Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 4 (2012) 506 – 514

Procedia Technology

C3IT - 2012

Intrusion Detection System using Bayesian Network and Hidden Markov Model

Nagaraju Devarakonda[a], Srinivasulu Pamidi[b], Valli Kumari V[c], Govardhan A[d]

[a]*Department of cse, Acharya Nagarjuna University, Guntur- 522510, India*
[b]*Department of cse, V R Siddhartha Engineering College, Vijayawada-520007, India*
[c]*Department of CS & SE, Andhra University, Visakhapatnam-, India*
[d]*Department of CSE, J N T University, Hyderabad-, India*

Abstract

Across the globe, billions of dollars are spending every year to provide security to the network systems to prevent the intrusions. Some consider the disruption of the vital systems as a serious threat which disables the work of hospitals, banks, military and various internet services across the world. To avert this impending threat, there are many possible solutions: one of these solutions is intrusion detection systems (IDS). The paper proposes to discuss the IDS model in its elaboration using Bayesian Network and the Hidden Markov Model (HMM) approach with KDDCUP dataset. The IDS framework has been designed with various levels of processing such as model learning with training data and constructing the Bayesian Network and this structure has been used as HMM state transition diagram. The preprocessed KDDCUP dataset has been used to train and test the model. The IDS model has been trained and tested for normal and attack type connection records separately. The results evince that the performance of the model is of high order for classification of normal and intrusions attacks.

Keywords: IDS, State Transition Probabilities, Observations, Training, and Evaluation.

1. Introduction

An intrusion detection system (IDS) [1][5][15][16] is used to monitor network traffic, check for suspicious activities and notifies the network administrator or the system. In some instances, the IDS might also react to malicious or anomalous traffic and will take action such as barring the user or perhaps the IP address source from accessing the system.

IDS [8] are available in many different types and will approach the mission of uncovering shady traffic in various ways. Several types of IDSs exist such as Network Intrusion Detection System (NIDS), Host Based Intrusion System (HIDS), and Hybrid Based Intrusion Detection System. These systems use either statistical anomaly-based IDS or Signature based IDS for intrusion detection. It is impossible for

2212-0173 © 2012 Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
doi:10.1016/j.protcy.2012.05.081

IDS to be perfect because network traffic is so high. The objective of the IDS is to minimize false positive rate and maximize true positive rate.

This paper aims at investigating the capabilities of HMM and Bayesian Network for building IDS. The Bayesian Network will be constructed with training data. Using the Bayesian network conditional probabilities can be estimated and the dependencies among the variables can be found. Based on network information the state transition probabilities and emission probability matrices can be initialized and these parameters will act as HMM parameters for model building. The structure of paper is as follows: Section 2 gives brief introduction to the concepts of Bayesian Network and HMM, section 3 covers kddcup99 dataset description, section 4 describes the experimental setup for building IDS using Bayesian Network and HMM Model, and last section 5 covers the concluding remarks.

HMM-based classifiers are capable of detecting intrusion attempts on network systems [9]. IDS using HMM based predictive model capable of discriminating between normal and abnormal behaviors of network traffic [5]. This paper investigates the problem of intrusion detection while reducing the number of false positives.

2. Bayesian Network and Hidden Markov Model (HMM)

Hidden Markov Model (HMM) approach can be used to various kinds of applications, such as speech recognition, Speech synthesis, Gene prediction, Crypt analysis and many more. The incorporation of HMM's for intrusion detection system is still in its infancy. Hidden Markov models are generative models in which the joint distribution of observations and hidden states are equivalent prior to the distribution of hidden states (the transition probabilities). This is based on conditional distribution of observations and given states (the emission probabilities) [4].

2.1 Bayesian Network

A Bayes net also called a belief network is an augmented directed acyclic graph, represented by G(V, E) where V is a set of vertices and E is a set of directed edges joining vertices. In Bayes net no loops of any length are allowed. Each vertex in V contains the name of a random variable and probability distribution table indicating how the probability of this variable's values depends on all possible combinations of parental values.

