# N-그램 증강 나이브 베이스 알고리즘과 일반화된 k-절단 서픽스 트리를 이용한 확장 가능하고 정확한 침입 탐지 기법

강대기* · 황기현*

# Scalable and Accurate Intrusion Detection using n-Gram Augmented Naive Bayes and Generalized k-Truncated Suffix Tree

Dae-Ki Kang*, Gi-Hyun Hwang*

본 연구는 2008년도 동서대학교 학술연구조성비 지원 과제의 연구비와 동서대학교 유비쿼터스 어플라이언스 지역혁신 센터의 연구비를 지원받았음

## 요 약

기계 학습을 응용한 많은 침입 탐지 시스템들에서 n-그램 접근 방법이 사용되고 있다. 그러나, n-그램 접근 방법은 확장이 어렵고, 주어진 시퀀스에서 획득한 n-그램들이 서로 겹치는 문제들을 가지고 있다. 본 연구에서는 이러한 문제들을 해결하기 위해, 일반화된 k-절단 서픽스 트리 (generalized k-truncated suffix tree; k-TST) 기반의 n-그램 증강 나이브 베이스 (n-gram augmented naive Bayes) 알고리즘을 침입 시퀀스의 분류에 적용하여 보았다. 제안된 시스템의 성능을 평가하기 위해 n-그램 특징들을 사용하는 일반 나이브 베이스 (naive Bayes) 알고리즘과 서포트 벡터 머신 (support vector machines) 알고리즘과 본 연구에서 제안한 n-그램 증강 나이브 베이스 알고리즘을 호스트 기반 침입 탐지 벤치마크 데이터와 비교하였다. 공개된 호스트 기반 침입 탐지 벤치마크 데이터인 뉴 멕시코 대학(University of New Mexico)의 벤치마크 데이터에 적용해 본 결과에 따르면, n-그램 증강 방법이, n-그램이 나이브 베이스에 직접 적용되는 경우(예: n-그램 특징을 사용하는 일반 나이브 베이스), 생기는 독립성 가정에 대한 위배의 문제도 해결하면서, 동시에 더 정확한 침입 탐지기를 생성해 냄을 알 수 있었다.

## ABSTRACT

In many intrusion detection applications, n-gram approach has been widely applied. However, n-gram approach has shown a few problems including unscalability and double counting of features. To address those problems, we applied n-gram augmented Naive Bayes with k-truncated suffix tree (k-TST) storage mechanism directly to classify intrusive sequences and compared performance with those of Naive Bayes and Support Vector Machines (SVM) with n-gram features by the experiments on host-based intrusion detection benchmark data sets. Experimental results on the University of New Mexico (UNM) benchmark data sets show that the n-gram augmented method, which solves the problem of independence violation that happens when n-gram features are directly applied to Naive Bayes (i.e. Naive Bayes with n-gram features), yields intrusion detectors with higher accuracy than those from Naive Bayes with n-gram features and shows comparable accuracy to those from SVM with n-gram features. For the scalable and efficient counting of n-gram features, we use k-truncated suffix tree mechanism for storing n-gram features. With the k-truncated suffix tree storage mechanism, we tested the performance of the classifiers up to 20-gram, which illustrates the scalability and accuracy of n-gram augmented Naive Bayes with k-truncated suffix tree storage mechanism.

키워드

N-그램 나이브 베이스 알고리즘, 일반화된 k-절단 서픽스 트리, 호스트 기반 침입 탐지

* 동서대학교 컴퓨터 정보 공학부

접수일자 2009. 01. 28

## Ⅰ. 서 론

데이터 마이닝 알고리즘은 호스트 기반 침입 탐지 작업에서 프로그램의 트레이스(program trace)를 분류하는 데 널리 사용되어 왔다. 구체적으로 침입 탐지 작업에서 데이터의 전처리로서, 시스템 콜 트레이스에서의 특징 추출을 위해 n-그램(프로그램 트레이스 내에서 n 개의 연속된 시스템 콜)[1] 방법이 널리 사용되어 왔다[2-6]. 그러나, 이러한 n-그램 접근 방법은 침입 탐지에 적용되기에는 세 가지 심각한 문제점을 안고 있다.

