Pattern Recognition Letters 157 (2022) 16–26

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Cloud security based attack detection using transductive learning integrated with Hidden Markov Model

Yassine Aoudni[a,*], Cecil Donald[b], Ahmed Farouk[c], Kishan Bhushan Sahay[d], D. Vijendra Babu[e], Vikas Tripathi[f], Dharmesh Dhabliya[g]

[a] Faculty of sciences and arts, Turaif, Northern Border University, Arar 91431, Kingdom of Saudi Arabia
[b] Department of Computer Science, CHRIST (Deemed to Be University), Bangalore, India
[c] Department of Computer Science, Faculty of Computers and Artificial Intelligence, South Valley University, Hurghada, Egypt
[d] Department of Electrical Engineering, Madan Mohan Malaviya University of Technology, Gorakhpur, Uttar Pradesh 273010, India
[e] Aarupadai Veedu Institute of Technology, Chennai, Vinayaka Mission's Research Foundation, Deemed to be University, Chennai, Tamilnadu, India
[f] Department of Computer Science & Engineering, Graphic Era Deemed to be University Dehradun, Uttarakhand, India
[g] Department of Computer Engineering, Vishwakarma Institute of Information Technology, Pune, India

ARTICLE INFO

Article history:
Received 10 November 2021
Revised 31 January 2022
Accepted 26 February 2022
Available online 28 February 2022

Edited by: Maria De Marsico.

Keywords:
Zero-day attack security
Transductive deep learning
Hidden Markov Model (HMM)
Soft labels
K-medoids clustering

ABSTRACT

In recent years, organizations and enterprises put huge attention on their network security. The attackers were able to influence vulnerabilities for the configuration of the network through the network. Zero-day (0-day) is defined as vulnerable software or application that is either defined by the vendor or not patched by any vendor of organization. When zero-day attack is identified within the network there is no proper mechanism when observed. To mitigate challenges related to the zero-day attack, this paper presented HMM_TDL, a deep learning model for detection and prevention of attack in the cloud platform. The presented model is carried out in three phases like at first, Hidden Markov Model (HMM) is incorporated for the detection of attacks. With the derived HMM model, hyper alerts are transmitted to the database for attack prevention. In the second stage, a transductive deep learning model with k-medoids clustering is adopted for attack identification. With k-medoids clustering, soft labels are assigned for attack and data and update to the database. In the last phase, with computed HMM_TDL database is updated with computed trust value for attack prevention within the cloud.

1. Introduction

In cloud security, detecting attack software is considered as essential tasks. Among several attack types, a zero-day attack is considered as most problematic because the antivirus cannot able to remove. Even in a well-secured network, with persistent probing attacker able to identify loophole and perform attack within the network [1]. It is exploited that the targeted network gains entry of the actor threat [2]. As a result, to examine the zero-day exploits are difficult for identification [3]. On other hand, hacker incorporates vulnerability with zero-day attack facilitate then in terms of; providing information about the detected vulnerability to software vendor; an attacker may sell high crucial data to the broker, those may incorporate the highest rate of exploits [4]. Zero-data utilizes several elements that are all not identified previously; an attacker will incorporate zero-day in vulnerabilities charted list and with processing program payloads are mixed, and leads to the influence of attack in server [5,6]. Attackers always trace networks to identify weaknesses for gaining access through the network with the execution of their codes. In this manner, the network is breached and subjected to a zero-day attack [7]. When zero-day attack is identified within the network there is no proper mechanism when observed [8]. Therefore, identification of a zero-day attack is considered a challenging task [9]. To detect and prevent zero-day attacks, this paper proposed a model stated as Hidden Markov Model Transductive Deep Learning (HMM_TDL), which generates hyper alerts when an attack is implemented. Also, the HMM_TDL assigns labels to data in the network and periodically update to the database (DB). Initially, the HMM model detects the attacks with hyper alerts in the database. In the next stage, transductive deep learning incorporates k-medoids for clustering attacks and assign labels. Finally, the trust value of the original data is computed and computed in the database based on the value network able to classify attacks and data. The remainder of the paper is organized as

[*] Corresponding author.
E-mail addresses: yassineaoudni@hotmail.com (Y. Aoudni), cecil.donald@christuniversity.in (C. Donald), ahmed.farouk@sci.svu.edu.eg (A. Farouk), kbsee@mmmut.ac.in (K.B. Sahay), vijendrababu@avit.ac.in (D.V. Babu), vikastripathi.cse@geu.ac.in (V. Tripathi), dharmesh.dhabliya@viit.ac.in (D. Dhabliya).

https://doi.org/10.1016/j.patrec.2022.02.012
0167-8655/© 2022 Published by Elsevier B.V.

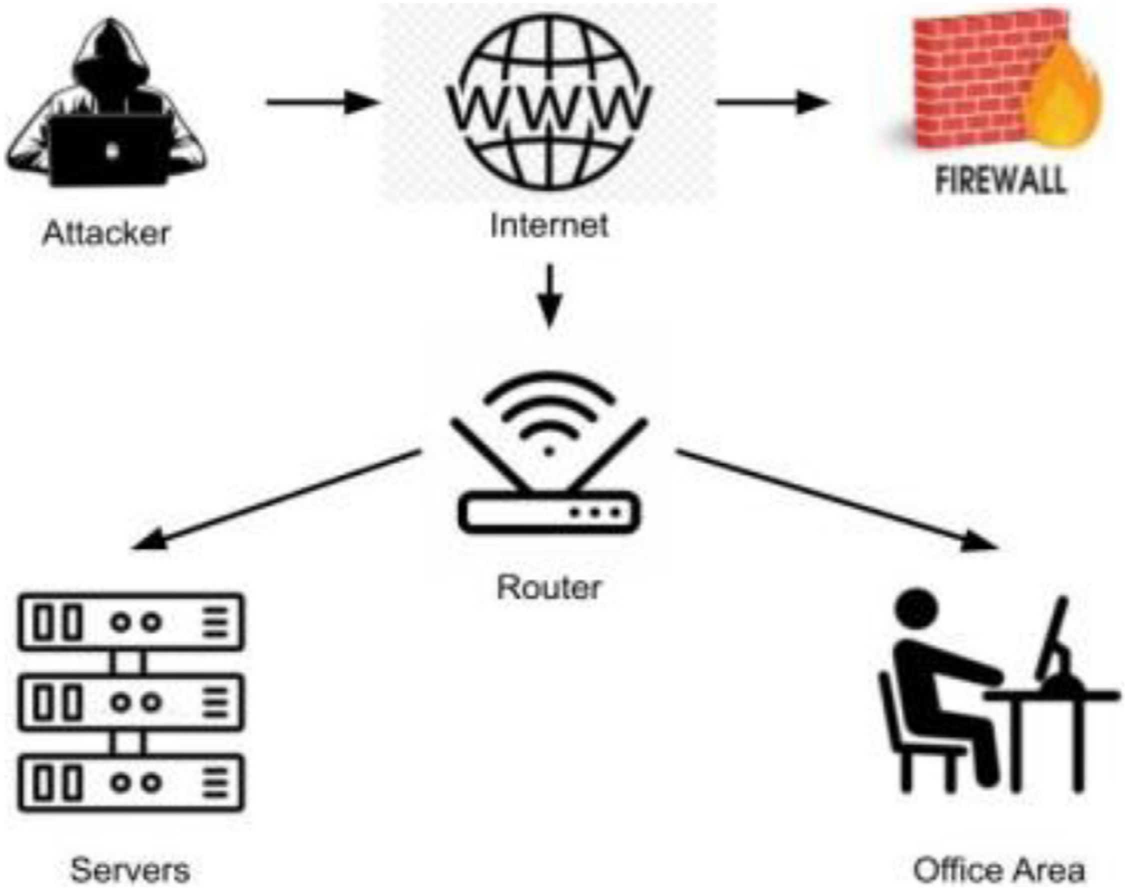

Fig. 1. The typical process in data stealing.

follows: section II presented a review of conventional techniques on attack detection. In section III problem domain related to the zero-day attack and model framework are presented. Section IV described developed HMM_TDL for attack prevention. The results obtained for developed HMM_TDL are presented in Section V with comparative analysis. Finally, in section VI provides the overall conclusion of developed HMM_TDL.