The following procedure has been used in building the Bayes Net:

1. Choose a set of relevant variables from training dataset. These variables are state variables in HMM
2. Choose an ordering for them.
3. Assume they're called X_1, X_2, ... , X_m (where X_1 is the first in the ordering, X_2 is the second, etc)
4. For i = 1 to m:
    a) Add the X_i node to the network
    b) Set Parents(X_i) to be a minimal subset of {$X_1 \ldots X_{i-1}$ } such that we have conditional independence of X_i and all other members of {$X_1 \ldots X_{i-1}$ } given Parents(X_i)
    c) Define the probability table of P(X_i = k | Assignments of Parents(X_i)).

2.2 HMM Architecture

The figure 1 shows the general architecture of an instantiated HMM. Each oval shape represents a random state variable that can adopt any number of values. The random variable $Z(t)$ is the hidden state at time t, where $Z(t) \in \{ Z_1, Z_2, Z_3 \ldots Z_M \}$. The random variable $X(t)$ is the observation at time t with

$X(t) \in \{ X_1, X_2, X_3, \ldots X_N \}$. The arrows in the diagram often called a trellis diagram denote conditional dependencies.

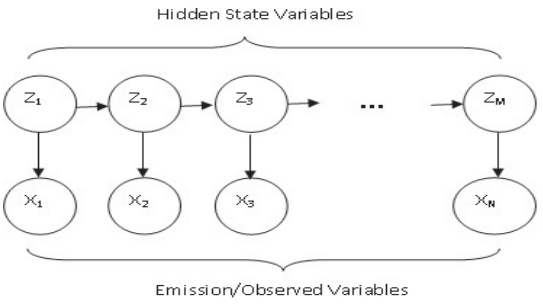

Figure1. The HMM Model Architecture

From the figure 1, it is clear that the conditional probability distribution of the hidden variable Z(*t*) at time *t*, given the values of the hidden variable Z at all times, depends *only* on the value of the hidden variable Z($t-1$): the values at time $t-2$ and before have no influence. This is called the Markov property. Similarly, the value of the observed variable X(*t*) only depends on the value of the hidden variable Z(*t*).

In the standard type of hidden Markov model considered here, the state space of the hidden variables is discrete, while the observations themselves can either be discrete (typically generated from a categorical distribution) or continuous (typically from a Gaussian distribution). The parameters of a hidden Markov model are of two types, transition probabilities and emission probabilities (also known as output probabilities). The transition probabilities control the way the hidden state at time *t* is chosen given the hidden state at time $t-1$.

2.3 Hidden Markov Model

The Hidden Markov Model (HMM) [14] is a finite set of *states*, each of which is associated with a probability distribution. Transitions among these states are governed by a set of probabilities called *state transition probabilities.* In a particular state an outcome or *observation* can be generated, according to the associated probability distribution. It is only the outcome, not the state visible to an external observer and therefore states are "hidden" to the outside; hence the name Hidden Markov Model.

In order to define an HMM [11] completely, following five elements are needed.

(i) The number of states of the model, *N*.

(ii) The number of observation symbols in the alphabet, *M*. If the observations are continuous then *M* is infinite.

(iii) A set of state transition probabilities. $A = \{a_{ij}\}$

$$a_{ij} = p\{q_{t+1} = j \mid q_t = i\}, \qquad 1 \le i, j \le N, \tag{1}$$

Where q_t denotes the current state.

Transition probabilities should satisfy the normal stochastic constraints,

$$a_{ij} \ge 0, \qquad 1 \le i, j \le N \text{ and } \sum_{j=1}^{N} a_{ij} = 1, \qquad 1 \le i \le N$$

(iv) A probability distribution in each of the states, $B = \{b_j(k)\}$ (2)

$$b_j(k) = p\{o_t = v_k \mid q_t = j\}, \quad 1 \le j \le N, \quad 1 \le k \le M$$

where v_k denotes the k^{th} observation symbol in the alphabet, and o_t the current parameter vector. Following stochastic constraints must be satisfied

$$b_j(k) \geq 0, \quad 1 \leq j \leq N, \quad 1 \leq k \leq M \text{ and } \sum_{k=1}^{M} b_j(k) = 1, \quad 1 \leq j \leq N$$

(v) The initial state distribution, $\pi = \{\pi_i\}$ .where, $\pi_i = p\{q_1 = i\}, \quad 1 \leq i \leq N$

Therefore we can use the compact notation $\lambda = (A, B, \pi)$ to denote an HMM with discrete probability distributions.