1. 운영 체제에서 시스템 콜의 개수는 약 200 여개이므로, n-그램 방식의 특징들의 개수는 n이 증가하면 지수적으로 빠르게 증가한다. 예를 들면, 뉴 멕시코 대학의 벤치마크 데이터로 사용된 SunOS의 시스템 콜의 개수는 183 개인데, 만일 침입탐지시스템이 20-그램을 사용한다면 전체 특징들의 개수는 자그마치 18320=1,774,278,518,944,245,232,888,176,323,498,992,582,562,189,601 이나 되므로, 실제 상황에는 실용적이지 못하다.
2. n-그램 특징들은 고정된 크기의 윈도우를 사용하여 원래의 프로그램 트레이스로부터 생성된다. 따라서 프로그램 트레이스 내의 하나의 특정 시스템 콜이 적어도 n 개의 특징들 내부에 중복되어 포함되는 문제가 있다[7,8].
3. 만일 생성된 침입 탐지 시스템이, 예를 들면 나이브 베이스 알고리즘과 같이, 특징들 간의 통계적인 독립성에 대한 가정에 의지한다면, 2에서 언급한 n-그램 특징 생성 방법은 근본적으로 이러한 가정을 위배한다.

서포트 벡터 머신 (SVM)[9,10]과 같이 비선형적인 복잡도를 가지는 데이터 마이닝 알고리즘은 첫 번째 문제에 취약하다. 본 연구에서 우리는 이 첫 번째 문제를 구조적으로 해결하기 위해 강력한 스트링 색인 구조인 일반화된 k-절단 서픽스 트리[11,12]를 이용하였다. 두 번째와 세 번째 문제를 해결하기 위해 텍스트 분류 및 바이오인포매틱스 분야에서는 n-그램 증강 나이브 베이스 기법이 사용되어왔으나[7,8], 침입 탐지 분야에서는 이 기법이 연구된 바 없다. 게다가, 기존의 연구들은 n이, 예를 들면 15나 20 정도까지, 심각하게 큰 경우에 대해서 n-그램 특징 벡터를 제대로 사용하지 못하였다.

이러한 배경에 비추어, 우리는 n-그램 증강 나이브 베이스를 호스트 기반 침입 탐지 작업에 적용하였고, 그 성능을 n-그램 특징을 사용하는 나이브 베이스와 n-그램 특징을 사용하는 SVM과 비교하였다. 우리는 일반화된 k-절단 서픽스 트리 구조를 이용하여 시스템 콜 트레이스를 본 연구의 용도에 맞게 효율적으로 선형 시간(linear time)에 저장할 수 있었고, 각각의 n-그램 특징들에 대해 침입 탐지기가 필요로 하는 개수 (count) 정보를 상수 시간(constant time)에 제공할 수 있었다.

호스트 기반 침입 탐지 벤치마크 데이터에 대해 행한 실험 결과에 따르면, 본 연구에서 응용한 n-그램 증강 나이브 베이스가 n-그램 특징을 사용하는 나이브 베이스, 즉 특징들 간의 통계적인 독립성 가정을 위배하는 문제를 가지고 있는 나이브 베이스 알고리즘보다 더 좋은 결과를 보였다. 또한, n-그램 특징을 사용하는 SVM와 비슷한 정확도를 보였다.

본 논문의 순서는 다음과 같다. 2 장에서는 우리의 연구 방법에 대해 기술하도록 하겠다. 3 장에서는 실험 결과를 기술하고, 4 장에서는 연구 결과에 대한 토의와 관련 연구를 기술하겠다.

## Ⅱ. 연구 방법

우리는 호스트 기반 침입 탐지 (host-based intrusion detection) 문제를 형식적으로 정의하고, n-그램 특징을 사용하는 나이브 베이스 (NB n-gram)과 n-그램 증강 나이브 베이스(NB(n)), 그리고 n-그램 특징을 사용하는 SVM (SVM n-gram)에 대해 설명하고자 한다.

