimilarity

The data sets that are used to evaluate IDS can be characterized by i) its mixed type (e.g. binary, ordinal, nominal, and numeric) ii) it has missing values due to errors in collecting devices [32]. The dissimilarity measure (distance) between two patterns in data is the core of various machine learning techniques such as clustering and classification which significantly affects the final classification results [33, 34]. Therefore, a special kind of dissimilarity function has been employed to solve the problems of mixed type and incomplete data set as defined in Eq. (1).

$$Dist\left(pt_i, pt_j\right) = \frac{\sum_{f=1}^{N} \text{ŋ}_{pt_ipt_j}^{f}\, dist_{pt_ipt_j}^{f}}{\sum_{f=1}^{N} \text{ŋ}_{pt_ipt_j}^{f}} \qquad (1)$$

Where: Dist(pt_i,pt_j) represents the dissimilarity measure between patterns pt_i, pt_j and N is the number of features in each pattern, the parameter $\text{ŋ}^{f}_{pt_ipt_j}$ is equal to either:

- If there is no measurements for feature f in patterns pt_i or pt_j.
- If f is asymmetric binary feature and pt_i^f=0, pt_j^f =0.
- Otherwise, $\text{ŋ}^{f}_{pt_ipt_j}$ =1.

The contribution of feature f to the dissimilarity between pt_i and pt_j is calculated based on its type:

- If feature f is a numeric type: $dist^f_{pti,ptj}$ = | x_{pti}^f – x_{ptj}^f | / (Max ^{f} – Min^f) , where Max^f and Min^f are the maximum and minimum values of the feature f over all the none missing values.
- If feature f is a nominal type or binary: $dist^f_{pti,}$ptj =0 if pt_i^f = pt_j^f; otherwise, $dist^f_{pti,ptj}$ =1.
- If feature f is ordinal type: convert the rank of attributes r_{pti}^f and r_{ptj}^f to z_{pti}^f and z_{ptj}^f as defined in Eq. (2).

$$z_p^f = \frac{rpf\ -1}{Mf-1} \qquad (2)$$

Where Mf is the possible states number that an ordinal attribute can have, then calculate the dissimilarity as defined in Eq. (3):

$$Dist_{pti,ptj}^{f}\ = \left| z_{pti}^{f} - z_{ptj}^{f} \right| \qquad (3)$$

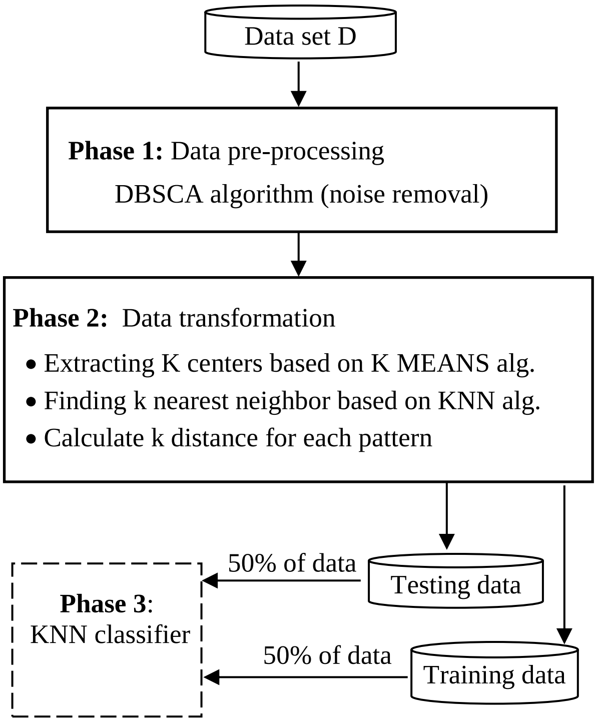
Data set D
Phase 1: Data pre-processing
DBSCA algorithm (noise removal)
Phase 2: Data transformation
• Extracting K centers based on K MEANS alg.
• Finding k nearest neighbor based on KNN alg.
• Calculate k distance for each pattern
50% of data
Testing data
Phase 3: KNN classifier
50% of data
Training data

Figure. 2 The IDS-MIU classifier

Eventually, the supported similarity measure combines the various attributes into a single dissimilarity measure onto a common scale of the scale interval [0.0, 1.0].

