AMSE JOURNALS-AMSE IIETA publication -2017-Series: Advances D; Vol. 22; N°1; pp 45-61
Submitted March 2017; Revised April 27, 2017, Accepted Nov.15, 2017

Optimisation of Hidden Markov Model for Distributed Denial of Service Attack Prediction Using Variational Bayesian

*A. A. Afolorunso, ** O. Abass

*Department of Computer Science, Faculty of Science, National Open University of Nigeria, Nigeria, 91, Cadastral Zone, Jabi, Abuja, (aafolorunsho@noun.edu.ng)
**Department of Computer Sciences, Faculty of Science, University of Lagos, Nigeria, Akoka-Lagos, (oabass@unilag.edu.ng)

Abstract

Distributed Denial of Service (DDoS), is a coordinated attack majorly carried out on a massive scale against the availability of services/resources of a target system. Several DDoS attack detection, prevention or prediction techniques have been proposed. Some of these techniques have shortcomings such as high false positive rate, high computational time, low prediction precision and so on. This paper presents a novel machine learning technique based on variational Bayesian algorithms to obtain an Hidden Markov Model (HMM) with optimised number of model states and parameters for DDoS attack prediction. This method not only overcomes the slow convergence speed of the HMM approach, but it also avoids the problem of overfitting the model structure by removing excess transition and emission processes. Experiments with the DARPA 2000 intrusion datasets shows this method is able to find the optimal topology in every case and achieves better average precision rate compared to classic HMM.

Keywords: DDoS, Variational Bayesian, Hidden Markov model, network attacks

1. Introduction

With the increase in global interconnectivity via the Internet comes the challenge of security/protection of connected systems. Vulnerability of inter-networked systems has been exerbated by new paradigms such as Internet of Things (IoT) and Internet of everything (IoE). In order to respond to the challenges, researches into techniques for protecting and safeguarding

network systems continue to emerge. One of such research areas is network attacks prediction. The different types of network attacks can be classified into four main categories (Sharma *et al.,* 2015):

i) ***Denial of Service (DoS):*** where an attacker makes network resources too busy to serve legitimate requests. Examples include mail bomb, apache, syn flood

ii) ***Probing (Probe):*** in probing attack, the attacker scans a network device so as to gather information about weaknesses or vulnerabilities that can be exploited to compromise the target system. Examples include nmap, saint, mscan.

iii) ***User to Root (U2R):*** in this category, an authorized user attempt to abuse the vulnerabilities of the system in order to gain privilege of root user he/she is not authorized for. Examples include perl, Fd-format, xterm.

iv) ***Remote to Local (R2L):*** here, a remote user sends packets to a machine over the internet to gain access as a local user to a local machine i.e. the weaknesses of the system is exploited by an external intruder to access the privileges of a local user. Examples include phf, xlock, guest.

The various categories of network attacks aim at undermining the CIA (Confidentiality, Integrity and Availability) properties of the network (Sodiya *et al*., 2004). But specifically, Distributed Denial of Service (DDoS), which is a type of DoS attack where multiple compromised systems, which are often infected with a Trojan, are used to target a single system leading to unavailability of the system services/resources to legitimate users. Commercial web servers, banks, educational institutions and government websites are usually major victims of such attacks. A typical instance and more recent occurrence of DDoS attack is the large-scale DDoS against New Hampshire-based Internet performance company, Dyn, which caused major Internet disruptions on Friday, 21st October, 2016. The attack disrupted internet service across Europe and United States of America (USA). Users were unable to access many major websites such as Twitter, Spotify, Netflix, Amazon, Tumblr, Reddit and other sites.

Several of the techniques such as Time series, Machine Learning (Seng *et al.,* 2010; Zhang *et al.,* 2012; Satpute *et al.,* 2013), Markov Chain (Shin *et al.,* 2013), Hidden Markov Model (HMM) (Cheng *et al.,* 2012; Sendi *et al.,* 2012), Statistical Profiling (Saganowski *et al.,* 2013), Data Mining (Jiao, 2012), Neural Network, and combinations of these methods, which had been applied to detecting and predicting DDoS attacks (Siani *et al.,* 2014; Sharma *et al.,* 2015) have weaknesses which include false positives, low prediction precision, high computational time and false negatives. For these reasons, efforts continue to evolve on how to improve on these

weaknesses. However, among the aforementioned approaches, HMMs, which is a kind of hybrid techniques that incorporate time series and probabilistic techniques, has proven to be very promising for anomaly prediction over several other techniques because of their high accuracy in identifying attacks (Badajena *et al.,* 2012). However, research has shown that the efficiency of HMM-based algorithms is hindered by long training time during model construction (Sendi *et al.,* 2012). This study aims at overcoming this limitation by employing Variational Bayesian inference (VB) in optimizing the HMM learning algorithm.