3. KDD CUP 99 Dataset Description

Since 1999, KDD'99 [3] has been the most wildly used data set for the evaluation of intrusion detection systems. This data set is prepared by Stolfo et al. [7] and is built based on the data captured in DARPA'98 IDS evaluation program [6]. DARPA'98 is about 4 gigabytes of compressed raw (binary) tcpdump data of 7 weeks of network traffic, which can be processed into about 5 million connection records, each with about 100 bytes. The two weeks of test data have around 2 million connection records. KDD training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features and is labeled as either normal or an attack, with exactly one specific attack type. The simulated attacks fall in one of the following four categories [13][15][16] such as denial of service(DoS), User to Root(U2R), Remote to Local(R2L), and Probe. The data has been preprocessed before using for training and testing of the IDS model. The preprocessing of the dataset has been explained in section 4.2.

4. Framework for IDS using Bayes Network and HMM

The framework for building IDS using Bayes network and HMM is shown in figure 2. The framework consists of different levels of processing: dataset reading, preprocess the data, building the Bayes network, initializing the HMM parameters, generate sequence and states, estimate state transition and emission probability matrices, and evaluate the model. These are explained below.

4.1 Read the training Data

There are many standard data sets are available for building IDS, we have chosen the standard KDDCUP dataset. The description of the dataset has been given in section 3. The IDS Model is trained and tested with the KDDCUP99 dataset. After choosing the dataset it should be preprocessed. The preprocess task is given in the next section.

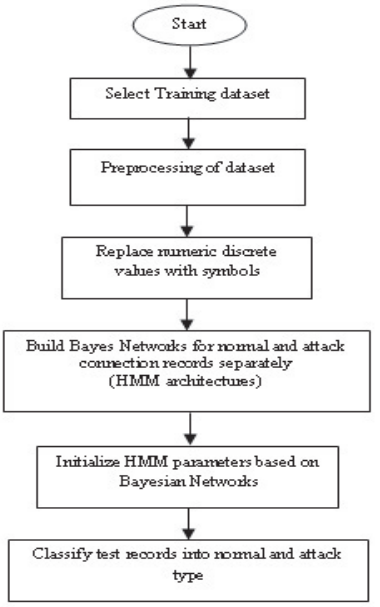

Figure 2. Framework for NIDS using Bayesian Network and HMM

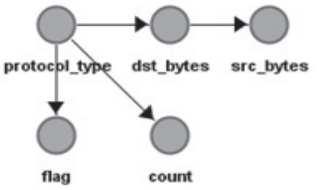

Figure 3. State Transition Diagram for normal records based on Bayes Network

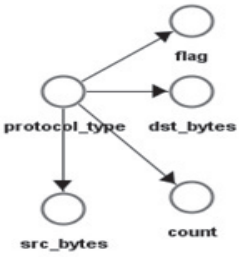

Figure 4. State Transition Diagram for attack type records based on Bayesian Network

4.2 Pre-processing of KDDCUP dataset

The standard KDDCUP dataset is in text format. It has 41 dimensions and 4,900,000 records of 10% of total size. The attributes of the data types are continuous, categorical, and binary types. We have taken samples of around 35,000 records with five attributes. The selected attributes and samples of connection records are listed in table 1. The format of text data is changed to comma separated values (CSV) which is easy to read and to analyze the data. With this we achieve data size reduction by reducing number of attributes from 41 to 5 and number records from 4,900,000 to 35,000.records. The discrete values have been replaced by symbols.

The preprocessed data has been used for training and testing the model. The chosen five attributes are protocol_type, flag, src_bytes, dst_bytes, and, count. We have discretized the continuous variables and the discrete values are represented by symbols. The discretized values represented by symbols are shown in table 2.