4. Proposed system

The IDS-MIU system has been proposed to detect malicious patterns in mixed type, incomplete, and uncertain data set. The IDS-MIU consists of three phases: the data pre-processing, transformation, classification phase as shown in Fig. 2. The first phase aims to remove noise (uncertain) data based on the DBSCAN algorithm since it is robust to noise. The second phase has been dedicated to transform n dimensional data into one dimensional data in order to deal with mixed type data. The final phase of IDS-MIU adopted the KNN classifier, it is a non-parametric machine learning technique uses to assign an unlabelled pattern to the group of its k nearest neighbours

4.1 Data pre-processing

The amount of data information used in IDS is massive, noisy, and redundant [32, 35]. Due to this, the first phase of IDS-MIU system has been dedicated for removing the noisy patterns form data set based on the DBSCAN clustering algorithm. The algorithm sensitivity depends on two input parameters: the clustering distance *eps* and the minimum number of neighbours *minpts*. These parameters significantly affect the clustering results; therefore the accuracy of intrusion detection metric can be used to set the appropriate values of these parameters. According to Cassisi et al [31], noisy pattern can be defined as the pattern, which not belong to any cluster or belongs to small size cluster.

4.2 Data Transformation

The data transformation phase aims to transform n dimensional data of mixed type into one dimensional numerical data in order to enhance the efficiency of the system (execution time and usage memory) and to solve the challenging of mixed type data set. The following steps have been followed to achieve the objective of this phase:

step1:The k means algorithm can be implemented to classify data patterns Dp={Dp_1, Dp_2, …,Dp_m} to n classes (C1, C2 … Cn) which represent various kind of data flow such as the normal traffic and other types of attacks such as denial of service (DoS), probe (PRB), user to root (U2R), and remote to login (R2L).

step2: The KNN algorithm can be applied to locate the k nearest neighbors (N_1..N_k) for each Dp_i in the same cluster.

step3: After the cluster centers (C_1..C_n) and nearest neighbor (N_1..N_k) for each data pattern of the dataset are founded, two distances are computed and summed as defined in Eq. (6). The first distance ($dist_1$) is the summation of distances between each data pattern Dp_i and the cluster centers (C_1, C_2 … C_n), as illustrated in Eq. (4). The second distance ($dist_2$) is the distance between data pattern Dp_i and the nearest neighbors N_k as illustrated in Eq. (5) and Fig. 3, Note that, the distance between two patterns has been explained in section 2.

$$dist_1 = \sum_{j=1}^{n} distance(Dp_i, C_j) \tag{4}$$

$$dist_2 = \sum_{k=1} distance(Dp_i, N_k) \tag{5}$$

$$distance = dist_1 + dist_2 \tag{6}$$

Where n represents the number of clusters and k is the number of the nearest neighbour which performs the best detection accuracy for KNN classifier. At the end of this phase, the n dimensional data will be summarized into one-dimensional data to solve the problem of mixed type data in IDS data set which leads to maximize system efficiency (usage storage and running time).

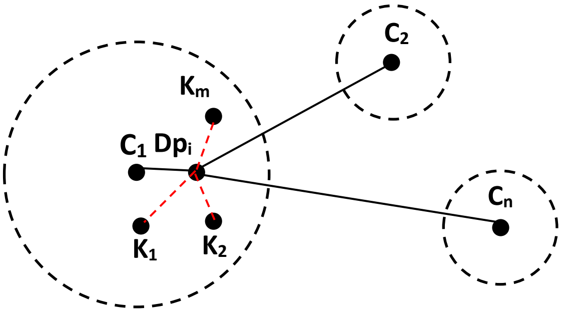

Figure. 3 The new data formation

4.3 KNN classifier

The k-Nearest Neighbor classifier (KNN) has been used in the third phase of IDS-MIU to classify summarized data from the previous phase. The main idea of KNN classifier is that, in a sample space, if most of its *K* closest neighbours samples belong to a class, then the sample belongs to the same class. Note that, The Euclidian distance has been used to compute the closest k Neighbours since the transformed data consists of one dimension. Finally, the KNN classifier has been chosen due to its low computation cost.