According to Lee *et al., (*2008), DDoS progresses in stages and can therefore be said to have different phases. The experiments run by the MIT Lincoln Lab (2000) partitioned DDoS attack session into five phases as follows:

1) IPsweep to the DMZ (demilitarized zone) hosts from a remote site.
2) Probe of live IP's to look for the Sadmind daemon running on Solaris hosts.
3) Breaks-in via the Sadmind vulnerability, both successful and unsuccessful on those hosts.
4) Installation of the Trojan mstream DDoS software on three hosts in the DMZ.
5) Launching the DDoS.

At each phase, there are some observable events that occur and these events can be used to predict the state of the system and what could happen in the system in the foreseeable future (Afolorunso *et al*., 2016).

Lee *et al.,* (2008) also identified nine features viz. *Entropy of source IP address, Entropy of source port number, Entropy of destination IP address, Entropy of destination port number, Entropy of packet type, Occurrence rate of Packet type (ICMP, UDP, TCP-SYN)* and *Number of packets* that could be used in analyzing the characteristics of the network during a DDoS attack.

A DDoS attack prediction system is expected to predict the possibility of attack in time for steps to be taken to avert it without adding too much overhead in terms of resources consumption, which might adversely affect the performance of the system.

In this study, the VB algorithm is employed to develop a novel parsimonious and computationally efficient model for predicting DDoS attacks in network systems. The rest of this paper is organised as follows: Section 2 presents previous relevant works to this study; Section 3 gives the proposed research methodology; Section 4 presents the experimental results and discussions of the proposed model; while Section 5 presents the conclusion of the study and future work.

2. Related Research

HMM, which is an excellent tool when it comes to modelling large number of temporal sequences, has been widely used for pattern matching in speech recognition (Rabiner, 1989), image identification (Bunke, 2001), diagnotics (Nkemnole *et al.,* 2013) and network attacks (Cuppens, 2001). Since its introduction into anomaly detection by Warrender *et al.,* (1999), HMM has been deployed either singly or in combination with other techniques in network anomaly detection and prediction. Some of such works are discussed below:

Haslum *et al.,* (2008) used an HMM model that models only integrity and confidentiality, and makes no attempt to model availability. They believe that availability is best modelled separately. Preliminary experimental results from this system indicates that the proposed framework is efficient for real-time distributed intrusion monitoring and prevention.

Khosronejad *et al.,* (2013) worked on a hybrid approach for modelling IDS. C5.0 and HMM were combined as a hierarchical hybrid intelligent system model. Experimental results with KDD Cup '99 benchmark Intrusion data showed that the proposed hybrid system provide more accurate intrusion detection compared to ordinary HMM approach.

Rao *et al.,* (2012) applied HMM to monitor Application Layer DDoS attacks on web servers. They applied forward-backward algorithm to train HMM model thereby increasing the response time of the application. In their work, which is a counter-solution to diverse Application layer DDoS attacks, the web site design was customised so as to minimise Application layer DDoS attacks.

Divya *et al.,* (2015) proposed a hybrid framework, which combined two machine-learning techniques, hidden Markov model (HMM) and genetic algorithm (GA) for predicting future intrusion attacks in network systems. As indicated, the framework was made up of two main components: the first component uses GA to formulate efficient intrusion detection rules which leads to a precise attacks detection, the second component employs HMM in predicting the next attack plan of the attacker. The combination of these two gives a good intrusion prediction capability with reduced false positive rate.

Thanthrige *et al.,* (2016) proposed an HMM-based alert prediction framework. Alert clustering was employed to group selected alert attributes together. A given sequence of alerts is

converted to a sequence of alert clusters and then HMM is used to predict future alert clusters based on the input. Experimental results show good performance. However, they identified the following challenges to be addressed in the proposed alert prediction framework: (1) increasing the prediction accuracy with the increase of cluster size and predicting intrusion types that are not present in the training data set, and (2) identifying false alerts and misleading intrusion actions generated by the attacker in order to mislead intrusion detection systems.