우선, 호스트 기반 침입 탐지 (host-based intrusion detection) 문제를 형식적으로 정의해보고자 한다. $\Sigma=\{s_1,s_2,s_3,\ldots,s_m\}$을 시스템 호출(system call)들의 집합으로 정의하면, $m=|\Sigma|$이며, 데이터 집합 D는 레이블이 붙은 시퀀스 (즉 트레이스)의 집합들 $D=\{\langle Z_i,c_i\rangle | Z_i\in\Sigma^*, c_i\in\{0,1\}\}$로 정의될 수 있다. 여기서, $Z_i=z_1,z_2,z_3,\ldots,z_l$은 길이가 l인 입력 시퀀스이고, $c_i$는 이 입력 시퀀스에 상응하는 클래스 레이블로, 0은 침입이 아님을 뜻하고, 1은 침입을 뜻한다. 이렇게 데이터 집합 D가 주어지면, 침입 탐지 학습 알고리즘의 목표는 정확도, F1-measure, 탐지율(detection rate),

거짓양성율(false positive rate)과 같은 주어진 평가 기준을 최대화하는 침입 탐지기 $h:\Sigma^{*}\rightarrow\{0,1\}$를 발견하는 것이다.

만일, 예를 들어, 나이브 베이스(Naïve Bayes)와 같은 확률적인 모델을 침입 탐지기 h 에 사용한다면, 결과적인 확률 모델 $P_h$는 주어진 시퀀스 Z 에 대해 다음과 같이 확률 $P_h(z_1,z_2,z_3,...,z_l)$를 설정한다.

1. 각 클래스 $c_i$에 대해, $c_i$와 연관된 시퀀스들을 샘플링하여 확률 $P_h(c_i)$를 추정함
2. 새로운 시퀀스 Z 에 대하여 클래스 c 를 다음 식에 근거하여 설정함

$$c_h = \arg max_{c\in\{0,1\}} P_h(Z=z_1,z_2,z_3,...,z_l|c) \cdot P_h(c) \quad (1)$$

### 2.1 나이브 베이스 분류기 (Naïve Bayes Classifier)

호스트 기반 침입 탐지기로서의 나이브 베이스 분류기의 중요한 가정 중 하나는, 주어진 클래스에 대해 시퀀스의 각 시스템 콜이 서로 독립적이라는 것이다. 그러므로, 나이브 베이스의 경우 (식 1에서 보인 바와 같이) 새로운 시퀀스에 대한 분류는 다음과 같이 형식화될 수 있다.

$$c_{NB} = \arg max_{c\in\{0,1\}} P_h(c) \cdot \prod_i P_h(z_i|c)$$

나이브 베이스 분류기가 텍스트 또는 단백질 시퀀스 (protein sequence) 분류에 적용되는 경우, 분류기의 알고리즘은 각각의 문서나 시퀀스를 단어나 아미노산을 나타내는 문자들의 백(bag) 또는 집합(set)으로 다룬다[8,13]. 시스템 콜의 집합(set)이나 백(bag 또는 multiset) 표현을 침입 탐지 작업에 적용해 본 연구[14,15,16]가 다소 있기는 하지만, 대부분의 침입 탐지 연구[2,3,5,6,17,18,19]는 n-그램 표현에 초점을 맞추고 있다.

### 2.2 N-그램 특징을 사용하는 나이브 베이스 분류기 (NB n-gram)

각 시퀀스들은 길이가 고정되어 있지 않으므로, 고정되고 유한한 크기의 입력을 받는 컴퓨터 알고리즘에 적용하는 데에는 어려움이 있다. 따라서, 주어진 시퀀스는 유한한 크기의 n 차원 특징 벡터(n-그램)로 변환된다.

호스트 기반 침입 탐지 작업에서는 프로그램의 행동을 모니터하게 되고, 이를 위해서 프로그램의 트레이스를 시퀀스로 간주한다. 따라서, n-그램 특징들을 생성하기 위해서, 길이가 n 인 슬라이딩 윈도우를 트레이스의 시작부터 끝까지 시스템 콜 하나씩 옮겨가면서, 주어진 트레이스로부터 n-그램 특징들을 생성해 낸다.

특징 추출이 끝나면, 생성된 n-그램들에 대한 확률적 모델은 다음과 같이 형식화될 수 있다. 여기서 $l$ 은 전체 시퀀스의 길이이다.

$$c_{NB\,n-gram} = \arg max_{c\in\{0,1\}} P_h(c) \cdot \prod_{i=1}^{l-n+1} P_h(z_i,...,z_{i+n-1}|c)$$

이러한 n-그램 특징을 사용하는 나이브 베이스 모델은 한 가지 심각한 문제점을 가지고 있다. 그것은 n-그램 특징이 슬라이딩 윈도우를 통해 생성되는 동안, 트레이스 안의 특정 시스템 콜 하나가 많으면 n 번 이상 슬라이딩 윈도우 안에 포함된다는 것이다. 결국 서로 이웃한 n-그램 특징은 구조적으로 서로 독립적이지 않으므로, 나이브 베이스 학습 알고리즘의 독립성 가정에 위배되게 된다.