5. Experimental results

In order to evaluate the efficiency (running time) and the accuracy of intrusion detection of IDS-MIU system, it has been compared with SVM [23], CANN [24], EIDS-ACC-OD [29], and Random Forest [18] classifiers using the KDD Cup99 data set. The KDD Cup99 consists of 494,020 patterns and each pattern consists of 41 dimensions of mixed type (binary, nominal, and numeric) which represent network connections. Where, 9 dimensions represent the intrinsic types, 13 dimensions are related to content type and the traffic type is represented by 19 dimensions. Table 2 shows some of the dimensions and their type and description. The patterns in KDD cup99 data set are labelled into five classes: a normal data and 4 abnormal attacks, the abnormal attacks can be categorized into four classes, namely remote-to-local (R2l),denial-of-service (Dos), user-to-root (U2r), and Probe (Prb).

Various pre-processing steps have been done such as: firstly, mapping some nominal attributes to numeric-valued. Secondly, mapping some nominal attributes (e.g. "protocol" and “TCP status flag”) to

Table 2. TCP connection Attributes IN KDD'Cup99

Category Name	**Features Names**	**Type**	**Description**
Basic Attributes	Duration	Integer	connection Duration (seconds)
	Protocol type	Nominal	Protocol Type (TCP, UDP, etc.)
	Service	Nominal	Service of Network (http; telnet; others)
Content Attributes	Logged in	Binary	0 if not success to login else 1.
	Number of Failed Login	Interger	Num. of failed logins into a connection
	Root shell	Binary	0 if root shell is not obtained else 1
Traffic Attributes	destination host count	Integer	connections Sum to the same destination IP address
	destination host same srv rate	Real	Percentage of connections that were to the same service, among the connections aggregated into destination host count (P32)

binary attributes. Finally, normalizing some attributes based on Min-Max and Z-score normalization methods since the data attributes have different scales such as "destination host count" which has a range of (0-255), whereas "sourcebytes" ranges from (0-693375630). Note that, theexperiments have been conducted with a Core i3 Intel processor and 8 GB RAM.

5.1 Efficiency evaluation

The efficiency (running time) of IDS-MIU classifier has been compared with SVM, CANN, EIDS-ACC-OD and Random Forest classifiers based on 20 selected dimensions of KDD Cup99. The comparison shows that the running time of IDS-MIU is faster than SVM and Random Forest but slower than CANN and EIDS-ACC-OD as illustrated in Table 3. This is due to using more complicated dissimilarity measure that maintains missing value to improve classification accuracy in incomplete data set as explained in section 3.

5.2 Classification performance

In this section, the classification performance of IDS-MIU, SVM, CANN, EIDS-ACC-OD, and Random Forest classifiers has been tested based on the Detection Rate DER, False Positive Rate FPR, and Accuracy AC metrics [3] as given in Eq. (7),(8), and (9).

$$DER = TP/(TP + FN) \tag{7}$$

$$FPR = FP/(FP + TN) \tag{8}$$

$$AC = (TN + TP)/(TN + TP + FP + FN) \tag{9}$$

Where

Table 3. The running time (mins) of the CANN, SVM, EIDS-ACC-OD, Random Forest and IDS-MIU based on selected 20 dimensions(KDD Cup99)

Classifiers	Data preparation	Training and testing	overall
CANN	159	1480	1639
SVM	-	6155	6155
EIDS-ACC-OD	123	1480	1603
Random Forest	112	2534	2646
IDS-MIU	170	1504	1674

Table 4. Confusion matrix obtained with IDS-MIU for the five classes

	Normal	**Prb**	**R2l**	**Dos**	**U2r**	**Actual**	**Correct %**
Nor	596	60	78	138	80	6053	98.5
Prb	408	36	31	85	0	4164	87.4
R2l	150	87	10	18	59	1624	06.4
Dos	435	15	51	2234	0	2298	97.2
U2r	59	15	15	0	9	234	03.8

- False positive (FP): denoted to the number of normal patterns, which are classified as an attack instances.
- False negative (FN): denoted to the number of attacks patterns, which are classified as normal instances.
- True positive (TP): denoted to the number of detected attacks and in fact they are attacks.
- True negative (TN): denoted to the number of detected normal instances and in fact they are normal.

