

Viterbi Algorithm for Intrusion Type Identification in Anomaly Detection System

january 14th 2019

Context

Intrusion Type

- . Buffer overflow
 - . xlock vulnerability
 - . lpset vulnerability
 - . kcms_sparc vulnerability
- . S/W security vulnerability
- . Setup vulnerability
- . Denial of service

Markov Chain

A markov Chain is defined by :

- . S , A finite set of N states
- . π , A vector of initial probabilities over S :

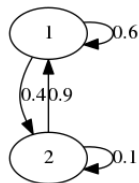
$$\pi_i = P(S_1 = i), 1 \leq i \leq N$$

- . A, A matrix of probabilities of transitions over $S \times S$:

$$a_{ij} = P(S_t = j | S_{t-1} = i), 1 \leq i \leq N$$

- Markov assumption :

$$P(S_t|S_{t-1}, S_{t-2}, \dots, S_1) = P(S_t|S_{t-1})$$



$$A = \begin{pmatrix} 0.6 & 0.4 \\ 0.9 & 0.1 \end{pmatrix}$$

Figure: Simple example of Markov Chain

HMM - Hidden Markov Model

- Hidden Markov Model is a statistical model in which the modeled system is supposed to be a Markovian process of unknown parameters.

HMM - Hidden Markov Model

- Hidden Markov Model is a statistical model in which the modeled system is supposed to be a Markovian process of unknown parameters.
- Hidden Markov Model can be viewed as a Bayesian Network

HMM - Hidden Markov Model

- Hidden Markov Model is a statistical model in which the modeled system is supposed to be a Markovian process of unknown parameters.
- Hidden Markov Model can be viewed as a Bayesian Network
- We define a HMM including :
 - V, A finite set of M observations
 - B, A a matrix of probabilities of observations over state :

$$b_i(k) = P(o_t = V_k | S_t = i)$$

HMM - Forward Algorithm

input : λ The model, O Observed sequence

output : $P(O|\lambda)$

Step 1, Initialization : $\forall i, \alpha_1(i) = \pi_i b_i(O_1)$

Step 2, Induction :

for $t \leftarrow 2 : T$ **do**

$$\alpha_t(i) = \left[\sum_{j=1}^N \alpha_{t-1}(j) a_{ij} \right] b_i(O_t)$$

1

end

Step 3, Termination : $P(O|\lambda) = \sum_{i=1}^N \alpha_t(i)$

¹L. R. Rabiner (1989). "A tutorial on hidden Markov models and selected applications in speech recognition". In: *Proceedings of the IEEE* 77.2, pp. 257–286



Normal Behaviour Modeling

Normal Behaviour is modelised by a left-to-right HMM λ .

The forward algorithm is used to decide whether normal or not with a threshold.

Intrusion Detection

Data

$$S = \{1, 2, 3, 4\}$$

$$M = \{1, 2, 3, 4\}$$

$$\pi = \{1.0, 0, 0\}$$

$$O = \{2, 1, 2, 4, 2, 3, 4, 3, 4, 3\}$$

$$A = \begin{pmatrix} 0.28 & 0.34 & 0.28 & 0 \\ 0.0 & 0.32 & 0.21 & 0.47 \\ 0.0 & 0.0 & 0.32 & 0.68 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{pmatrix}$$

$$B = \begin{pmatrix} 0.8 & 0.04 & 0.1 & 0.06 \\ 0.0 & 0.13 & 0.45 & 0.42 \\ 0.0 & 0.9 & 0.1 & 0.0 \\ 0.64 & 0.12 & 0.06 & 0.18 \end{pmatrix}$$

Intrusion Detection

Initialization

$$\alpha_1 = \begin{pmatrix} 0.04 & 0 & 0 & 0 \end{pmatrix}$$

Intrusion Detection

Induction

$$\alpha = \begin{pmatrix} 0.04 & 0 & 0 & \\ 0.00896 & 0 & 0 & 0 \\ 0.00010035 & 0.00039603 & 0.0022579 & 0 \\ 1.8882e^{-08} & 2.8849e^{-06} & 1.3193e^{-05} & 4.0995e^{-05} \\ 5.287e^{-10} & 4.1831e^{-07} & 4.8329e^{-07} & 3.0793e^{-06} \\ 8.8822e^{-12} & 5.6297e^{-08} & 0 & 6.4882e^{-07} \\ 2.487e^{-13} & 8.1081e^{-09} & 1.1825e^{-09} & 4.0517e^{-08} \\ 4.1782e^{-15} & 1.0898e^{-09} & 0 & 8.1237e^{-09} \\ 1.1699e^{-16} & 1.5693e^{-10} & 2.2885e^{-11} & 5.1816e^{-10} \end{pmatrix}$$

Intrusion Detection

Termination

$$P(0|\lambda) = \sum_{i=1}^N \alpha_t(i) = 6.9797e^{-10}$$



Intrusion Detection

Decision

```
if  $\log(P(0|\lambda)) > \text{threshold}$  then  
  | return Normal Behaviour  
else  
  | return Intrusion  
end
```

$$\log(P(0|\lambda)) = -21.083 < \text{threshold}(-20.83) \implies \text{Intrusion}$$