### 2.3 N-그램 증강 나이브 베이스 (NB(n))

앞서 언급한 문제를 해결하기 위해, Peng과 Schuurmans[7]은 n-그램 증강 나이브 베이스를 도입하여 텍스트 분류 문제에 적용하였다. 그들은 시퀀스로부터 만들어진 n-gram 특징 내부에 있는 요소들 간의 의존성을 명시적으로 모델링하는 접근 방법을 택했다. 그림 1 은 이러한 접근 방법으로 시퀀스 내의 여섯 개의 서로 이웃한 요소들의 의존 관계를 표현한 것이다.

정션 트리 정리 (Junction tree theorem) [20]에 의해, n-그램 증강 나이브 베이스의 확률적 모델은 다음과 같이 형식화될 수 있다.

$$c_{NB(n)} = \arg max_{c\in\{0,1\}} P_h(c) \cdot \frac{\prod_{i=1}^{l-n+1} P_h(z_i,...,z_{i+n-1}|c)}{\prod_{i=2}^{l-n+1} P_h(z_i,...,z_{i+n-2}|c)}$$

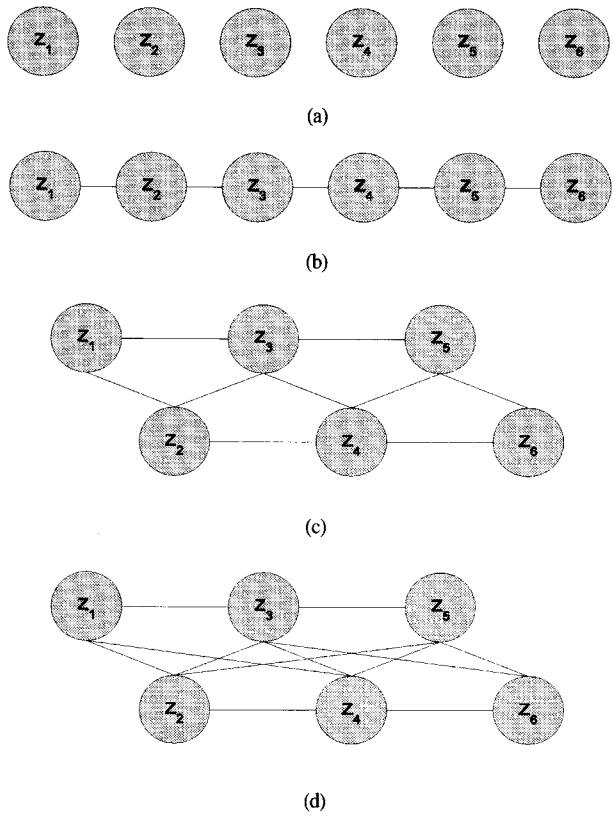


그림 1. 시퀀스 안의 연속된 여섯 개의 요소들 간의 의존도를 표현한 그래프 모델들
(a) NB(1) = 나이브 베이스
(b) 임의의 요소와 그 요소의 선행 요소 간의 의존 관계를 묘사한 NB(2)
(c) 임의의 요소와 그 요소의 바로 다음 요소와 그 다음 요소 간의 의존 관계를 묘사한 NB(3)
(d) 임의의 요소와 그 요소와 이웃한 세 개의 요소 간의 의존 관계를 묘사한 NB(4)
Figure 1. Graphical Models that represent dependencies among six elements in the sequence
(a) NB(1)=Naive Bayes
(b) NB(2) depicts a dependency between an element and its direct successor
(c) NB(3) depicts dependencies among an element, its 1st successor, and its 2nd successor
(d) NB(4) depicts dependencies among an element and its three closest successors

그림 1과 위의 식에서 알 수 있듯이, n-그램 증강 나이브 베이스의 확률적 그래피컬 모델은, 근본적으로 마르코프 네트워크(Markov Network)으로 그 확률 분포는 최대 클리크(clique, 최대한 연결된 서브그래프)들의 마지널(marginal)들의 곱을 세퍼레이터(separator, 클리크들 간의 오버랩들)들의 마지널(marginal)들의 곱으로 나누어서 구해진다.