The confusion matrix which has been constructed by IDS-MIU in the testing phase aims to understand the classification accuracy of IDS-MIU according to the five classes of KDD Cup99 data set as illustrated in Table 4. It is clear that most of normal network

Table 5. Classification results of SVM, CANN, random forest, EIDS-ACC-OD and IDS-MIU for the five classes of KDD cup99

	Metric	**SVM**	**CANN**	**Random Forest**	EIDS-ACC-OD	**IDS-MIU**
Normal	DER%	97.3	96.9	98.7	98.5	**98.6**
	FPR%	8.8	8.6	**8.5**	8.6	**8.5**
Prb	DER%	79.4	86.3	87.0	86.9	**87.2**
	FPR%	**0.45**	**0.45**	0.42	.43	0.8
R2l	DER%	6.4	**7.15**	6.9	7.10	6.25
	FPR%	0.1	0.2	**.09**	0.1	0.21
Dos	DER%	97.1	96.9	97.1	97.05	**97.5**
	FPR%	0.5	0.9	**0.30**	0.33	0.35
U2r	DER%	11.7	**9.15**	5.32	4.32	4.35
	FPR%	**0**	0.1	0.05	0.05	**0**
Overall	AC%	91.5	92.2	**93.3**	92.4	92.9

Table 6. Classification results of SVM, CANN, random forest, EIDS-ACC-OD and IDS-MIU for the five classes KDD cup99 after randomly removing 5% of the data set.

	Metric	**SVM**	**CANN**	Random Forest	EIDS-ACC-OD	**IDS-MIU**
Normal	DER%	96.3	96.1	97.9	97.73	**98.1**
	FPR%	8.6	8.35	8.33	8.50	**8.3**
Prb	DER%	78.4	84.7	86.53	85.70	**86.9**
	FPR%	0.41	0.43	0.41	**0.40**	0.73
R2l	DER%	5.9	6.4	6.65	**6.75**	**6.0**
	FPR%	**0.09**	0.18	0.18	0.17	0.185
Dos	DER%	96.2	96.1	96.30	96.25	**97.1**
	FPR%	0.4	0.8	0.34	0.35	**0.33**
U2r	DER%	**11.2**	8.9	5.15	4.10	4.5
	FPR%	**0**	0.08	0.10	0.04	**0**
Overall	AC%	88.9	89.3	92.1	91.9	**92.2**

Table 7. Classification results of SVM, CANN, random forest, EIDS-ACC-OD and IDS-MIU for the five classes of KDD cup99 after randomly removing 10% of the data set.

	Metric	**SVM**	**CANN**	**Random Forest**	EIDS-ACC-OD	**IDS-MIU**
Normal	DER%	96.9	96.4	96.1	96.0	**97.2**
	FPR%	8.6	8.35	8.40	8.45	**8.3**
Prb	DER%	78.6	84.9	84.95	84.95	**86.0**
	FPR%	**0.41**	0.43	0.40	0.65	0.70
R2l	DER%	5.9	6.4	6.00	6.60	**6.0**
	FPR%	**0.09**	0.18	.08	0.1	0.185
Dos	DER%	96.2	96.1	96.10	96.00	**96.9**
	FPR%	0.4	0.8	**0.27**	0.30	0.33
U2r	DER%	**11.2**	8.9	9.10	8.95	4.30
	FPR%	**0**	0.08	.06	0.07	**0**
Overall	AC%	88.2	88.9	89.7	89.1	**91.2**

records can be classified by IDS-MIU (98.50%) during testing phase. While, the system exhibit poor performance towards U2r (3.8%) and R2l (6.46%) attacks. In addition, three experiments have been done to evaluate the classification accuracy of IDS-MIU classifier for the Normal, Dos, Prb, and Overall Accuracy and compare the results with SVM, CANN, Random forest, and EIDS-ACC-OD classifiers.