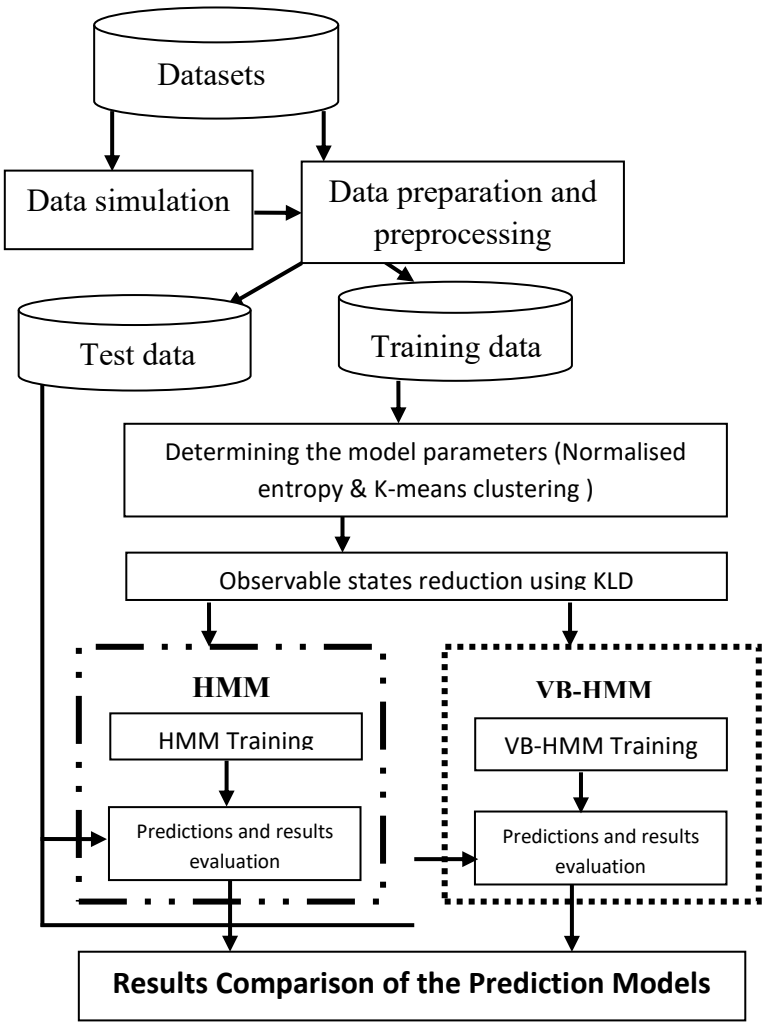
3. Research Methodology

As earlier mentioned, this study aims at developing a novel parsimonious DDoS attack prediction model with high prediction precision and improved computational time. This is achieved by combining VB with HMM algorithms to predict DDoS attacks. This section briefly presents the research model of this study and the proposed procedure for prediction, which, to the best of our knowledge, no other work has used the combination of all the methods here in the same context.

The proposed model is based on HMM algorithms. The entropy-based features to be used as the observable states of the HMM are many so Kullback-Liebler Divergence (KLD) is used to select the minimum number of the features that could represent the whole to achieve improved performance without loss of information. Due to the shortcomings of HMMs especially the traditional learning algorithm of the HMM, VB is deployed in training the model.

The experimental procedure consists of four major steps. In the first step, the network states are defined by means of clustering the network traffic based on the entropy values of the network traffic features and the observables states of the model reduced using adapted relative entropy algorithm. In the second step, the parameters of the model, that is, the initial probability distribution, the state transition probability and the emission transition probability of the HMM is built based on the definitions got from the first step. In the third step, the traditional HMM algorithm is used to train the model formulated in step 2 using the DARPA 2000 intrusion dataset after which two sets of test data (DARPA 1999 (no attack) dataset and simulated real time dataset) are used to test the model and make predictions. In the fourth step, VB algorithm was used to train the HMM model of step 2. The VB-HMM was also tested and used for prediction. Finally, the results and computational efficiency of the two models were compared.

The architecture of the proposed model is as in Figure 1 below:



3.1 Model construction

3.1.1 Estimating the values of network features

For an information source with *n* independent symbols each with probability of choice *P*(*i*), the entropy, *H,* is defined as below (Shannon, 1948):

$$H = -\sum_{i=1}^{n} P(i) \log_2 P(i) \quad (1)$$

Therefore, entropy can be computed on a sample of consecutive packets. Comparing the value for entropy of some sample of packet header fields to that of other samples of packet header fields provides a mechanism for detecting and predicting changes in the randomness.

In order to construct the HMM, using the concept of entropy (Berezinski *et al.,* 2015), the desirable features of the temporal network data as listed in section 1 were estimated using the normalised entropy algorithm in Afolorunso *et al.,* (2016) at regular interval. To compute the

entropies, the probabilities of each quantity in the training data was computed and plugged into equation (1)

Then, *K*-means clustering algorithm (MacQueen, 1967) is applied to classify the network behaviour into states. The state of each observation is identified by the cluster it belongs to. To achieve model parsimony, the adapted KLD algorithm in Afolorunso *et al.,* (2016) was then applied in reducing the observable states of the model.

3.1.2 Determining model parameters

The values of the estimated features in Section 3.1.1 above were then used in determining the HMM parameter $\lambda = (A, B, \pi)$.