### 2.4 n-그램 특징을 사용하는 SVM

n-그램 증강 나이브 베이스의 성능을 다른 데이터 마이닝 알고리즘과 비교해 보기 위해, 우리는 n-그램 특징들을 사용하는 SVM을 고려해보았다. 즉, 우리는 원래의 프로그램 트레이스에서 n-그램 특징들이 구하고, 구해진 특징들은 선형 커널을 사용하는 SVM 알고리즘의 입력으로 사용하였다.

표 1. UNM live lpr 데이터에 대해 NB(n), NB n-gram, SVM n-gram의 정확도와 거짓양성율을 비교한 결과
Table 1. Experimental results of NB(n), NB n-gram, SVM n-gram on UNM live lpr data

| n | NB(n) | | NB n-gram | | SVM n-gram | |
|---|---|---|---|---|---|---|
| | 정확도 | 거짓양성 | 정확도 | 거짓양성 | 정확도 | 거짓양성 |
| 1 | 84.09±0.02 | 28.84±0.02 | 84.09±0.02 | 28.84±0.02 | **100.00±0.00** | **0.00±0.00** |
| 2 | 99.78±0.00 | 0.41±0.00 | 98.30±0.01 | 3.09±0.01 | 99.96±0.00 | 0.00±0.00 |
| 3 | 99.96±0.00 | 0.00±0.00 | 99.01±0.01 | 1.79±0.01 | N/A | N/A |
| 4 | 99.96±0.00 | 0.00±0.00 | 99.60±0.00 | 0.73±0.00 | N/A | N/A |
| 5 | 99.96±0.00 | 0.00±0.00 | 99.82±0.00 | 0.32±0.00 | N/A | N/A |
| 6-8 | **100.00±0.00** | **0.00±0.00** | **99.87±0.00** | **0.24±0.00** | N/A | N/A |
| 9-10 | 99.96±0.00 | 0.08±0.00 | 99.82±0.00 | 0.32±0.00 | N/A | N/A |
| 11-20 | 99.96±0.00 | 0.08±0.00 | 99.78±0.00 | 0.41±0.00 | N/A | N/A |

우리의 관심사는 n-그램 증강 나이브 베이스와 n-그램 특징을 사용하는 나이브 베이스를 n-그램 특징을 사용하는 SVM과 비교하는 것인데, 그 이유는 n-그램 특징을 사용하는 나이브 베이스와 달리, SVM은 독립성에 대한 가정에 의존하지 않기 때문이다.

그러나, SVM 의 커널 매트릭스를 준비하는 데만, 적어도 $O(n^2)$ 의 시간 및 공간 복잡도를 가진다.

즉, SVM 알고리즘은 비선형 복잡도를 가지므로, 입력 변수의 개수 n이 증가함에 따라 필요한 저장 공간은 나이브 베이스 같은 선형 복잡도를 가지는 알고리즘보다 더 빠르게 증가하는 차원의 저주(curse of dimensionality) 문제가 있다. 이러한 이유로, 실제 실험에서는, 이러한 컴퓨팅 및 메모리 문제로, 우리는 SVM을 n 이 1 또는 2 인 경우에 대해서만 수행할 수 있었다.

### 2.5 일반화된 k-절단 서픽스 트리

서픽스 트리(suffix tree)는 스트링이나 시퀀스를 색인하는 데 효과적인 자료구조[21,22]이다. 만일 스트링의 길이가 l 이라면, 서픽스 트리를 구성하는 시간 복잡도는 O(l)이며, 일단 서픽스 트리가 생성되고 나면 길이가 m인 패턴 스트링을 찾는 데, 단지 O(m)의 시간 복잡도가 걸린다. 또한 에지 라벨 압축을 통해 서픽스 트리를 저장하는 공간 복잡도도 O(l)이 가능하다. Ukkonen [21] 이나 그밖의 연구들을 통해 선형 시간에 서픽스 트리를 만들 수 있는 알고리즘들이 고안된 바 있다. 그러나 스트링이나 시퀀스가 매우 길고, 그 중에서 본 연구의 경우와 같이 n-그램의 서브스트링(substring)들만 필요한 경우에는 미리 주어진 길이의 서브스트링들만을 저장하는 색인 구조인 k-절단 서픽스 트리[11,12]가 더 편리하다. 또한 대부분의 실제 문제들처럼 여러 개의 스트링들을 하나의 트리에 색인하여 저장하기 위해서는, 주어진 스트링 집합의 모든 서픽스들을 저장할 수 있는 일반화된 서픽스 트리(Generalized Suffix Tree; GST)가 사용된다.