2. Related works

This section presented existing proposed methods for securing a zero-day attack. Through analysis, existing security schemes are categorized as signature-based, hybrid detection, statistical-based, and behavior-based approach [10]. In statistical-based techniques, attack profile is generated from past elements. Through those identified elements, the profile parameters are updated in historical exploits for the detection of attacks [11]. However, these statistical-based techniques cannot be able to apply instantaneously for attack protection and detection [2].

A behavior-based mechanism derives worms characteristics for accurate estimation of attack in the webserver. With the identification of attacks in the server machine able to identify victims or deny services that are not involved [12]. In a hybrid-based mechanism, signature schemes are integrated based on the applications [13]. Based on this, Kaur and Singh [14] developed a zero-day attack identification using a hybrid approach. To evaluate vulnerabilities risk levels Hazard metrics are developed in [15,16]. In [17,18] estimated hazard level frequency and impact factors. However, the existing technique is subjected to difficulty due to the dynamic characteristics of attacks. Hence, in this paper developed HMM_TDL for attack detection for improving the robustness of cloud platform.

3. Problem domain and model framework

An attack is considered as complex behaviour with several goals based on that attack phases are implemented. In Fig. 1 illustrates, attackers are engaged to steal data by exploiting the vulnerabilities within the intranet. In the next stage, Trojans are injected into to file server via a network file interface system. This inclusion of Trojan within the system leads to data leakage. In Fig. 2 zero-day attack in the cloud server is illustrated.

Initially, to train the model with exploitation of attack vulnerabilities historical alerts are defined as $\{ah_1, ah_2, ah_3\}$. In Fig. 3, real-time evidence of data stream for multiple results and its corresponding probabilities are presented. For every intent of attack, through associated probabilities next attack probability can be defined.

3.1. Model framework

The overall framework of proposed zero-day attack detection model is stated in Fig. 3. The proposed model comprises of four phases such as; (1) data collection - obtain intrusion dataset; (2) pre-processing of data - Training and testing of pre-processed data; (3) Training Model - Provides historical alert to data for processing; and (4) Testing Model - Real time attack data were processed. Initially, Hidden Markov Model (HMM) is provided with transfer relationship between variables to provide temporal alerts to data. In next stage, HMM two-layer model is converted in to Bayesian network with set of rules with probabilistic inference. This HMM model with probabilistic inference uses transductive learning for rule update. By this, attacks within the network with each time is identified for unknown attacks.

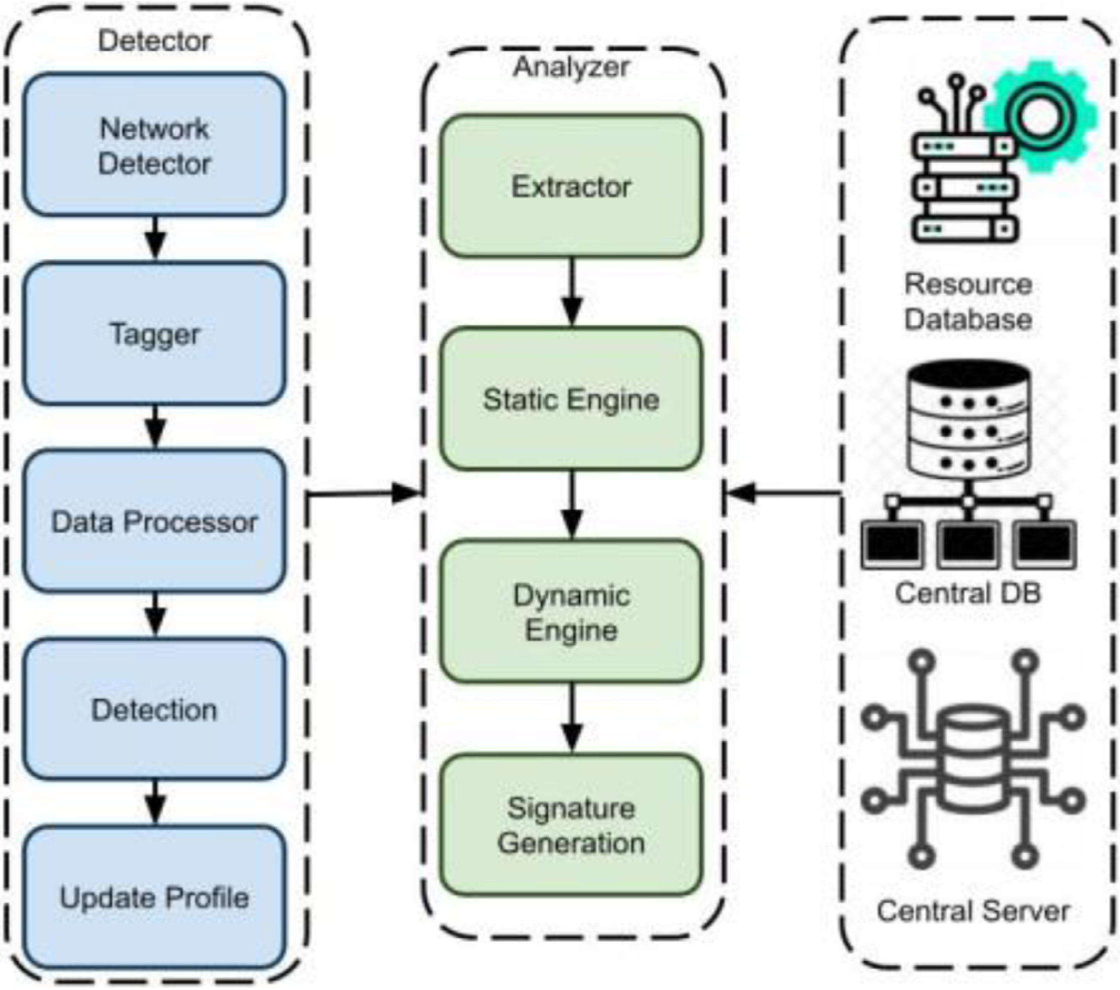

Fig. 2. Zero-day attack scenario.

4. Developed HMM_TDL for zero-day attack

In this section, presented about detailed description of developed HMM_TDL for zero-day attack detection and prevention. Initially, HMM is explained for attack detection with hyper alerts. In next sub-section, presented about transductive model for soft labeling is described. Finally, overall performance of developed HMM_TDL is presented with algorithm.

4.1. Hidden Markov Model for attack detection

As defined, HMM has been applied in vast range of applications like image recognition, biology, signal processing and so on. HMM is applied in intrusion detection system.

This provides significant relationship between undefined sequences and attack with temporal estimation. With application of HMM malicious attacks in cloud are identified with transfer probabilities as presented in Fig. 4.

In HMM model the state are represented as s at time instance t with number of node as n_i. The HMM model is defined using following Eq. (1)

$$\lambda = (B, H, C, J, \pi) \tag{1}$$

As defined in Fig. 4 observation sequences are measured in observation layer and it is denoted as $N = (n_1, n_2, \ldots.., n_T)$. At certain time instances, attack are intents are denoted as $n_t = v_i$. In hidden layer attack intents are denoted as $H(i_1, i_2, \ldots., i_T)$. Also, at some time instances $i_t = S_i$. The conditional probability of attack sequences and intents are given as $P(al_i|S_i)$ for vertical attacks. Similarly, for horizontal attack intents conditional probability of $S_j$as $P(S_j|S_i)$.