HMM, a bivariate discrete-time process (Ahani *et al*., 2011), is a type of finite state machine with a set of hidden states, Q, an output alphabet (observations), O, transition probabilities, A, output (emission) probabilities, B, and initial state probabilities, π. The current model state is usually hidden and not observable but each state produces an output with a specific probability (B). Usually the states, Q, and outputs, O, are understood, hence an HMM is customarily a triple, (A, B, π). Traditionally, an HMM is characterized by the following:

i) Hidden states $Q = \{ q_i \}, i = 1, \ldots, N$.
ii) Transition probabilities $A = \{a_{ij} = P(q_j \text{ at } t+1 \mid q_i \text{ at } t)\}$, where $t = 1, \ldots, T$ is time, and q_i in Q. That is, A is the probability that the next state is q_j given that the current state is q_i.
iii) Observations (symbols) $O = \{o_k\}, k = 1, \ldots, M$.
iv) Emission probabilities $B = \{b_{ik} = b_i(o_k) = P(o_k \mid q_i)\}$, where o_k in O.
v) Initial state probabilities $\pi = \{p_i = P(q_i \text{ at } t = 1)\}$.

As in Afolorunso (2016), the five phases of DDoS resulting from Section 3.1.1 and an additional state N that represents the normal state when no malicious activity is going on in the system, forms the set of hidden states from which the parameters A and π are derived. These states are represented by the symbols, I, P, R, T, D and N respectively. Hence, $Q_i = (q_1 = N; q_2 = I; q_3 = P; q_4 = R; q_5 = T; q_6 = D)$.

3.1.3 Model training and testing

In training and testing the model, first, the model so formulated was trained using the Baum-Welch algorithm (Ibe, 2013) until convergence. Then the two sets of test data as aforementioned were used to test the model and make predictions. The prediction module is implemented using Viterbi algorithm (Ibe, 2013). Secondly, in pursuance of good performance in overall computational time and prediction precision, the model derived in Section 3.1.2 was again trained using VB algorithms (Beal, 2003).

3.1.3.1 Variational Bayesian inference (VB)

In machine learning, VB is mostly used to infer the conditional distribution (also known as the posterior distribution) over the latent variables given the observations (and parameters). VB for HMMs seeks to minimise the divergence between the true posterior and an approximation in which the parameters and hidden variables are assumed independent, which assumption allows for a very efficient iterative solution (Beal, 2003; Ahani *et al*., 2011). The paramount idea is to pick a family of distributions over the latent variable with its own variational parameters ($q(Y_{1:m}|V)$ and then find the setting of the parameters that makes q close to the posterior of interest. q is used with the fitted parameters as a proxy for the posterior. The closeness of the two distributions is measured with KLD as in Beal (2003).

The concept of KLD embedded in the VB used in optimizing the HMM learning algorithm is as given below:

$$KL(q||p) = E_q\left[\log \frac{q(Y)}{p(Y|x)}\right] \quad (2)$$

Where $x = x_1, x_2, ..., x_n$ are the observations and $y = y_1, y_2, ..., y_m$ are the hidden variables

It is not easy to minimize the KLD as a function of variational distribution. But this can be achieved by maximizing the evidence lower bound (ELBO) of the function. ELBO is obtained by applying Jensen inequality (f (E[X] ≥ E[f(X)] when f is concave) on the log probability of the observations, (Beal, 2003)

$$\log p(x) = \log \int_y p(x,y) \quad (3)$$

$$= \int_y p(x,y) \frac{q(y)}{q(y)} \quad (4)$$

$$= \log \left(E_q \left[\frac{p(x,Y)}{q(y)}\right]\right) \quad (5)$$

$$\geq E_q [\log p(x,Y)] - E_q [\log q(Y)] \quad (6)$$

Equation (6) is the ELBO and it is the same bound used in deriving the EM algorithm (Ibe, 2013). A family of variational distributions is chosen to make the expectations computable. In this study, Dirichlet distribution is chosen because it is the conjugate to the complete-data likelihood terms of the HMM (Beal, 2003). It can be shown that the difference between the ELBO and KLD is the log normalizer, which is what the ELBO bounds (Beal, 2003). Hence, minimizing KLD is the same as maximizing the ELBO.

Finally, the results obtained from each of the two models above were compared using standard metrics for intrusion prediction such as false positive (FP) rate, false negative (FN) rate, true positive (TP) rate, true negative (TN) rate, precision rate, confusion matrix.

4.Results and Discussion

4.1 Implementation platform

In this section, the model architecture and the design methodology steps enumerated in Section 3 were implemented in suitable software platform and the experimental results evaluated using appropriate metrics. The training data and one of the test data are available at http://www.ll.mit.edu/IST/ideval/data/2000/2000_data_index.html, https://www.ll.mit.edu/ideval/data/.

4.1.1 The prediction models

Implementing the step 1 of the design methodology, the desirable network traffic features is calculated at regular interval. First, at regular interval of one second and then five seconds. It was discovered that the two intervals produced identical results. So, five seconds interval was stuck to. Six hidden states corresponding to the number of clusters were arrived at. The states correspond to the phases of DDoS attack as listed in Section 1 (denoted by *I, P, R, T* and *D* respectively) and an additional normal state (denoted by *N*) when there are no traces of malicious

activity or any attempt to break into the system. So, $Q_i = (q_1 = N; q_2 = I; q_3 = P; q_4 = R; q_5 = T; q_6 = D)$.