따라서, 본 연구에서는 여러 개의 시스템 콜 트레이스(trace)들을 저장하고 이러한 트레이스들에서 n-그램 특징들만 추출해서 사용하므로, 본 응용을 위한 학습과 테스트를 수행하는 데는 일반화된 k-절단 서픽스 트리를 사용하였다.

## III. 실험 및 결과

### 4.1 결과 요약

n-그램 증강 나이브 베이스(NB(n))의 성능을 평가하기 위해, 우리는 그 성능을 n-그램 특징을 사용하는 나이브 베이스(NB n-gram)와 n-그램 특징을 사용하는 SVM(SVM n-gram)과 비교하였다. 실험을 위한 데이터로 공개적으로 사용가능한 뉴 멕시코 대학(University of New Mexico)의 **"UNM live lpr"**, **"UNM live lpr MIT"**, 그리고 **"UNM denial of service"** 시스템 콜 트레이스들을 사용하였다.[1)]

표 1은 "UNM live lpr" 데이터에 대해 이러한 세 개의 알고리즘의 결과로 나온 정확도(accuracy)와 거짓 양성

1) **"UNM live lpr MIT"**, 그리고 **"UNM denial of service"** 데이터의 결과는, 본 논문에서는 지면 문제로 생략하였다.

(false positive) 값을 나타낸 것이다. 정확도와 거짓양성율은 10겹 교차검증을 이용하여 측정되었고, 99%의 신뢰 수준에서 신뢰 구간을 측정하였다. 결론을 말하면, n-그램 증강 나이브 베이스는 n이 6에서 8일 때 가장 좋은 성능을 보였다. 이 때 NB(n)의 정확도는 100.00 이고 거짓 양성 값은 0.00 이며, NB n-gram의 경우는 정확도는 99.87이고 거짓 양성 값은 0.24이다. 이 경우, NB(n)이 보여준 성능은 SVM의 경우와 필적하는 수준이었다. SVM n-gram은 NB(n) 과의 차이는 그다지 현저하지 않으나, n이 단지 2인 경우에도 183×183 =33489 개의 어트리뷰트로 구성된 특징 벡터와 그 제곱 배의 커널 매트릭스를 가지게 되어 학습 단계에서 discriminative model을 학습하기 위해 몇 시간이 넘는 학습 시간을 필요로 하였으나, generative model을 학습하는 NB(n)의 경우 전술한 공식들을 계산하는 데 필요한 값을 서픽스 트리에서 구하면 되므로 1~2 분 정도의 학습 시간 밖에 걸리지 않았다.

### 4.2 결과 고찰 및 선행 연구와의 비교

Peng 등[7]이 n-그램 증강 나이브 베이스 알고리즘을 고안하였고, 텍스트 분류에 적용하였다. 그러나, 침입 탐지 문제를 시퀀스 분류 문제로 간주하고 이에 대해 n-그램 증강 나이브 베이스를 적용한 예는 없었다.

Rieck과 Laskov[5]가 언어 모델을 알려지지 않은 네트워크 공격을 찾아내는 데 적용한 바 있다. 그들은 trie 데이터 구조를 사용하였고, 두 개의 trie 들 간의 거리를 구하는 방법으로 시퀀스 들간의 유사도를 계산하였다. 일반화된 k-절단 서픽스 트리는 n-그램 특징을 저장하는 데 있어 더 유리한 방법이다. 그 이유는 서픽스 트리는 시퀀스를 저장하는 데 선형 시간이 걸리는 것은 물론, 주어진 패턴을 찾는 데에도 그 패턴의 길이만큼의 시간만 걸리기 때문이다. 우리는 이러한 자료 구조를 통해 n이 1에서 20인 경우까지 계산하였지만, Rieck과 Laskov는 n이 1,3 그리고 5인 경우에 대해서만 계산하였다.