The alerts are generated with attack sequences with random length of $L = (n_1, n_2, \ldots.., n_M)$. In this, M is defined as positive integer value given by $N = (al_1, al_2, \ldots., al_n)$. The hidden layer attack intents are given as H. The probabilistic model is represented as follows Eq. (2):

$$P(N|\lambda) = \sum_I P(N, I|\lambda) = \sum_I P(N|I, \lambda)P(I|\lambda) \tag{2}$$

To minimize intents each attack intents are evaluated based on consideration of forward and backward intents. The hidden state time is represented as t_i. For the forward probability $\alpha_t(i)$the observed node are represented as $n_1, n_2, \ldots.., n_t$. For same hidden state time $t+1$, the corresponding observed states are denoted as $n_{t+1}, n_{t+2}, \ldots.., n_T$. The backward probability $\beta_t(i)$ is derived as follows stated in Eqs. (3) and (4):

$$\alpha_t(i) = P(n_1, n_2, \ldots., n_t, i_t = q_i|\lambda) \tag{3}$$

$$\beta_t(i) = P(n_{t+1}, n_{t+2}, \ldots., n_T|i_t = q_i, \lambda) \tag{4}$$

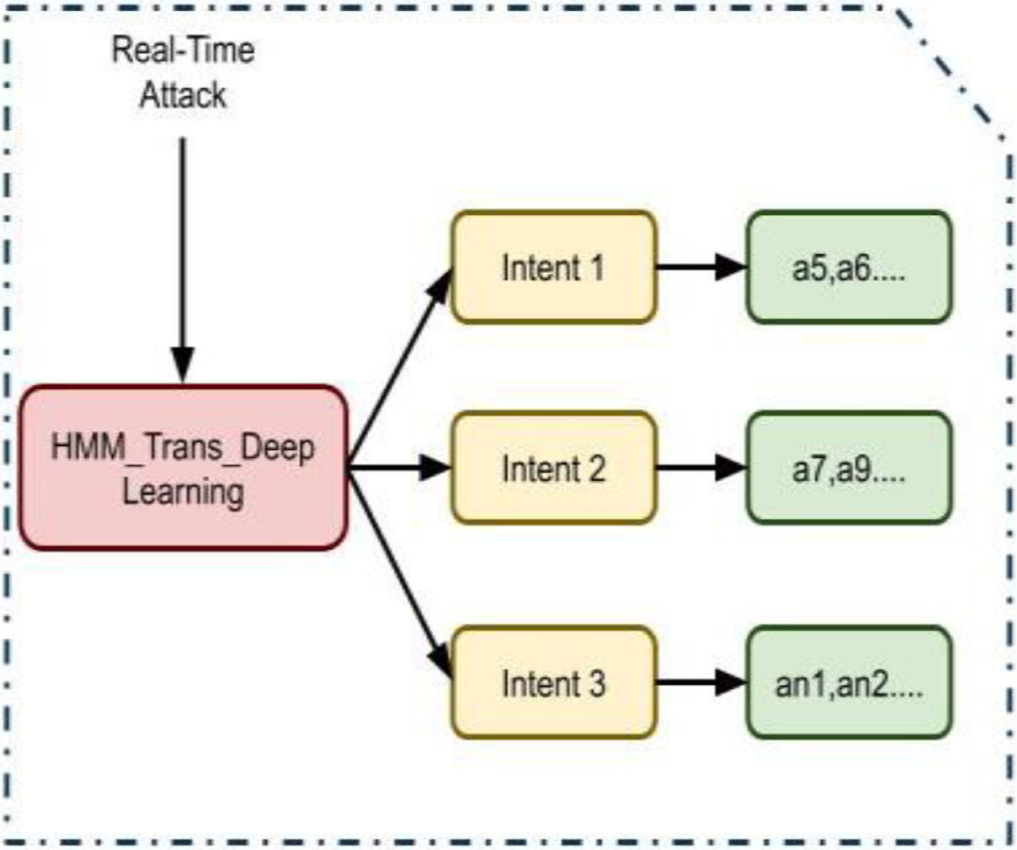

Fig. 3. Overview of proposed HMM_TDP.

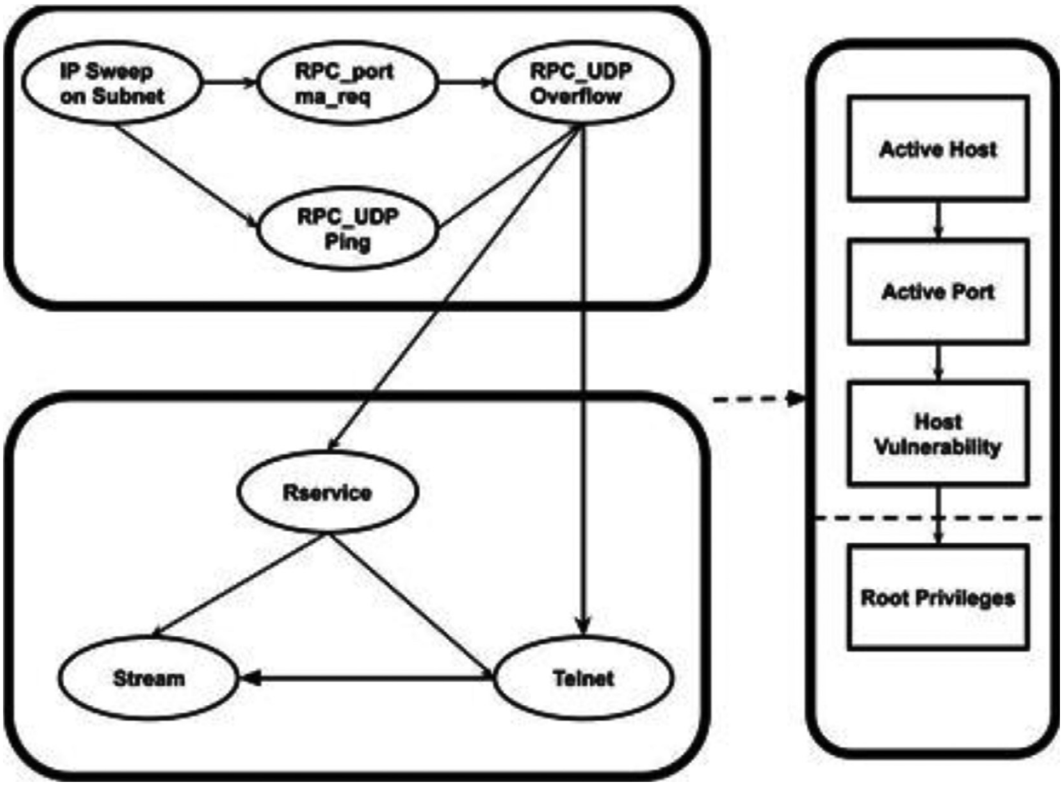

Fig. 4. HMM model for attack detection.