4.3 Build the Bayesian Network

The simplest kind of Dynamic Bayesian Network (DBN) is a Hidden Markov Model (HMM). A **dynamic Bayesian network** is a Bayesian network that represents sequences of state variables. These state variables represent the nodes of the graph. These sequences are often time-series or sequences of symbols. The hidden Markov model can be considered as a simple dynamic Bayesian network. The dependency or relationship between the variables (nodes) is shown by a directed edge. Each relationship (edge) is associated with conditional probability table (CPT). Each edge has one conditional table showing the relationship with all the remaining variables. The Bayesian networks for normal and attack type records of kddcup99 dataset were shown in figures 3 and 4.

4.4 HMM Parameters initializations

The IDS has been designed using KDDCUP'99 dataset. This dataset has been described above, which has 41 features. We have chosen five features out of 41 features as state hidden variables, namely protocol_type, flag, src_bytes, dst_bytes, and, count. These chosen variables are nothing but hidden state variables and each has set of distinct values which can emit a symbol. Each state variable has K distinct values say, V_1; V_2; . . . V_K, forming the observations X(t). The state transition diagrams of HMM for normal records and attack type records were shown in figures 3 and figure 4 respectively.

- The number of hidden state variables N is 5 based on the number of chosen variables. Therefore the size of the state transition matrix is 5x5.
- The number of distinct emission symbols is 18, so M is 18. Therefore the size of the emission transition probability matrix is 5x18.
- The initial probability distribution π = {0.000581, 0.261902, 0.08983, 0.375828, 0.271858};

The state transition matrices A_{norm} and A_{attack} were initialized with random values using the Bayesian networks and shown in tables 3 and 4 respectively. The emission probability matrices B_{norm} and B_{attack} were initialized based on the emission probability of state variables and are shown in tables 5 and 6 respectively.

4.5 Estimation of Transition and Emission Probabilities

To determine the parameters of HMM [14], it is necessary to make a rough guess for state transition probability matrix and for emission probability matrix values at what they might be. Once this is done, more accurate (in the maximum likelihood sense) parameters can be found by applying the so-called Baum-Welch re-estimation formulae [9]. Generally, the learning problem is how to adjust the HMM parameters, so that the given set of observations (called the *training set*) is represented by the model in the best way for the intended application. Here we are using Baum-Welch Algorithm [12] to train the IDS Model. The *Baum-Welch algorithm* is also known as *Forward-Backward algorithm*. The estimated HMM parameters A and B using the above algorithm are shown in tables 6 and 7 respectively.

4.6 Model Evaluation: The Evaluation Problem and the Forward Algorithm

We want to find the probability of an observed sequence, when the HMM [14] parameters (π, A, B) are known. Consider for our problem to find the probabilities of sequences corresponds to connection records of KDDCUP observations. The observations are: normal, dos, r2l, u2r and probe. In each of these observations, the record may have been normal, DoS, R2L, U2R, and Probe. We have taken only normal connection records, for which we have built the model. For this data we have shown hidden states as a trellis in figure 2 [10]. The Forward Algorithm [12] can be used for calculating the probability of sequence.

Table 1: Sample data set after discretization of attribute values

Protocol_type	flag	src_bytes	dst_bytes	count
udp	SF	<=162.155	<=334	<=377.616
icmp	SF	<=695.122	<=0	>377.616
tcp	SF	>695.122	>334	<=65.487
tcp	SF	<=695.122	>334	<=65.487
udp	SF	<=162.155	<=0	<=65.487
tcp	SF	<=695.122	>334	<=65.487
tcp	SF	<=695.122	>334	<=65.487
tcp	REJ	<=162.155	<=0	<=184.033
tcp	S0	<=162.155	<=0	<=184.033
tcp	SF	<=695.122	>334	<=65.487
icmp	SF	<=695.122	<=0	>377.616
tcp	SF	<=695.122	>334	<=65.487
icmp	SF	>695.122	<=0	>377.616
icmp	SF	>695.122	<=0	>377.616
tcp	SF	<=695.122	>334	<=65.487
tcp	S0	<=162.155	<=0	<=377.616
tcp	S0	<=162.155	<=0	<=184.033
tcp	SF	<=695.122	>334	<=65.487
icmp	SF	>695.122	<=0	>377.616
tcp	REJ	<=162.155	<=0	<=65.487
tcp	S0	<=162.155	<=0	<=184.033
tcp	SF	<=695.122	<=334	<=65.487
tcp	SF	<=695.122	>334	<=65.487
tcp	SF	<=695.122	>334	<=65.487
icmp	SF	>695.122	<=0	<=65.487

Table 2: After encoding discrete values with Symbols of Table 1.