The first experiment is based on the actual data set, the results show that the IDS-MIU classifier has the best performance for Prb (87.2%) and Dos (97.5%) and the second best performance for Normal (98.6%) after Random Forest classifier (98.7%). This is due to removing noisy patterns in pre-processing step of IDS-MIU as illustrated in Table 5. Whereas, the IDS-MIU classifier have bad performance for R2l (6.25%) and U2r (4.35). However, the IDS-MIU has the second rank after Random forest classifier according to the overall classification accuracy. The second experiment has been implemented after randomly removing 5% of the KDD Cup99 data set.

The results show that the IDS-MIU classifier has the best performance for Normal (98.1%), Prob (86.90%), Dos (97.1%), and the overall accuracy (92.2%) and sill has bad performance for R2l (6.0%)

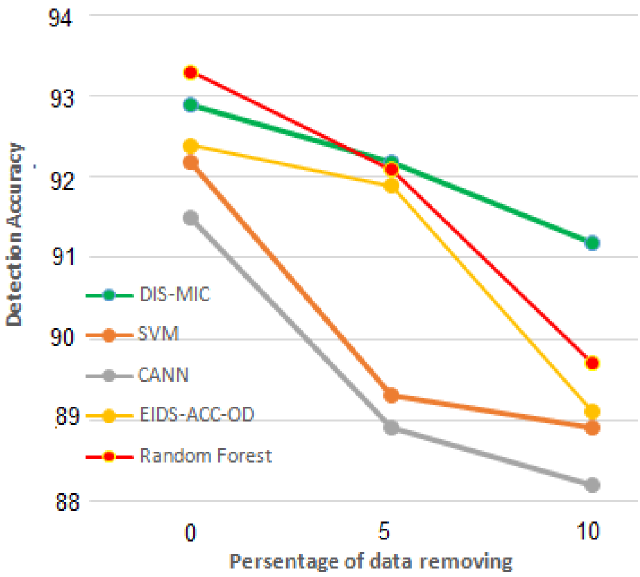

Figure. 4 The dropping in detection accuracy after removing 5 and 10 % of the data

Table 8. The best values for eps, minpts, and K parameters.

	e	*min*		Accu	Detect
IDS-	0	5		98.3	98.9

and U2r (4.5%). Note that, there are a small decrementing in overall classification accuracy for all the classifiers but the IDS-MIU has the best performance as illustrated in Table 6. This is due to using Incomplete and mixed type data dissimilarity in IDS-MIU classifier. In the third experiment, 10% of the KDD Cup99 data set has been randomly discarded. The IDS-MIU still has the best overall classification accuracy (91.2%) but the accuracy gap is increased with the other classifiers SVM (88.2%) , CANN (88.9%), Random Forest (89.7%), EIDS-ACC-OD (89.1%) as shown in Table 7. Ultimately, the results show that IDS-MIU classifier has high accuracy detection than SVM, CANN, Random Forest, and EIDS-ACC-OD when deal with incomplete data set (5, 10%) in spite of dropping the overall accuracy detection as illustrated in Fig. 4.

5.3 Parameter Sensitivity

The IDS-MIU classifier has been implemented many times in order to obtain the typical values for DBSCAN algorithm parameters (eps, minpts) in the pre-processing phase and K for KNN classifier in the second phase which perform the best accuracy detection. The accuracy was tested for parameter values eps (0-1normalized), minpts (2-10), and K (2-10) as shown in Table 8.

6. Conclusions

In this study, IDS-MIU system has been proposed to detect an intrusion in mixed type, incomplete and noisy data set. The system has used a various machine learning techniques to enhance the performance such as: the DBSCAN clustering algorithm to remove noise data, and K-means and KNN classifier to transfer multi dimensions data into one dimension data. The results show that the IDS-MIU system has the best overall classification accuracy (92.2 and 91.2%) when randomly removing 5 and 10% of the data as compared with SVM(88.9 and 88.2 %), CANN (89.3 and 88.9%), Random Forest(92.1 and 89.7%) and EIDS-ACC-OD (91.9 and 89.1%) classifiers. This is due to transfer multi dimensions data into one dimension data and using special similarity measure method which can deal with mixed type and incomplete data.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Supervision and project administration, Musaab; methodology, Musaab, Basim; software, Dina; writing—review and editing, Musaab, Dina.