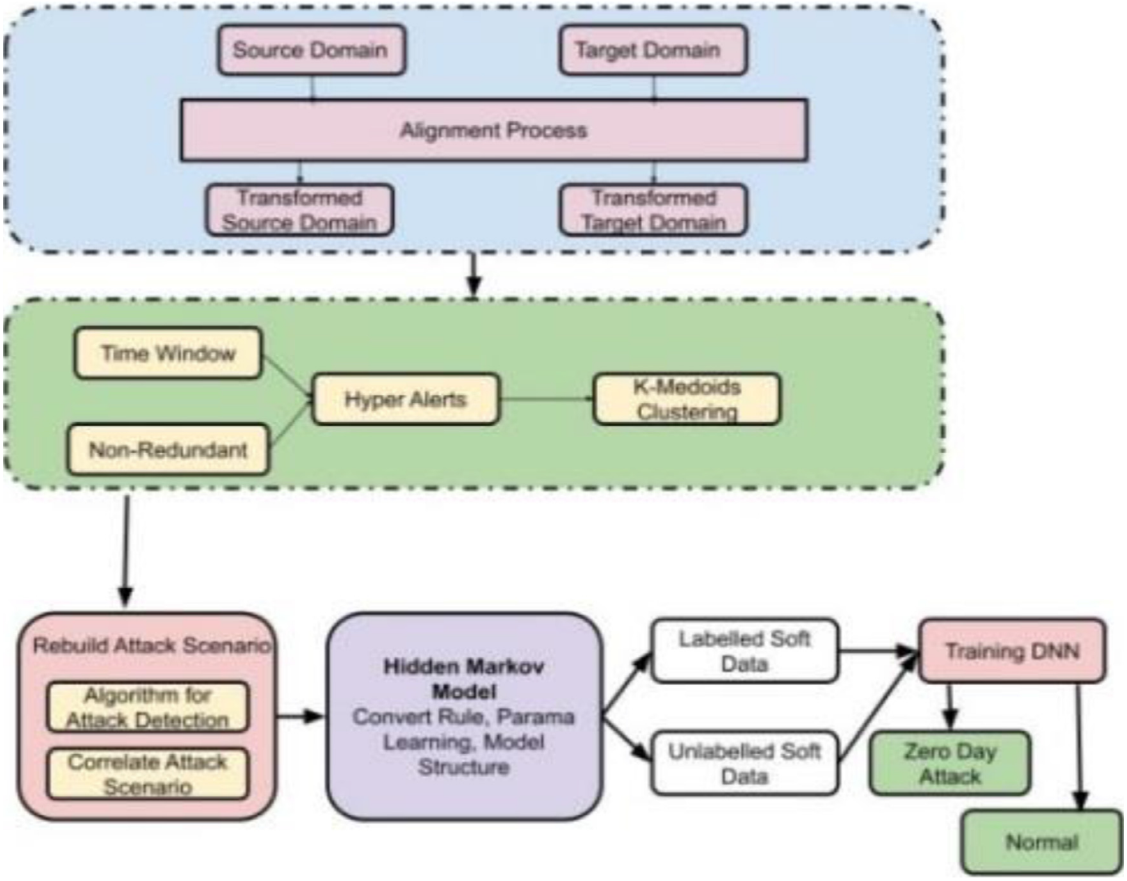

Fig. 5. Overall Architecture of HMM_TDL.

The probability state is presented as in Eq. (5)

$$\gamma_t(i) = P(i_t = q_i|N, \lambda) = \frac{P(i_t = q_i, N|\lambda)}{P(N|\lambda)} \tag{5}$$

For backward probability it is given in Eq. (6),

$$P(i_t = q_i, N|\lambda) = \alpha_t(i)\beta_t(i) \tag{6}$$

Which provides,

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N}\alpha_t(j)\beta_t(j)} \tag{7}$$

For the defined model λ, the sequence of observation is given as N, the probability of state q_i and $q_j$with time t and $t+1$ is given in Eqs. (8)–(10)

$$\delta_t(i,j) = P\big(i_t = q_i, i_{t+1} = q_j\big|N, \lambda\big) = \frac{P\big(i_t = q_i, i_{t+1} = q_j\big|N, \lambda\big)}{P(N|\lambda)} \tag{8}$$

$$P\big(i_t = q_i, i_{t+1} = q_j, N|\lambda\big) = \alpha_t(i)a_{ij}b_j(N_{t+1})\beta_{t+1}(j) \tag{9}$$

$$\delta_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(N_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(N_{t+1})\beta_{t+1}(j)} \tag{10}$$

The simplified equation is presented as in Eq. (11),

$$P(N|\lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(N_{t+1})\beta_{t+1}(j) \tag{11}$$

Through this HMM construct a trained model as $\lambda = (C, J, \pi)$

4.2. Transductive transfer learning framework

In next phase, this paper constructed transductive transfer learning (TL) framework for prevention of zero-day attack. The constructed TL perform binary classification with leverage of labeled instances denoted as L_s. To prevent zero-day attack A_s. In this, cluster or group without any attack is defined as λ. The developed transductive model is applied in HMM model $\lambda = (C, J, \pi)$ for detection and prevention of zero-day attack. In Fig. 5 overall architecture of HMM_TDL is presented.

4.2.1. Attack prevention with HMM_TDL

The relationship exists with i$^{\text{th}}$ cluster leads to construction of cluster of length $(i-1^{th}\text{and}i+1^{th})$ for other cluster domains. Based on this, i^{th} cluster similarity matrix is constructed as $M_S^{p,q}$. For the i^{th} cluster domain p and i^{th} domain of q provides the value of 1. Similarly, for $(i-1)^{th}$ and $(i+1)^{th}$ for domain p and q assigned values as 0.5. For other it assigned with 0. The source domain transformation is performed for target mapping function with consideration of latent space $d$between attack scenario. The domain attack source domain and targeted domain is represented as D_1^t and D_2^t with assigned Euclidean distance value of. The i^{th} cluster group source label is ranked as r_i, r_i+1, r_i-1, with HMM model and labeled with following steps such as:

Step 1: Initially, set label value as zero

Step 2: If cluster source ranked as $r_i$the attack $\alpha$is included else it will eliminated from the cluster group.

Step 3: For source cluster ranked as r_i+1, then attack $\frac{\alpha}{2}$ is included within the system else it will be removed from cluster group.

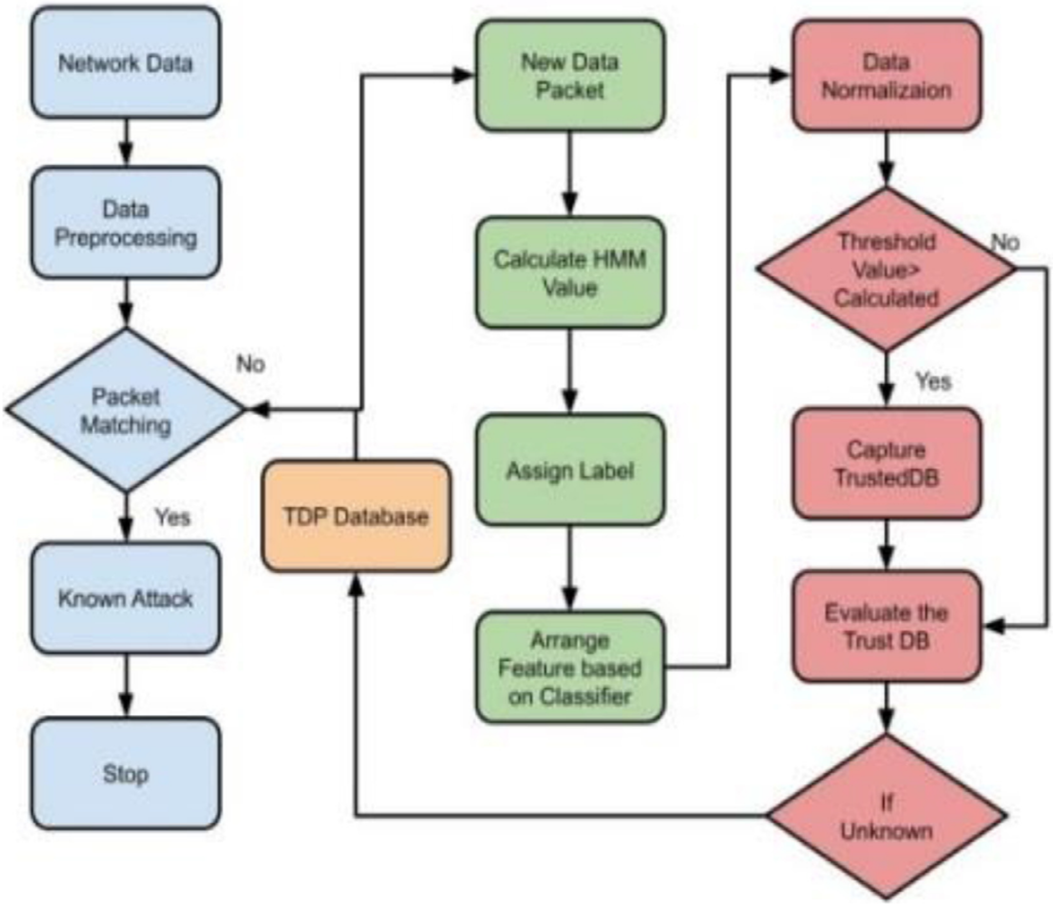

Fig. 6. Overall Flow of HMM_TDL.