The finite set of M possible symbols ($O = \{o_1, o_2, o_3, \ldots, o_M\}$) in this study, are the three (entropy of source IP (*SI*), entropy of destination IP (*DI*) and Occurrence rate of Protocol (*PO*)) that the KLD results shows are adequate in representing the system out of the nine network features listed in Section 1.

The State Transition Probability (A_{ij}), the Emission Transition Probability ($B_j(k)$) and the Initial State Probability (π_i) were obtained from the data and approximated to five decimal places. At system start-up, $\pi = (0.97183, 0.02452, 0.00118, 0.00098, 0.00108, 0.00041)$, which implies that the system, has the probability of 0.97183 of being in state *N*; 0.02452 of being in state *I*; 0.00118 of being in *P*; 0.00098 of being in *R*; 0.00108 of being in *T*, and 0.00041 of being in state *D*. Next, the state transition probability (*A*), which is a 6 X 6 matrix and the emission probability matrix (*B*), also a 6 X 3 matrix was estimated from the temporal network as depicted below:

$$A = \begin{matrix} 0.95880 & 0.00002 & 0.03819 & 0.00002 & 0.00216 & 0.00082 \\ 0.03026 & 0.00001 & 0.96659 & 0.00311 & 0.00001 & 0.00001 \\ 0.00001 & 0.95527 & 0.04468 & 0.00001 & 0.00001 & 0.00001 \\ 0.00166 & 0.00404 & 0.00041 & 0.84611 & 0.14777 & 0.00002 \\ 0.00011 & 0.00011 & 0.00011 & 0.96471 & 0.00011 & 0.03484 \\ 0.00046 & 0.00001 & 0.00001 & 0.00151 & 0.00060 & 0.99741 \end{matrix}$$

$$B = \begin{matrix} 0.89179 & 0.10011 & 0.00810 \\ 0.58648 & 0.31794 & 0.09559 \\ 0.55745 & 0.33734 & 0.10521 \\ 0.00960 & 0.98828 & 0.00212 \\ 0.00011 & 0.99977 & 0.00011 \\ 0.50798 & 0.30650 & 0.18551 \end{matrix}$$

$$\pi = \begin{matrix} 0.97183 & 0.02452 & 0.00118 & 0.00098 & 0.00108 & 0.00041 \end{matrix}$$

The HMM, $\lambda = (A, B, \pi)$, was trained as earlier stated in Section 3.1, the model converged after about 60 iterations in 59.52 seconds as shown in Table 1 below.

Table 1. Performance benchmark of the models.

MODELS	Computational time (CT) in seconds	True Positive Rate (TPR)	False Negative Rate (FNR)	False Positive Rate (FPR)	True Negative Rate (TNR)
VB-HMM	17.79	0.92	0.08	0.11	0.89
HMM	59.53	0.84	0.16	0.21	0.79

Two sets of test data, as earlier mentioned, were run through the model for prediction using the Baum-Welch algorithm (Ibe, 2013). It was discovered that the model has FNR of 16% and FPR of 21%.

The HMM was then trained with VB and used for prediction using the same sets of data. First the Maximum Likelihood (ML) algorithm was run to convergence, and then the VB algorithm was run from that point in parameter space to convergence. This was achieved by initialising each parameter's variational posterior distribution to be Dirichlet with the ML parameter as the mean and by arbitrarily setting strength to 6. For the VB algorithm, the prior over each parameter was a symmetric Dirichlet distribution of strength 4.

Note that as depicted in Table 1, where it takes HMM about 60 iterations to converge to a local optimum, it takes only about 20 iterations for the VB optimisation to converge to global optimum. This is expected since the VB is initialised to the ML parameters, and so has less work to do.

As shown in Table 1, the computation time was within 18 seconds; the false positive rate was considerably reduce to 8% and the false negative rate to 11%. Compared to the traditional HMM constructed in this study, VB-HMM shows better performance on all metrics used. The combination of the TPR, FPR, TNR and FNR form the confusion matrix for each of the models. For example the confusion matrices for VB-HMM, HMM are given as, $\begin{pmatrix} 0.92 & 0.08 \\ 0.11 & 0.89 \end{pmatrix}$, $\begin{pmatrix} 0.84 & 0.16 \\ 0.21 & 0.79 \end{pmatrix}$, respectively.

Table 1 depicts the performance benchmark of the models while Figures 2, 3 and 4 are the pictorial representation of their comparison based on confusion matrices, computational time and prediction accuracy, respectively.

As shown in Table 1 and Figures 2, 3 and 4 VB-HMM was better than HMM in terms of computational time, confusion matrix and prediction accuracy, respectively.