Shafig 등[6]은 조건부 n-그램을 맬웨어 검출에 사용하였다. 그들은 서로 연결된 n-gram 값들의중복되지 않는 조건부 표현을 Markov n 그램으로 정의하였는 데, 결국 고전적인 베이시안 분류기의 확률 값과 동일하다. 그들은 또한 엔트로피 레이트(entropy rate)를 분류를 위한 임계값으로 사용하였다. 우리가 선택한 방법과 비교해 볼 때, 그들의 방법은 조건부 확률을 큰 매트릭스에 저장해야 하기 때문에 확장이 쉽지 않다.

Forrest 등[17]은 Sequence Time-Delay Embedding (STIDE) 침입 탐지기를 고안하였다. 이것은 근본적으로 n-그램 접근 방법이며 내부적으로 5~6 개의 사용자가 조정해 주어야 하는 임계값을 가지고 있다. Tan과 Maxion[23]은 이 STIDE의 운영 상의 한계값에 대한 연구를 수행하였고, 알려지지 않은 공격을 감지하는 데에는 6-그램이 제일 적당하다는 결과를 제시하였다. 그러나, 본 연구에서 우리는 n-그램 증강 방식이 n-그램 방식보다 더 정확할 뿐만 아니라, 6이 언제나 매직 넘버는 아님을 보였다.

대부분의 침입 탐지 기술[5,6,17,18,19]은 n 그램 접근 방법을 사용하고 있다. 그러나 다른 접근 방법을 사용하는 연구들도 다소 수행된 바 있다. Liao와 Vemuri[14]는 bag of words 모델을 호스트 및 네트워크 시퀀스에 적용하고 k-nearest neighbor 분류기로 침입 탐지를 수행한 바 있다. Kang 등[15]은 bag of words 및 set of words 모델을 호스트 시퀀스에 적용하여 Naïve Bayes, 결정 트리, SVM 등의 다양한 기계 학습 알고리즘을 수행한 결과를 제시하였다. Liu[16] 등은 다양한 시스템 콜의 표현을 비교해 보고, 내부자 위협(insider threat)에 대해서는 시스템 콜만으로는 불충분하다는 결론을 제시하였다.

## Ⅳ. 결론

본 연구에서는 n-그램 증강 나이브 베이스 (n-gram augmented naive Bayes) 알고리즘을 침입 시퀀스의 분류에 적용하였다. 제안된 시스템의 성능을 평가하기 위해 n-그램 특징들을 사용하는 일반 나이브 베이스 (naive Bayes) 알고리즘과 서포트 벡터 머신 (support vector machines) 알고리즘과 본 연구에서 제안한 n-그램 증강 나이브 베이스 알고리즘을 비교하였다. 뉴 멕시코 대학의 벤치마크 데이터에 적용해 본 결과에 따르면, n-그램 증강 방법이, n-그램이 나이브 베이스에 직접 적용되는 경우(예: n-그램 특징을 사용하는 일반 나이브 베이스), 생기는 독립성 가정에 대한 위배 문제도 해결하면서, 동시에 n-그램 특징을 사용하는 일반 나이브 베이스보다 더 정확하며, n-그램 특징을 사용하는 SVM과 필적할만한 수준의 침입 탐지기를 생성해 냄을 알 수 있었다.

## 감사의 글

본 연구는 2008년도 동서대학교 학술연구조성비에 의하여 이루어진 연구이며, 심사위원님들께도 감사드립니다.

## 참고문헌

[ 1 ] E. Charniak, Statistical Language Learning, MIT Press, Cambridge, MA, USA, 1994.

[ 2 ] S. A. Hofmeyr, S. Forrest, and A. Somayaji, Intrusion detection using sequences of system calls, Journal of Computer Security, vol. 6, no. 3, pp. 151-180, 1998.

[ 3 ] W. Lee, S. J. Stolfo, and K. W. Mok, A data mining framework for building intrusion detection models, in: IEEE Symposium on Security and Privacy, pp. 120-132, 1999.

[ 4 ] A. Murali and M. Rao, A survey on intrusion detection approaches, in: First International Conference on Information and Communication Technologies (ICICT 2005), pp. 233-240, 2005.

[ 5 ] K. Rieck and P. Laskov, Detecting unknown network attacks using language models., in: Proceedings of Third International Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA 2006), Berlin, Germany, pp. 74-90, 2006.

[ 6 ] M. Z. Shafiq, S. A. Khayam, and M. Farooq, Embedded malware detection using markov n-grams., in: Proceedings of the Fifth Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA 2008), 2008.