Acknowledgments

The authors would like to thank Mustansiriyah University (www.uomustansiriyah.edu.iq Baghdad-Iraq) for its support in the present work.

References

[1] B. Gupta, A. Ramesh, and J. Agrawal, "A Comprehensive Survey on Various Machine Learning Methods used for Intrusion Detection System", In: *Proc. of IEEE 9th International Conference on Communication Systems and Network Technologies*, Gwalior, India, pp. 282-289, 2020.

[2] C. Guo, Y. Zhou, Y. Ping, Z. Zhang, G. Liu, and Y. Yang, "A distance sum-based hybrid method for intrusion detection", *Applied intelligence*, Vol. 40, No. 1, pp. 178-188, 2014.

[3] W. Lin, Wei-Chao, S. Ke, and C. Tsai, "CANN: An intrusion detection system based on combining cluster centers and nearest neighbors", *Knowledge-based systems*, Vol. 78, pp. 13-21, 2015.

[4] S. Sahu, S. Kumar, and D. Mohapatra, "A Review on Scalable Learning Approaches on

Intrusion Detection Dataset", In *Proc. of ICRIC Springer*, pp. 699-714, 2019.

[5] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges", Cybersecurity, Vol 2, No. 1, pp. 1-22, 2019.

[6] N. Ugtakhbayar, B. Usukhbayar, and S. Baigaltugs, "A Hybrid Model for Anomaly-Based Intrusion Detection System", In: *Proc. Advances in Intelligent Information Hiding and Multimedia Signal Processing, Springer, Singapore*, pp. 419-431, 2020.

[7] Y. Liao and V. Vemuri, "Use of k-nearest neighbor classifier for intrusion detection", *Computers & Security*, Vol. 21, No. 5, pp. 439-448, 2002.

[8] W. Li, P. Yi, Y. Wu, L. Pan, and J. Li, "A new intrusion detection system based on KNN classification algorithm in wireless sensor network", *Journal of Electrical and Computer Engineering*, Vol., 2014, pp. 1-9, 2014.

[9] M. Raman, N. Somu, S. Jagarapu, T. Manghnani, T. Selvam, K. Krithivasan, and V. Sriram, "An efficient intrusion detection technique based on support vector machine and improved binary gravitational search algorithm", *Artificial Intelligence Review*, Vol. 53, No. 8, pp. 1-32, 2019.

[10] S. Krishnaveni, P. Vigneshwar, S. Kishore, B. Jothi, and S. Sivamohan, "Anomaly-Based Intrusion Detection System Using Support Vector Machine", *Artificial Intelligence and Evolutionary Computations in Engineering Systems*, pp. 723-731, 2020.

[11] M. Schultz, E. Eskin, F. Zadok, and S. Stolfo, "Data mining methods for detection of new malicious executables", In: *Proc. of IEEE Symposium on Security and Privacy*, Oakland, USA, pp. 38-49, 2000.

[12] Y. Chew, S. Ooi, K. Wong, and Y. Pang, "Decision Tree with Sensitive Pruning in Network-based Intrusion Detection System", In: *Proc. of Computational Science and Technology*, Springer, Singapore, pp. 1-10, 2020.

[13] S. Mousavi, V. Majidnezhad, and A. Naghipour, "A new intelligent intrusion detector based on ensemble of decision trees", *Journal of Ambient Intelligence and Humanized Computing*, Vol. 56, No. 1, pp. 1-13, 2019.

[14] T. Ahmad, M. Anwar, and M. Haque, "Machine Learning Techniques for Intrusion Detection", *Handbook of Research on Intrusion Detection Systems*, IGI Global, pp. 47-65, 2020.

[15] A. Saleh, F. Talaat, and L. Labib, "A hybrid intrusion detection system (HIDS) based on prioritized k-nearest neighbors and optimized SVM classifiers", *Artificial Intelligence Review*, Vol. 51, No. 3, pp. 403-443, 2019.

[16] G. Mylavarapu, J. Thomas, and A. Kumar, "Real-Time Hybrid Intrusion Detection System Using Apache Storm", In: *Proc. of IEEE 12th International Conference on Embedded Software and Systems*, New York, USA, pp. 1436-1441, 2015.