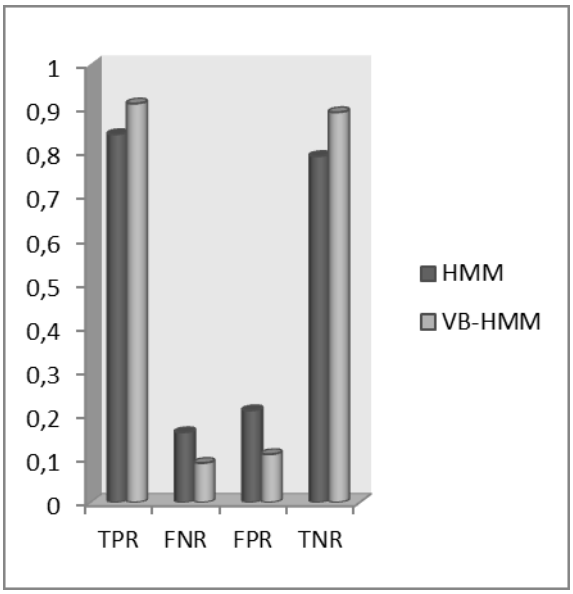


Figure 2. Graphical representation of the confusion matrices of the Models

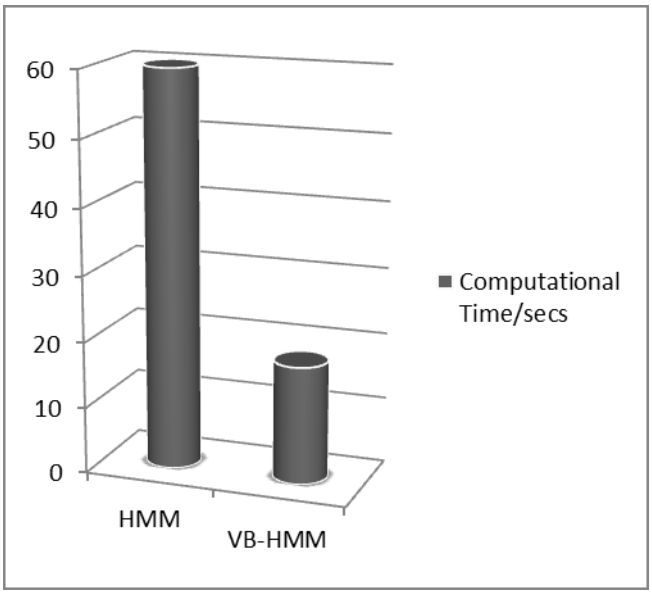


Figure 3. Graphical representation of the computational time of the Models

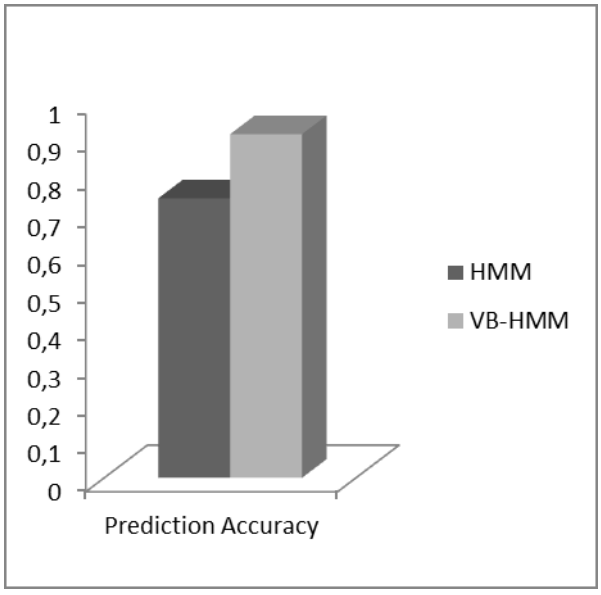


Figure 4. Graphical representation of the prediction accuracy of the models

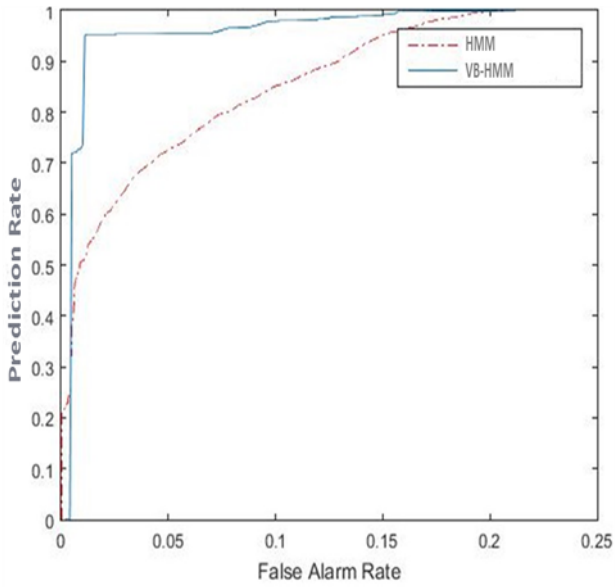


Figure 5. ROC curve of the performance of the HMM and VB-HMM

Figure 5, the Receiver Operator Characteristics (ROC) curve of the test data is a graphical metric that illustrates the performance of a classifier which in our case is an HMM model that classifies Packet sequence as Threat or Normal traffic. The plot shows the rate of prediction as against false alarm rate. The curve with the continual variation depicts the plot of the HMM

model and it shows an approximate variation between false and true classification of sequence packet data. The other curve representing the VB-HMM model shows a less accurate detection rate initially until a threshold (around 0.01) is overcome where the performance of the model becomes excellent.