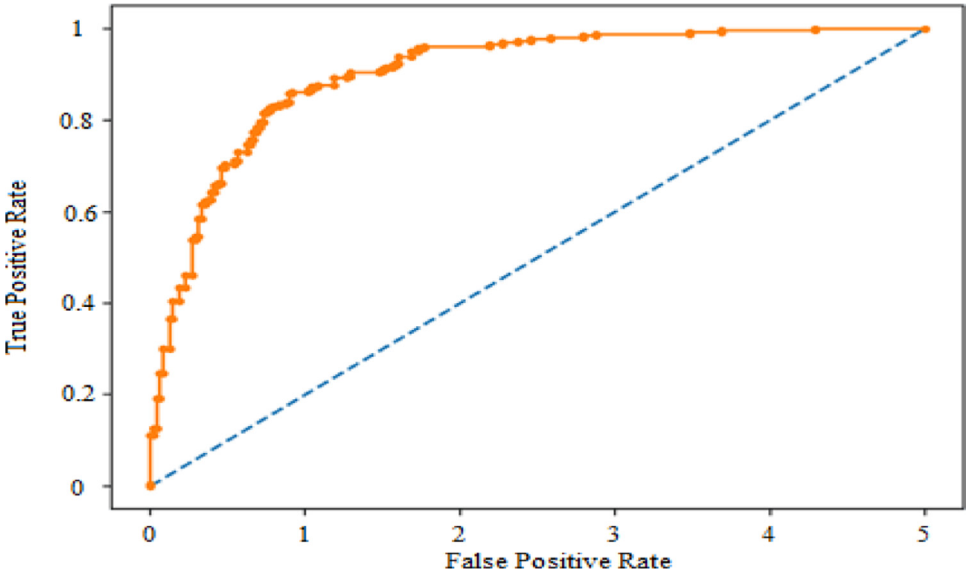

Fig. 7. ROC curve for DoS→R2L with AUC = 0.83.

Step 4: For source cluster ranked as $r_i - 1$, then attack $\frac{\alpha}{2}$ is included within the system else it will be removed from cluster group.

Finally, with estimation of target scores values of 0 and 1 are normalized with estimation of attack or normal cluster instances. The assigned soft labels for threshold instances of threshold T_1 is considered as "attack" else T_2 is set as below threshold value is defined as "normal". The target instances are denoted as follows: $T_1 = \alpha$ \\ Set as attack label, $T_2 = 1 - \alpha$ \\ Set as normal label

The node within the cluster consists of following factors such as a prior knowledge, edge probability and conditional probability table (CPT). The HMM involved in estimation of attacks in the network with consideration of causality. This paper uses HMM model with transductive transfer learning. Hence, based on the assigned labels unknown attack labels are estimated with update of rule in Eq. (12)

$$U_{ij}^{t} = \begin{cases} \delta + (1-\delta)U_{ij}^{t-1}, P(u_i|y_t) = 1P\left(u_j\middle|y_t\right) = 1 \\ (1-\delta)U_{ij}^{t-1}, P(u_i|y_t) = 1P\left(u_j\middle|y_t\right) = 0 \\ \mathrm{U_{ij}^{t-1}}\text{otherwise} \end{cases} \tag{12}$$

With respect to assigned label T_1 and T_2 the unknown attack is estimated with consideration of CPT attack as $U_{ij} = P(X = x_j|U = u_i)$. In Fig. 6 overall flow of developed HMM_TDL is illustrated with attack detection and prevention mechanism. Once the attack is detected by HMM model transductive network is trained for attack based on trust computational value of database.

For construction of label assigned with unknown attack new dataset is constructed with elimination of attacks defined as $D(D = \{y_1, y_2, y_3 \ldots\ldots\})$ with estimation of missing data attack as y_t. The attack scenario with HMM model for attack scenario is defined as $S = (I_1, I_2, \ldots\ldots I_n)$. Based on assigned label in transductive deep learning HMM estimate unknown attacks in zero-day. The Algorithm 1 provides the HMM for attack estimation is presented.

In Algorithm 2 label assigned for zero-day attack estimation is presented.

The estimation of zero-day attack involved in HMM model for attack estimation. HMM involved in zero-day attack detection with consideration of threshold in calculation. To characterize zero-day attack it is defined as A_{attack}. The real-data in the network is defined as D, hence the first order training in real-time data is given as $D(S(z))$ the term Sprovides transductive layer with value of 1. The second order of transductive neural network is about maximizing $V(S, D)$ in this $S$is layer and $D$is trained data. Similarly, for trained data $D(S(z))$ with second order 1 minimization of zero-day attack or prevention of attack is performed with $\min_{S}\max_{D} V(D, S) = E_x - A_{attack(x)}[\log D(x)] + E_{Z-p_z(Z)}[\log(1 - D(S(z)))]V(S, D)$ as

Algorithm 1 Parameter estimation.

```
**Input:** Sequence of attack in network = $\{(al_1, al_2 \ldots..)(al_3, al_4 \ldots..)\ldots.\}$
Output: $\delta^{n+1} = (C^{n+1}, J^{n+1}, \lambda^{n+1})$
// Start
For n = 0 generate $\delta_i^0$
For $a_{ij}^0 = \delta_i^0$ set $b_j(k)^0$
For unknown attack n =0,1,2…..
do
Calculate using Eq. (11)
Calculate using Eq. (12)
      End for
      End for
Set values
Set comparison value
      If $P(Z_i = 1|I_i = 1) > trustValue$
        then
Compare $T_1$ and $T_2$ with estimated value
      End if
          for $value(Z_i)$compare attack value
          If $value(Z_i) > trustValue$ then
          Compute new set $Z_i$
          End if
        End for
      End for
```

Algorithm 2 Parameter estimation.

```
**Input:** Estimation of target domain
Output: Label with consideration of attacks with d-dimension latent space
1. Construct matrix for data node $Z = \begin{pmatrix} X_1 & 0\ldots\ldots & 0 \\ \ldots\ldots & \ldots\ldots\ldots & \ldots\ldots \\ 0\ldots\ldots & 0 & X_k \end{pmatrix}$
2. Estimate matrix $W_k$ with estimation of distance between nodes
$W_k(i.j) = e^{-\|x_i - x_j\|^2}$
3. Compute Laplacian transformation with similarity indices
$L_x = D_x - W_x$
4. Estimate the Laplacian matric for computing attacks in node
$L = \begin{pmatrix} L_1 & 0\ldots\ldots & 0 \\ \ldots\ldots & \ldots\ldots\ldots & \ldots\ldots \\ 0\ldots\ldots & 0 & L_k \end{pmatrix}$
5. Compute dimensionality of mapping function for computing eigen value $Z(\delta L + L_s)Z^T x = \delta Z L_d Z^T x$
6. Transform target domain with consideration of latent data dimensions
```