[ 7 ] F. Peng and D. Schuurmans, Combining naive Bayes and n-gram language models for text classification., in: F. Sebastiani (Ed.), Advances in Information Retrieval, 25th European Conference on IR Research (ECIR 2003), Vol. 2633 of Lecture Notes in Computer Science, Springer, pp. 335-350, 2003.

[ 8 ] C. Andorf, A. Silvescu, D. Dobbs, and V. Honavar, Learning classifiers for assigning protein sequences to gene ontology functional families, in: Proceedings of the Fifth International Conference on Knowledge Based Computer Systems (KBCS 2004), pp. 256-265, 2004.

[ 9 ] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In COLT '92: Proceedings of the fifth annual workshop on Computational learning theory, pages 144-152, New York, NY, USA, 1992.

[10] V. N. Vapnik. The nature of statistical learning theory. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[11] J. C. Na and K. Park, Data compression with truncated suffix trees Proceedings of Data Compression Conference 2000, p. 565, 2000.

[12] M. H. Schulz, S. Bauer, and P. N. Robinson, The generalised k-Truncated Suffix Tree for time-and space-efficient searches in multiple DNA or protein sequences International Journal of Bioinformatics Research and Applications, 4(1), pp. 81-95, 2008.

[13] T. M. Mitchell, Machine Learning McGraw-Hill, 1997.

[14] Y. Liao, and V. R. Vemuri, Using Text Categorization Techniques for Intrusion Detection Proceedings of the 11th USENIX Security Symposium, USENIX Association, 51-59, 2002.

[15] D. Kang, D. Fuller, and V. Honavar, Learning Classifiers for Misuse and Anomaly Detection Using a Bag of System Calls Representation Proceedings of 6th IEEE Systems Man and Cybernetics Information Assurance Workshop (IAW), 2005

[16] A. Liu, C. Martin, T. Hetherington, and S. Matzner, A Comparison of System Call Feature Representations for Insider Threat Detection Proceedings of 6th IEEE Systems Man and Cybernetics Information Assurance Workshop (IAW), 2005

[17] S. Forrest, A. S. Perelson, L. Allen, and R. Cherukuri, Self-Nonself Discrimination in a Computer SP '94: Proceedings of the 1994 IEEE Symposium on Security and Privacy, IEEE Computer Society, 202, 1994.

[18] W. Lee, and S. Stolfo, Data mining approaches for intrusion detection Proceedings of the 7th USENIX

Security Symposium, 1998.
[19] C. Warrender, S. Forrest, and B. A. Pearlmutter, Detecting Intrusions using System Calls: Alternative Data Models IEEE Symposium on Security and Privacy, 133-145, 1999.
[20] R. G. Cowell, S. L. Lauritzen, A. P. David, D. J. Spiegelhalter, D. J. Spiegelhater, Probabilistic Networks and Expert Systems, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1999.
[21] E. Ukkonen, On-line construction of suffix-trees Algorithmica, 14, 249-260, 1995.
[22] D. Gusfield, Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology Cambridge University Press, 1997
[23] K. M. C. Tan, and R. A. Maxion, "Why 6?" Defining the Operational Limits of STIDE, an Anomaly-Based Intrusion Detector Proceedings of the 2002 IEEE Symposium on Security and Privacy, IEEE Computer Society, 2002, 188

## 저자소개

강대기 (Kang, Dae-Ki)

1992년 : 한양대학교 전자계산학과 졸업
1994년 : 서강대학교 전자계산학과 (이학 석사)
1994년~1999년 : 한국전자통신연구원 (연구원)
2006년 : Iowa State University(PhD in Computer Science)
2007년2월~2007년8월 : 국가보안기술연구소 (선임연구원)
2007년9월~현재 : 동서대학교 컴퓨터정보공학부 전임강사
※관심분야 : 기계학습, 관계학습, 통계적그래피컬모델, 온톨로지학습, 침입탐지, 웹방화벽, 웹마이닝, 컴퓨터비전

황기현(Gi-Hyun, Hwang)

1996년 : 부산대학교 전기공학과 석사 졸업.
2000년 : 부산대학교 전기공학과 박사 졸업.
2003년 동서대학교 컴퓨터정보공학부 교수
※관심분야: RFID, 임베디드, 영상처리