In the implementation of a DDoS attack prediction system, this threshold value that translates to an improved performance should be taken into account when developing such systems.

This information depicts the trade-off between the models but it can be concluded that the VB-HMM model is more robust in terms of the prediction accuracy as depicted by its confusion matrix.

The VB-HMM model performed better in terms of classification of packet sequence as either normal or attack-prone. Computationally, VB-HMM is more efficient as it reduces the propensity for over-fitting data due to model complexity which is not addressed by HMM.

Overall for real time application, the VB-HMM is recommended for use since it can compute and predict traffic status in a relatively shorter time. It also ensures the efficiency of prediction over all other models.

5. Conclusion

This study proposes a robust and efficient architecture for DDoS attack prediction. The proposed model was formulated, implemented, and tested with different types of datasets. Experimental results on the DARPA datasets have shown that the proposed model converges faster, which translates into computational efficiency, and shows good performance in predicting attacks compared to traditional HMM. In future, it is our plan to extend this work by using the proposed model to predict other types of network intrusions.

References:

[1] A. A. Afolorunso, A. P. Adewole, O. Abass, H. O. D. Longe, "Kullback-Liebler divergence for reducing the observable states space of hidden Markov model for predicting distributed denial of service attack". 11th Unilag Conference and Fair, 2016, Lagos, Nigeria, Proc. pp. 184-193, 2016.

[2] B. Agarwal and N. Mittal, "Hybrid approach for detection of anomaly network traffic using data mining techniques", 2nd International Conference on Communication, Computing & Security [ICCCS-2012], Procedia Technology, Vol. 6, pp. 996-1003, 2012

[3] E. B. Ahani, O. Abass and R. A. Kasumu, "Sequential Monte Carlo and expectation maximization algorithm for estimating parameters of a hidden Markov model", AMSE Journals, Series Advances D, Vol. 16, No. 1, pp 1-21, 2011

[4] J. C. Badajena and C. Rout, "Incorporating hidden Markov model into anomaly detection technique for network intrusion detection", International Journal of Computer Applications, vol. 53, No. 11, pp. 42-47, 2012

[5] M. Beal, "Variational algorithms for approximate bayesian inference. Ph.D. thesis, The Gatsby Computational Neuroscience Unit, University College, London, 2003

[6] P. Berezinski, B. Jasiul and M. Szpyrka, An entropy-based network anomaly detection method, Entropy 2015, vol. 17, 2367-2408, 2015

[7] H. Bunke and T. Caelli, Hidden Markov models: Applications in computer vision, 2001; World Scientific Pub Co Inc, Exeter. United Kingdom,

[8] X. Cheng and Y. Ni, "the research on dynamic risk assessment based on hidden Markov models", 2012 International Conference on Computer Science & Service System (CSSS), Nanjing, China, August 2012, Proc. pp. 1106-1109, 2012.

[9] R. Clausius and T. Hirst, The Mechanical Theory of Heat: With its applications to the steam-engine and to the physical properties of bodies, 1867; J. van Voorst: London, UK, 1867.

[10] F. Cuppens, “Managing alerts in a multi-intrusion detection environment", 17th Annual Computer Security Applications Conference, ACSAC '01, Washington, DC, USA, December, 2001, Proc. pp 22-31, 2001

[11] T. Divya and K. Muniasamy, "Real-time intrusion prediction using hidden Markov model with genetic algorithm",. In: Suresh L., Dash S., Panigrahi B. (Eds.) Artificial Intelligence and Evolutionary Algorithms in Engineering Systems. Advances in Intelligent Systems and Computing, Vol. 324. Springer, New Delhi, India, 2015.

[12] P. Dorogovs, A. Borisov and A. Romanovs, "Building an Intrusion Detection System for IT Security Based on Data Mining Techniques", Scientific Journal of Riga Technical University, Computer Science, Information Technology and Management Science, vol. 45, No. 1, pp. 43-48, 2011

[13] J. J. Flores, A. Antolino and J. M. Garcia, "Evolving hidden Markov models for network anomaly detection", 10.1109/ICNS.2010.44. Sixth International Conference on Networking and Services (ICNS) 2010, Cancun, Mexico, Mexico, March 2010, Proc. pp. 1-9, 2010.