[17] W. Al-Yaseen, Wathiq, Z. Othman, and M. Nazri, "Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system", *Expert Systems with Applications*, Vol. 67, pp. 296-303, 2017.

[18] N. Kaja, A. Shaout, and D. Ma, "An intelligent intrusion detection system", *Applied Intelligence*, Vol. 49, No. 9, pp. 3235-3247, 2019.

[19] B. Setiawan, S. Djanali, and T. Ahmad, "Increasing accuracy and completeness of intrusion detection model using fusion of normalization, feature selection method and support vector machine", *International Jorurnal of Intelligent Engineering and Sys*tems, Vol. 12, No. 4, pp. 378-389, 2019.

[20] B. Kumar, M. Raju, and B. Vardhan. "Enhancing the performance of an intrusion detection system through multi-linear dimensionality reduction and Multi-class SVM", *International Journal of Intelligent Engineering and Systems*, Vol. 11, No. 1, pp. 181-192, 2018.

[21] G. Dong, Y. Jin, S. Wang, W. Li, Z. Tao, and S. Guo, "DB-Kmeans: An Intrusion Detection Algorithm Based on DBSCAN and K-means", In: *Proc. of 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Matsue, Japan, pp. 1-4, 2019.

[22] Z. Chen and Y. Li, "Anomaly detection based on enhanced DBScan algorithm", *Procedia Engineering*, Vol. 15, pp. 178-182, 2011.

[23] C. Guo, Y. Zhou, Y. Ping, Z. Zhang, G. Liu, and Y. Yang, "A distance sum-based hybrid method for intrusion detection", *Applied Intelligence*, Vol. 40, No. 1, pp. 178-188, 2014.

[24] W. Lin, S. Ke, and C Tsai, "CANN: An intrusion detection system based on combining cluster centers and nearest neighbors", *Knowledge-based systems*, Vol. 78, pp. 13-21, 2015.

[25] B. Manjunatha and P. Gogoi, "Anomaly based intrusion detection in mixed attribute dataset using data mining methods", *Journal of*

Artificial Intelligence, Vol. 9, No. 1-3, pp. 1-11, 2016.

[26] V. Shakya and R. Makwana, "Feature selection based intrusion detection system using the combination of DBSCAN, K-Mean++ and SMO algorithms", In: *Proc. of international conference on trends in electronics and informatics (ICEI)*, Tirunelveli, India, pp. 928-932, 2017.

[27] K. Bhosale, M. Nenova, and G. Iliev, "Modified Naive Bayes Intrusion Detection System (MNBIDS)", In: *Proc. of International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, Belgaum, India, pp. 291-296, 2018.

[28] J. Hussain and S. Lalmuanawma, "Feature analysis, evaluation and comparisons of classification algorithms based on noisy intrusion dataset", *Procedia Computer Science*, Vol. 92, pp. 188-198, 2016.

[29] S. Sandosh, V. Govindasamy, and G. Akila, "Enhanced intrusion detection system via agent clustering and classification based on outlier detection", *Peer-to-Peer Networking and Applications*, Vol. 13, No. 3, pp. 1038-1045, 2020.

[30] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise", *Journal of Knowledge Discovery and Data Mining KDD*, Vol. 96, No. 34, pp. 226-231, 1996.

[31] C. Cassisi, A. Ferro, R. Giugno, G. Pigola, and A. Pulvirenti, "Enhancing density-based clustering: Parameter reduction and outlier detection", *Information Systems*, Vol.38, No. 3, pp. 317-330, 2013.

[32] X. Zuo, Z. Chen, L. Dong, J. Chang, and B. Hou, "Power information network intrusion detection based on data mining algorithm", *The Journal of Super computing*, pp. 1-19, 2019.

[33] J. Han, M. Kamber, and J. Pei, "Data mining: concepts and techniques", *The Morgan Kaufmann Series in Data Management Systems*, Vol. 5, No.4, pp. 83-124, 2011.

[34] M. Riyadh, N. Mustapha, and D. Riyadh, "Review of Trajectories Similarity Measures in Mining Algorithms", In: *Proc. Al-Mansour International Conference on New Trends in Computing, Communication, and Information Technology (NTCCIT)*, Baghdad, Iraq, pp. 36-40, 2018.