Intrusion Detection

Results

Table: The performance of HMM-based IDS. Best results are in bold

Length	Thresold	Detection Rate	F-P Error
10	-9.43	100%	2.626
15	-9.43	100%	3.614
10	-14.42	100%	1.366
15	-14.42	100%	2.718
10	-16.94	100%	0.789
15	-16.94	100%	2.618
10	-18.35	100%	0.553
15	-18.35	100%	2.535
10	-19.63	100%	0.476
15	-19.63	100%	2.508
10	-20.83	100%	0.372
15	-20.83	100%	2.473



Intrusion Type Identification

Process in two steps :

- Viterbi algorithm used to find the optimal state sequence

- Euclidian distance to identify the intrusion type with the optimal state sequence



Intrusion Detection

Data

$$S = \{1, 2, 3, 4\}$$

$$M = \{1, 2, 3, 4\}$$

$$\pi = \{1.0, 0, 0\}$$

$$O = \{2, 1, 2, 4, 2, 3, 4, 3, 4, 3\}$$

$$A = \begin{pmatrix} 0.28 & 0.34 & 0.28 & 0 \\ 0.0 & 0.32 & 0.21 & 0.47 \\ 0.0 & 0.0 & 0.32 & 0.68 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{pmatrix}$$

$$B = \begin{pmatrix} 0.8 & 0.04 & 0.1 & 0.06 \\ 0.0 & 0.13 & 0.45 & 0.42 \\ 0.0 & 0.9 & 0.1 & 0.0 \\ 0.64 & 0.12 & 0.06 & 0.18 \end{pmatrix}$$



Intrusion Type Identification

Initialization

$$\delta_1 = \begin{pmatrix} 0.04 & 0 & 0 & 0 \end{pmatrix}$$

$$\psi_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \end{pmatrix}$$



Intrusion Type Identification

Recursion

$$\delta = \begin{pmatrix} 0.04 & 0 & 0 & 0 \\ 0.00896 & 0 & 0 & 0 \\ 0.00010035 & 0.00039603 & 0.0022579 & 0 \\ 1.6859e^{-06} & 5.3227e^{-05} & 0 & 0.00027637 \\ 1.8882e^{-08} & 2.2142e^{-06} & 1.006e^{-05} & 3.3164e^{-05} \\ 5.287e^{-10} & 3.1885e^{-07} & 3.2192e^{-07} & 1.9899e^{-06} \\ 8.8822e^{-12} & 4.2853e^{-08} & 0 & 3.5817e^{-07} \\ 2.487e^{-13} & 6.1709e^{-09} & 8.9992e^{-10} & 2.149e^{-08} \\ 4.1782e^{-15} & 8.2937e^{-10} & 0 & 3.8683e^{-09} \\ 1.1699e^{-16} & 1.1943e^{-10} & 1.7417e^{-11} & 2.321e^{-10} \end{pmatrix} \quad \psi = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 1 & 3 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 0 & 3 \\ 0 & 1 & 1 & 3 \\ 0 & 1 & 0 & 3 \\ 0 & 1 & 1 & 3 \end{pmatrix}$$



Intrusion Type Identification

Termination

$$P^* = 2.321e^{-10}$$



Intrusion Type Identification

Backtracking

Optimal Sequence $S^* = \{1, 1, 3, 4, 4, 4, 4, 4, 4, 4\}$



Intrusion Type Identification

Decision

Table: Sequences for each type of intrusion

Type	Sequence	Distance
xlock	{2, 2, 3, 3, 3, 4, 4, 4, 4, 4}	3.7417
ipset	{2, 3, 3, 3, 4, 4, 4, 4, 4, 4}	4.4721
kcms_sparc	{1, 1, 2, 2, 2, 2, 4, 4, 4, 4}	3



Intrusion Type Identification

Results

Table: The performance of Viterbi-based Intrusion Type Identification

Attack	Trial	Correct	Incorrect	Rate
Buffer Overflow	20	18	2	90%
Denial of Service	25	9	16	36%
Buffer Overflow	45	27	18	60%

Limitations & Remarks

Try other distance metrics for Intrusion Type Identification :
[Ja-Min Koo and Sung-Bae Cho \(2005\)](#). “Effective Intrusion
Type Identification with Edit Distance for HMM-Based
Anomaly Detection System”. In: *Pattern Recognition and
Machine Intelligence*. Ed. by Sankar K. Pal,
Sanghamitra Bandyopadhyay, and Sambhunath Biswas.
Springer Berlin Heidelberg

Limitations & Remarks

Try other distance metrics for Intrusion Type Identification :
[Ja-Min Koo and Sung-Bae Cho \(2005\)](#). “Effective Intrusion Type Identification with Edit Distance for HMM-Based Anomaly Detection System”. In: *Pattern Recognition and Machine Intelligence*. Ed. by Sankar K. Pal, Sanghamitra Bandyopadhyay, and Sambhunath Biswas. Springer Berlin Heidelberg

Bad results for Denial of Service : [W. Bongiovanni et al. \(2015\)](#). “Viterbi algorithm for detecting DDoS attacks”. In: *2015 IEEE 40th Conference on Local Computer Networks (LCN)*

Methods using HMM

Other Methods