Protocol_type	flag	src_bytes	dst_bytes	count
udp	SF	L	m	mid
icmp	SF	M	l	high
tcp	SF	H	h	low
tcp	SF	M	h	low
udp	SF	L	l	low
tcp	SF	M	h	low
tcp	SF	M	h	low
tcp	REJ	L	l	low
tcp	S0	L	l	low
tcp	SF	M	h	low
icmp	SF	M	l	high
tcp	SF	M	h	low
icmp	SF	H	l	high
icmp	SF	H	l	high
tcp	SF	M	h	low
tcp	S0	L	l	mid
tcp	S0	L	l	low
tcp	SF	M	h	low
icmp	SF	H	l	high
tcp	REJ	L	l	low
tcp	S0	L	l	low
tcp	SF	M	m	low
tcp	SF	M	h	low
tcp	SF	M	h	low
icmp	SF	H	l	low

Table 3: State Transition Matrix A_{normal} for normal records of figure 3

	Protocol_type	flag	src_bytes	dst_bytes	count
Protocol_type	0	0.33	0	0.33	0.33
flag	0	0	0	0	0
src_bytes	0	0	0	0	0
dst_bytes	0	0	1	0	0
count	0	0	0	0	0

Table 4: State Transition Matrix A_{attack} for attack records of figure 4

	Protocol_type	flag	src_bytes	dst_bytes	count
Protocol_type	0	0.4	0.3	0.2	0.1
flag	0	0	0	0	0
src_bytes	0	0	0	0	0
dst_bytes	0	0	0	0	0
count	0	0	0	0	0

We have a model $\lambda = (A, B, \pi)$ and a sequence of observations $O = O_1, O_2, \ldots, O_T$, and $p\{O \mid \lambda\}$ must be found. We can calculate this quantity using simple probabilistic arguments. But this calculation involves number of operations in the order of N^T. This is very large even if the length of the sequence, T is moderate. Therefore we have to look for another method for this calculation. Fortunately there exists one which has a considerably low complexity and makes use an auxiliary variable, $\alpha_t(i)$ called *forward variable*.

Table 5: Emission Transition Matrix B_{normal} for normal records based on the frequency of outcome for each state variable

	udp	icmp	tcp	SF	REJ	SO	RSTR	SH	RSTO	L	M	H	l	m	h	low	mid	high
Protocol_type	0.13	0.01	0.864	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
flag	0	0	0	0.79	0.08	0.13	0.002	0.005	0.002	0	0	0	0	0	0	0	0	0
src_bytes	0	0	0	0	0	0	0	0	0	0.205	0.69	0.1	0	0	0	0	0	0
dst_bytes	0	0	0	0	0	0	0	0	0	0	0	0	0.11	0.22	0.66	0	0	0
count	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.997	0.008	1E-04

Table 6: Emission Transition Matrix B_{attack} for attack records based on the frequency of outcome for each state variable

	udp	icmp	tcp	SF	REJ	SO	RSTR	SH	RSTO	L	M	H	l	m	h	low	mid	high
Protocol_type	0.01	0.613	0.377	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
flag	0	0	0	0.74	0.08	0.16	0.004	0.006	0.004	0	0	0	0	0	0	0	0	0
src_bytes	0	0	0	0	0	0	0	0	0	0.369	0.06	0.6	0	0	0	0	0	0
dst_bytes	0	0	0	0	0	0	0	0	0	0	0	0	0.97	0.01	0.02	0	0	0
count	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.228	0.148	0.624

The forward variable is defined as the probability of the partial observation sequence $O = O_1, O_2, \ldots, O_T$, when it terminates at the state *i*. Mathematically,

$$\alpha_t(i) = p\{O_1, O_2, \ldots, O_t, q_t = i \mid \lambda\} \quad (3)$$

The complexity of this method, known as the *forward algorithm* is proportional to N^2T , which is linear with respect to *T* whereas the direct calculation mentioned earlier, had an exponential complexity.