The $V(D,S)$ involved in formulation of convergence with estimation of distributed data. The transductive learning for attack detection is computed as (Z_{ij}). It is defined as in Eq. (13)

$$Z_{ij} = \sum_{a=0}^{m-1}\sum_{b=0}^{m-1} w_{ij}x_{ij} \tag{13}$$

To compute zero-day attack in from the TDL it is given as in Eqs. (14) and (15),

$$\min_{S} \log\left(1 - D(S(Z))\right) \tag{14}$$

$$\Leftrightarrow D(S(Z)) \approx 1 \tag{15}$$

$\Leftrightarrow S(z) \approx x \in \chi$; the real-time dataset is denoted as χ

The complete training data from HMM-TDL is stated as in Eq. (16)

$$P(D^*, S^*) \le P(D^*, S)\forall S \tag{16}$$

The attack detected from HMM is incorporated in transductive deep learning model which subjected to two constraints such as defined as follows from Eqs (17)–(27):

Case 1: If $A_{attack} \approx A_s$ for training all data

Consider $P(D^*, S^*)$ as minimal value for data $A_{attack}(x) \le A_s(x)$

This will be $A_{attack}(x) \le A^*{}_S(x)$

$$P(D^*, S) = \int_x A_{attack}(x)D^*(x)dx + \int_x A_{data}(x)D^*(S(z))dz \tag{17}$$

$$= \int_x A_{attack}(x)D^*(x) + A_S(x)(1 - D^*(x))dx \tag{18}$$

$$= \int_x A_{attack}(x)D^*(x) - p_S(x)dx + \int_x p_s(x)dx \tag{19}$$

$$= \int_x 1A_{attack}(x) > A_s(x)(A_{attack}(x) - A_s(x))dx + \int_x A_s(x)dx \tag{20}$$

$$\ge \int_x A_S(x)dx \tag{21}$$

Case 2: Consider $P(D^*, S^*)$ as minimal value for data $A_{attack}(x) \ge A_s(x)$

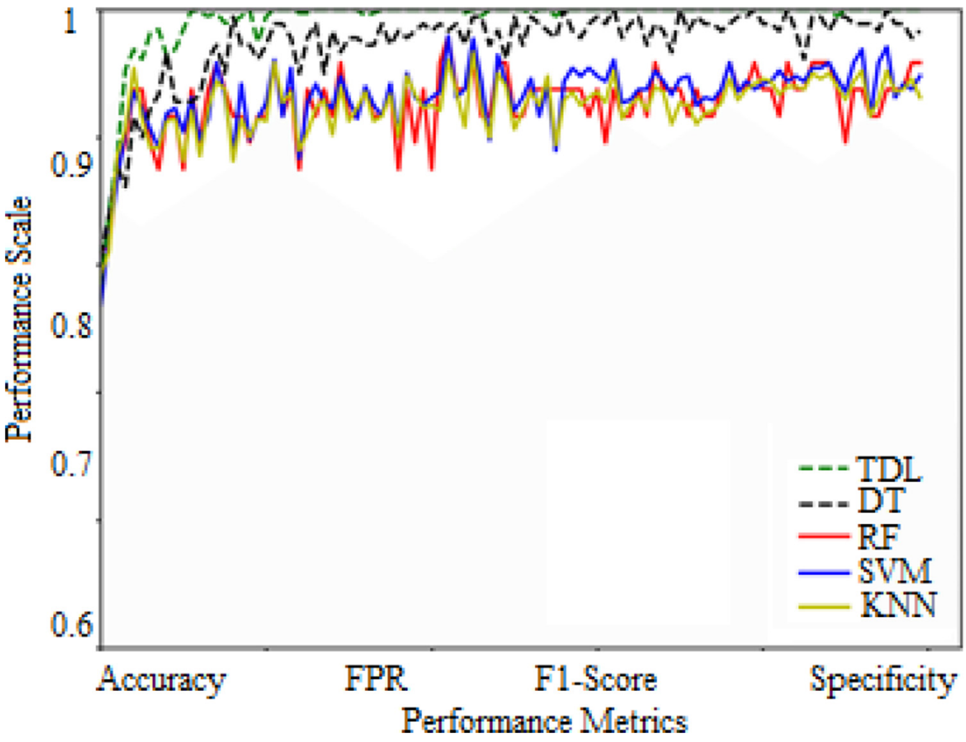

Fig. 8. Overall Comparison of HMM_TDL in DoS→R2L.

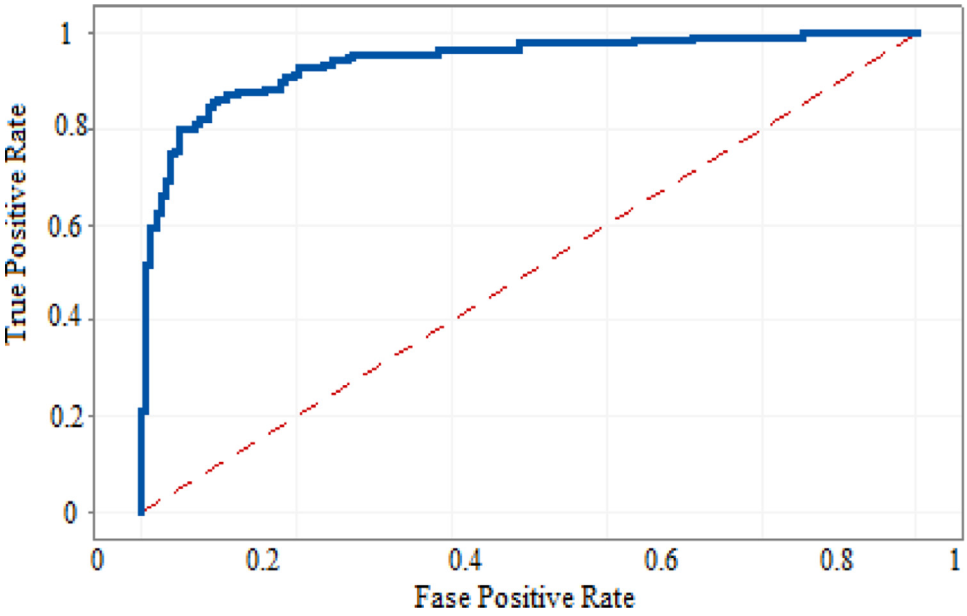

Fig. 9. ROC curve for DoS→Probe.

Table 1
Attack modules for HMM_TDL.

Parameters	DoS→R2L	DoS→Probe	NSL_DoS→CIDD
Cluster before transformation	12	12	24
Cluster after transformation	24	16	24
α	0.9124	0.9254	0.9145
T_1	0.91459	0.9156	0.9324
T_2	0.03144	0.0258	0.0097

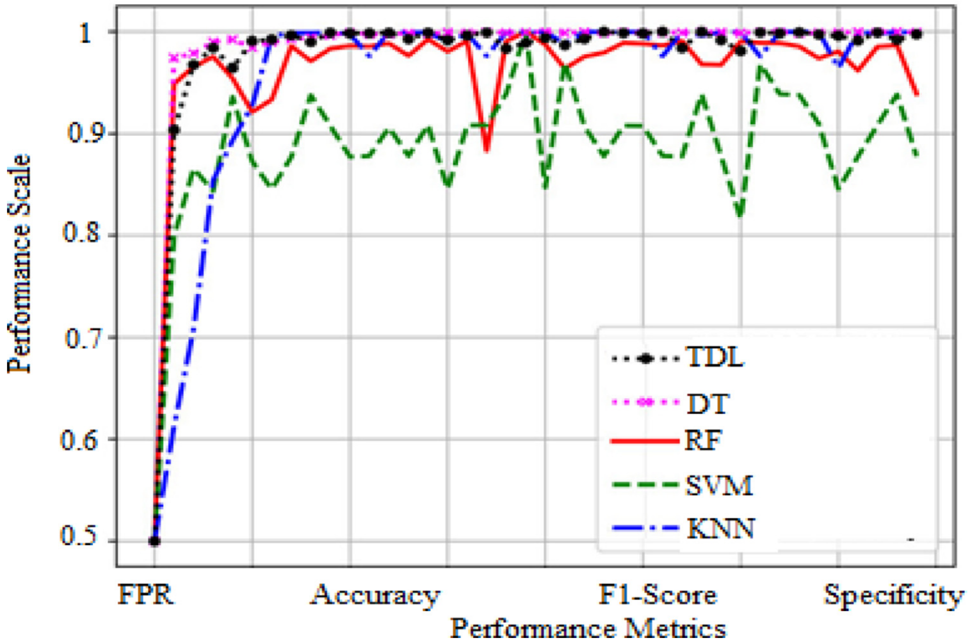

Fig. 10. Comparison of HMM_TDL in DoS→Probe.