[14] K. Haslum, M. E. G. Moe and S. J. Knapskog, "Realtime intrusion prevention and security analysis of networks using HMMs", 33rd IEEE Conference on Local Computer Networks, LCN 2008, Montreal, Que, Canada, October 2008, Proc. pp. 927-934, 2008

[15] O. C. Ibe, Markov Processes for Stochastic Modelling, 2013, Elsevier Academic Press, California, USA.

[16] F. Jemili, M. Zaghdoud and M. B. Ahmed, "Hybrid intrusion detection and prediction multiagent system", HIDPAS, (IJCSIS) International Journal of Computer Science and Information Security, vol. 5, No.1, pp. 62-71, 2009.

[17] M. Khosronejad, E. Sharififar, H. A. Torshizi and M. Jalali, "Developing a hybrid method of hidden Markov models and C5.0 as a intrusion detection system", International Journal of Database Theory and Application, vol. 6, No. 5, pp. 165-174, 2013.

[18] K. Lee, J. Kim, K. H. Kwon, Y. Han and S. Kim, "DDoS attack detection method using cluster analysis", Expert Systems with Applications, vol. 34, No. 3, pp. 1659–1665, 2008

[19] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations", Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability", Berkeley, University of California Press, vol. 1, 281-297, 1967

[20] MIT Lincoln Lab (2000). DARPA intrusion detection scenario specific datasets. http://www.ll.mit.edu/IST/ideval/data/2000/2000_data_index.html, access date January 2015.

[21] MIT Lincoln Lab (1999). DARPA intrusion detection scenario specific datasets. Available at http://www.ll.mit.edu/IST/ideval/data/1999/1999_data_index.html, access date January 2015.

[22] E. B. Nkemnole, O. Abass and R. K. Kasumu, "Parameter estimation of a class of hidden Markov model with diagnostics", Journal of Modern Applied Statistical Methods, vol. 12, No. 1, pp. 181 - 197, 2013.

[23] P. R. M. Rao, K.V. Reddy and S. V. Hemanth, "Minimizing application layer DDoS attacks using website customization", International Journal of Computer Science and Technology, vol. 3, No. 4, pp. 838-841, 2012.

[24] L. Saganowski, M. Goncerzewicz and T. Andrysiak, "Anomaly detection preprocessor for SNORT IDS system", In: Choraś R. (Eds) Image Processing and Communications Challenges 4. Advances in Intelligent Systems and Computing, vol 184. 2013, Springer, Berlin, Heidelberg, pp. 225-232, 2013.

[25] K. Satpute, S. Agrawal, J. Agrawal and S. Sharma, "A survey on anomaly detection in network intrusion detection system using particle swarm optimization based machine learning techniques In: Satapathy S., Udgata S., Biswal B. (Eds.) Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA). Advances in Intelligent Systems and Computing, 2013, Springer, Berlin, Heidelberg, vol. 199, pp. 441-452, 2013.

[26] S. Sendi, M. Dagenais, M. Jabbarifar and M. Couture, "Real time intrusion prediction based on optimized alerts with hidden Markov model", Journal of Networks, vol. 7, No. 2, pp. 311-321, 2012.

[27] J. L. Seng and T. C. Chen, "An analytic approach to select data mining for business decision", Expert Systems with Applications, vol. 37, No. 12, pp. 8042-8057, 2010.

[28] C. E. Shannon, "A mathematical theory of communication", The Bell System Technical Journal, vol. 27, pp. 379–423, 1948.

[29] S. Sharma and R. K. Gupta, "Intrusion detection system: A review", International Journal of Security and Its Applications, vol. 9, No. 5, pp. 69-76, 2015

[30] S. Shin, S. Lee, H. Kim and S. Kim, "Advanced probabilistic approach for network intrusion forecasting and detection", Expert Systems with Applications, vol. 40, No. 1, pp. 315-322, 2013.

[31] Sodiya, A. S., Longe, H. O. D. and Akinwale, A. T. (2004), "A new two-tiered strategy to intrusion detection", Information Management and Computer security, vol. 12, No. 1, pp. 27-44, 2004.

[32] U. S. K. P. M. Thanthrige, J. Samarabandu and X. Wang, Intrusion alert prediction using a hidden Markov model, https://arxiv.org/pdf/1610.07276, access date January 2017.

[33] C.Warrender, S. Forrest and B. Pearlmutter, Detection of Intrusion Using System Calls: Alternative Data Models[C], IEEE Symposium on Security and Privacy, 1999, www.researchgate.net/publication/2448365_Detecting_Intrusions_Using_System_Calls_Alternative_Data_Models, access date December 2016

[34] X. Zhang, L. Jia, H. Shi, Z. Tang and X. Wang, "The Application of Machine Learning Methods to Intrusion Detection", Spring Congress on Engineering and Technology (S-CET), 2012, Xian, China, November 2012, Proc. pp. 1-4, 2012.