In a similar way we can define the *backward variable* $\beta_t(i)$ as the probability of the partial observation sequence: $o_{t+1}, o_{t+2}, \ldots, o_T$, given that the current state is *i*. Mathematically ,

$$\beta_t(i) = p\{o_{t+1}, o_{t+2}, \ldots, o_T \mid q_t = i, \lambda\} \quad (4)$$

5. Conclusion

This paper has described the usage of Hidden Markov Model for Intrusion Detection System. Training and testing of the KDD Cup 1999 dataset for IDS using HMM for applicator. We have taken only five features out of 41 features in our dataset. As described above, the HMM has been trained for normal TCP connection records of the KDD Cup 1999 data set. While training the model, it is necessary to initialize appropriate values, because the performance of the model mainly depends on these values. So for this we have initialized the parameter B using chosen dataset. After the initialization of A, B, and π parameters, the model selection is a major issue. Training is performed using standard Baum- Welch algorithm. The forward algorithm is suitable for to test the network traffic. The traffic is classified as normal or intrusion. Thus, this shows that Hidden Markov Methodology, with suitable parameter estimation and the training, represents a powerful approach for creating Intrusion Detection System which can find whether the traffic is normal or intrusions at runtime that might solve the major concern of the Computer Security. We are extending our work on a more rigorous data set for building a highly reliable intrusion detection system.

References

1. http://www.intrusiondetectionsystem.org/.
2. S. S. Joshi and V. V. Phoha, "Investigating hidden Markov models capabilities in anomaly detection," in *Proc. 43rd Annual Southeast Regional Conf. (ACM-SE 43)*, Kennesaw, GA, Mar. 2005, pp. 98–103.
3. KDD Cup 1999. Available on: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, October 2007.
4. http://en.wikipedia.org/wiki/Hidden_Markov_model#Applications_of_hidden_Markov_ models
5. http://jabrauum.blogspot.com/
6. http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/data/index.html
7. http://kdd.ics.uci.edu/databases/kddcup99/task.html
8. Stefano Zanero, "Behavioral Intrusion Detection," in Proceedings of *ISCIS 2004*, volume 3280 of Lecture Notes in Computer Science, pages 657-66, Kemer-Antalya, Turkey, October 2004.Springer.
9. http://www.ee.columbia.edu/ln/rosa/doc/HTKBook21/node7.html
10. http://www.comp.leeds.ac.uk/roger/HiddenMarkovModels/html_dev/forward_algorithm/s1_pg1.html
11. http://jedlik.phy.bme.hu/~gerjanos/HMM/hoved.html
12. Jiawei Han and Micheline Kamber, "Data Mining Concepts and Techniques", Second Edition, Morgan Kaufmann Publishers is an imprint of Elsevier, 2006.
13. Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani "A Detailed Analysis of the KDD CUP 99 Data Set", Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defence Applications(CISDA 2009).
14. L. R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings of the IEEE, Vol. 77, No. 2, February 1989, http://www.cs.ucsb.edu/~cs281b/papers/HMMs%20-%20 Rabiner.pdf
15. P Srinivasulu, D Nagaraju, P Ramesh Kumar, and K Nageswara Rao, "Classifying the Network Intrusion Attacks using Data Mining Classification Methods and their Performance Comparison" JCSNS International Journal of Computer Science and Network Security, VOL.9 No.6, June 2009.
16. Nagaraju Devarakonda, Srinivasulu Pamidi, V Valli Kumari, A Govardhan "Outliers Detection as Network Intrusion Detection System Using Multi Layered Framework" Advances in Computer Science and Information Technology: First International Conference on Computer Science and Information Technology, Springer COSIT2011, Jan 2011.