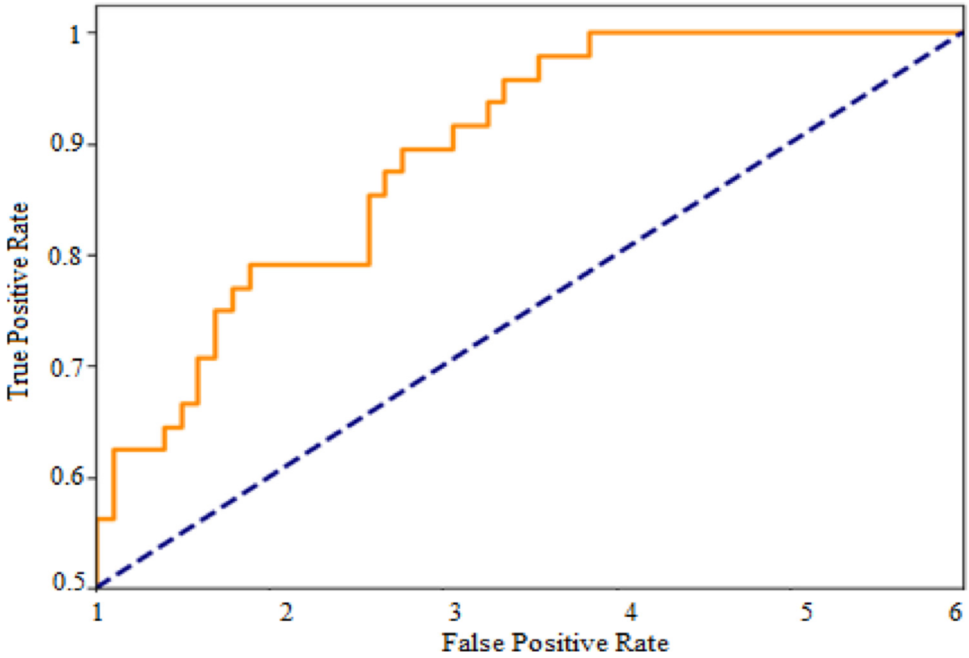

Fig. 11. ROC curve for Fig. 10: ROC curve for NSL_DOS→CIDD.

This will be $A_{attack}(x) \geq A^*{}_S(x)$

$$P(D^*, S) = \int_x A_{attack}(x)(1 - D(x)dx + \int_x A_{data}(x)D(S^*(z))dz \quad (22)$$

$$= \int_x A_{attack}(x)(1 - D(x))dx + \int_x A_S(x)(D(x))dx \quad (23)$$

$$= \int_x D(x) - (p_{G^*}(x) - p_{data}(x))dx + \int_x p_{data}(x)dx \quad (24)$$

$$= \int_x 1A_{attack}(x) > A_{G^*}(x)(A_{G^*}(x) - A_{data}(x))dx + \int_x A_{data}(x)dx \quad (25)$$

$$\geq \int_x 1A_{data}(x) > A_{G^*}(x)(A_{data}(x))dx \quad (26)$$

$$\geq -M \quad (27)$$

5. Experimental analysis & results

This research focused on improving security in cloud from zero-day attack scenario. The proposed scheme transmits hyper alerts with estimation of time window. HMM estimate the attacks in the network and attack is classified based on consideration of threshold values on network.

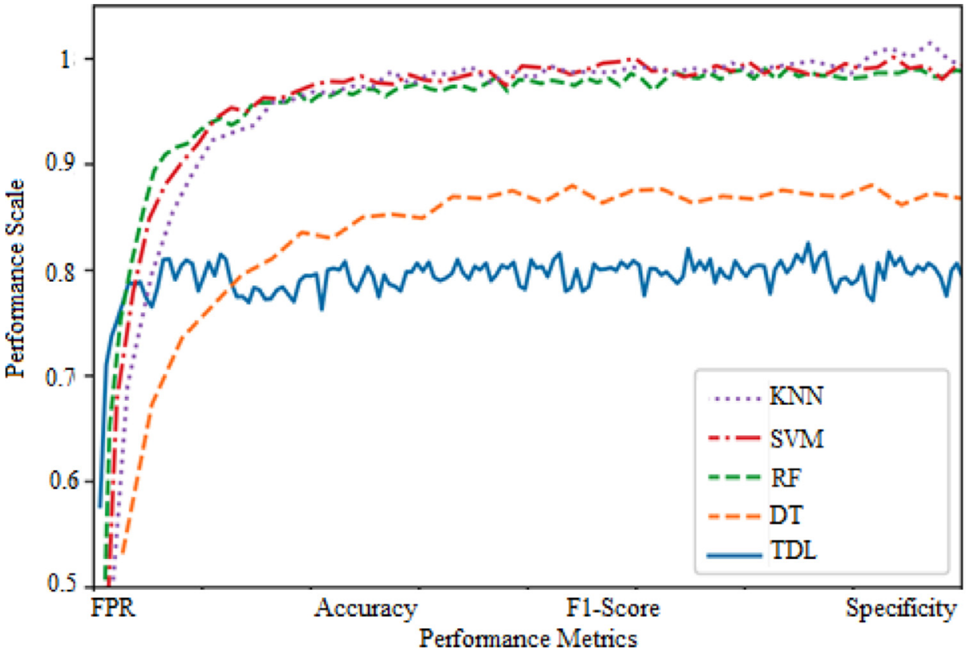

Fig. 12. Comparison of HMM_TDL in NSL_DOS→CIDD.

5.1. NSL-KDD

NSL - KDD is stated as standard dataset with distribution of packet with experimentation of IDS. This dataset consists of features and instances count of 43 and 147,907 respectively. The attack and normal instances are in count of 76,967 and 70,940 respectively. NSL-KDD dataset incorporates attacks such as probe, U2R, DoS and R2L.

5.2. CIDD dataset

CIDD dataset provides identification of cloud dataset with consideration of DoS attack and attack in cloud environment. CIDD dataset is based on time-based data distribution with 5274 instances and 25 features with data dimensionality. The data of CIDD dataset incorporates udp_flood, tcp_syn_flood, pod, dns_flood, land, icmp_flood (smurf) and slowloris.

5.3. Performance metrics

The performance of developed HMM_TDP is examined with consideration of parameters such as accuracy, False Positive Rate (FPR), F1-Score and Specificity.Accuracy is stated as number of websites those are phished to the number of all websites and it is calculated using the Eq. (28):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{28}$$

F1-Score provides the classification performance, which provides average of precision and recall and it is calculated using Eq. (29)

$$F1 - Score = 2 \times \frac{(Precision \times \mathrm{Recall})}{(Precision + \mathrm{Recall})} \tag{29}$$

Specificity describes the attacks those are correctly classified and it is calculated using Eq. (30)

$$Specificity = \frac{TN}{TN + FP} = 1 - FPR \tag{30}$$

Where, FPR - False positive rate

FPR provides the probability of model that reject the original data as attack in the network and it is presented in Eq. (31)

$$FPR = \frac{FP}{FP + TN} \tag{31}$$

For Eqs. (28)–(31) Where, TP - True Positive provides correctly classified attacks

TN - True Negative normal data without attack correctly identified

FP - False Positive defines normal attack classified

FN - False Negative defines normal data classified as attack

5.4. Experimental setup

To evaluate the performance of developed model zero-day data attack detection in cloud is considered.

5.4.1. Test Scenario: zero-day attack detection in cloud

The CIDD attack consists of different attack, this paper considers DoS attack module for detection of zero-day attack in cloud dataset. From analysis it is observed that smurf, pod and land are considered as common attack module for both CIDD and NSL-KDD dataset. Here, NSL-DoS and CIDD incorporates this common instances with consideration of source and target domains. For analysis of complete labeled NSL-DoS module and CIDD no label module in included with in the framework were stated in Table 1.

As presented in Table 1, the purity node $\alpha$is evaluated with generation of cluster group with size of 13. As stated hyper alerts uses k-medoids for R2L modules with DoS and cloud server. The k-value of medoids are considered as 25, with assigned value of $\alpha = 0.92327$. Based on this R2L instances are generated in cloud server with consideration of 25 clusters. The threshold estimated for identification of cluster purity α for $T_1 = 0.92327$ and $T_2 = 0.03566$. The labels for attack are assigned with set of instances for R2L module with cloud value of 0.92327 and normal attack module of R2L includes instances of cloud value of 0.03566.

The DoS module label instances are assigned with soft instances in R2L modules for training DNN module. The DoS→R2L provides accuracy of 0.9275, FPR of 0.0563 and corresponding ROC curve is presented in Fig. 7. The transductive DNN provides improved

Table 2
Performance of k-medoids for NSL-KDD.

Points	Accuracy (%)	FPR (%)	F1-Score (%)	Specificity (%)	Time cost (sec)
500	90	16	85	92	0.04
1000	85	20	92	85	0.44
1500	89	24	90	90	0.52
2000	85	10	91	85	0.36
2500	90	08	86	92	0.25
3000	86	11	90	93	0.32
3500	90	12	85	90	0.66
4000	84	15	85	91	0.43
4500	90	10	90	86	0.51
5000	85	14	90	85	0.40

Table 3
Performance of k-medoids for CIDD.

Points	Accuracy (%)	FPR (%)	F1-Score (%)	Specificity (%)	Time cost (sec)
500	82	11	91	85	0.43
1000	91	13	90	94	0.15
1500	90	06	89	90	0.04
2000	83	13	90	90	0.35
2500	83	19	85	86	0.35
3000	90	08	90	90	0.25
3500	92	07	82	84	0.03
4000	84	19	90	90	0.15
4500	90	18	86	84	0.26
5000	84	07	84	89	0.40

classification for zero-day attack detection compared with existing classifier presented in Fig. 8. The best validation is measured and obtained for transductive DNN with batch size 100 and epochs count of 90. The DoS→Probe attack provides accuracy value of 0.9249, FPR of 0.8416 with ROC curve presented in Figs. 9 and 11 provides ROC curve for NSL_DoS→CIDD. Also, from analysis of Figs. 10 and 12 it is observed that for all parameters accuracy, FPR, F1-Score and sensitivity developed HMM_TDL exhibits improved performance.

In Table 2 for different medoids the performance characteristics of HMM_TDL for two datasets such as NSL-KDD and CIDD. In Table 2 for different medoids performance in NSL-KDD dataset is presented.

Table 2 provides the analysis of performance of HMM_TDL in terms of accuracy and time cost. For fairness in the system medoids are implemented in NSL-KDD and CIDD dataset. Also, in Table 3 presented about the performance characteristics of CIDD is presented.

6. Conclusion

This paper presented HMM_TDL, a deep learning model for detection and prevention of zero-day attacks in cloud environment. In this model, HMM model is incorporated for detection of attack with k-medoids clustering for refining attack datasets. The difference between attack and original data is evaluated based on consideration of NSL-KDD and CIDD features.The constructed model is applied in transductive deep learning network for attack prevention with classifier. Finally, with estimated HMM_TDL performance of attack prevention is estimated with different performance metrics. Also, performance of NSL-KDD and CIDD dataset performance for varying k-medoids are presented. In feature, optimal feature based classification can be implemented for real-time network platform.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] G.D.L.T. Parra, P. Rad, K.K.R. Choo, N. Beebe, Detecting internet of things attacks using distributed deep learning, J. Netw. Comput. Appl. (2020) 102662.

[2] S. Kaur, M. Singh, A proactive framework for automatic detection of zero-day HTTP attacks on educational institutions, Comput. Fraud Secur. 2015 (2) (2015) 10–16.

[3] Ö. KASIM, An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks, Comput. Netw. 180 (2020) 107390.

[4] S. Velliangiri, H.M. Pandey, Fuzzy-taylor-elephant herd optimization inspired deep belief network for DDoS attack detection and comparison with state-of-the-arts algorithms, Future Gener. Comput. Syst. 110 (2020) 80–90.

[5] M.V. de Assis, L.F. Carvalho, J.J. Rodrigues, J. Lloret, M.L. Proença Jr, Near real-time security system applied to SDN environments in IoT networks using convolutional neural network, Comput. Electr. Eng. 86 (2020) 106738.

[6] W. Wei, Q. Ke, J. Nowak, M. Korytkowski, R. Scherer, M. &Woźniak, Accurate and fast URL phishing detector: a convolutional neural network approach, Comput. Netw. 178 (107) (2020) 107275.

[7] N. Sameera, M. Shashi, Deep transductive transfer learning framework for zero-day attack detection, ICT Express (2020) 80–90.

[8] A. Blaise, M. Bouet, V. Conan, S. &Secci, Detection of zero-day attacks: an unsupervised port-based approach, Comput. Netw. 180 (2020) 107391.

[9] H. Tran, E. Campos-Nanez, P. Fomin, J. &Wasek, Cyber resilience recovery model to combat zero-day malware attacks, Comput. Secur. 61 (2016) 19–31.

[10] U.K. Singh, C. Joshi, D. &Kanellopoulos, A framework for zero-day vulnerabilities detection and prioritization, J. Inform. Secur. Appl. 46 (2019) 164–172.

[11] J.Y. Kim, S.J. Bu, S.B. Cho, Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders, Inform. Sci. 460 (2018) 83–102.

[12] D. Hammarberg, in: The Best Defenses Against Zero-Day Exploits for Various–Sized Organizations‖, SANS Institute InfoSec Reading Room, 2014, p. 21.

[13] R. Kaur, M. Singh, Efficient hybrid technique for detecting zero-day polymorphic worms, in: Proceedings of the 2014 *IEEE International Advance Computing Conference(IACC)*, IEEE, 2014, pp. 95–100.

[14] R. Kaur, M. Singh, A survey on zero-day polymorphic worm detection techniques, IEEE Commun. Surv. Tutor. 16 (3) (2014) 1520–1549.

[15] U.K. Singh, C. Joshi, Quantifying security risk by critical network vulnerabilities assessment, Int. J. Comput. Appl. 156 (13) (2016) 26–33.

[16] M.N. Alraja, M.M.J. Farooque, B. &Khashab, The effect of security, privacy, familiarity, and trust on users' attitudes toward the use of the IoT-based healthcare: the mediation role of risk perception, IEEE Access 7 (2019) 111341–111354.

[17] U.K. Singh, C. Joshi, N. Gaud, Information security assessment by quantifying risk level of network vulnerabilities, Int. J. Comput. Appl. 156 (2) (2016) 37–44.

[18] X. Ma, Y. Niu, L. Gu, Y. Wang, Y. Zhao, J. Bailey, F. Lu, Understanding adversarial attacks on deep learning based medical image analysis systems, Pattern Recognit. 275 (2020) 107332.