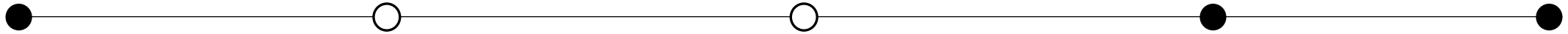
네트워크 침입탐지 기술 연구 동향

저자: 김동훈, 손인수
저널: 대한전자공학회
연도: 2019

콘텐츠IT전공
20195178 서영재

Contents



서론

침입 데이터 모델

침입 탐지 기술 연구 동향

분석 및 평가

결론

서론

서론

최근 인터넷이 급격하게 발달하면서 네트워크 상에서 공유되고 송수신되는 정보량 상승

정보들을 악용하기 위해 불법적으로 정보들을 해킹하려는 수법 또한 지능적으로 진화

사이버 공격들로 인한 침입들에 대응하기 위해서 우리는 방화벽이나 침입탐지 시스템을 구축해야 함

침입탐지 시스템

침입(공격)으로 탐지된 유형에 대해서만 접근을 차단하는 방식

네트워크 기반의 침입탐지 시스템(NIDS)

네트워크 침입탐지 시스템은 네트워크를 전반적으로 감시

네트워크상의 데이터 단위인 패킷을 분석하여 침입을 탐지하는 시스템

침입기법의 종류

루트킷	침입기법의 종류로는 침입한 경로의 흔적을 숨기고 해킹 시도
패스워드 크래킹	패스워드 정보를 빼내 침입을 시도
버퍼 오버플로우	데이터의 길이와 형태의 불명확한 정의를 파고들어 침입을 시도
패킷 스니핑	네트워크를 통과하는 정보를 볼 수 있는 패킷 스니퍼 프로그램을 악용
IP 스푸핑	IP 주소를 강탈해 권한을 취득

침입탐지 시스템 내에서 작용하는

침입모델

오용 탐지
Misuse Detection

전문가가 찾아낸 특정 침입 또는 공격 패턴을
시스템에 입력해두어 이 패턴과 시스템에 들어오는
데이터의 패턴이 일치하면 공격으로 판정

정확하게 전문가가 찾아낸 패턴과
일치해야 공격으로 데이터를 판정하므로 공격자가 패턴
을 약간 우회하여 공격할 경우 탐지하기 어려움

해결책

비정상 탐지
Anomaly Detection

일반적으로 시스템 내에서 학습되어진 동작과
다른 것으로 식별되는 네트워크 내의 패킷이나
데이터를 발견하여 이를 침입으로 판정

침입탐지 기술에 기계학습 기법 이용

데이터 수집
Data Acquisition

정형화된 데이터는
네트워크 침입이 일어날 때
발생되는 정보 등에서 추출

데이터 전처리
Data Preprocessing

추출한 데이터를
학습과정을 진행하기 위한
적합한 데이터로 가공

특징 선택
Feature Selection

학습과정을
더 좋은 성능으로
진행하기 위해
특징 선택의 과정

분류 또는 예측
Classification or Prediction

최종적으로 가공된 데이터와
분류할 카테고리 지정하여
학습을 진행

⬇

새로운 데이터를 입력 시,
분류 또는 예측을
할 수 있는 값을 출력

서론

기계학습 기반의 침입 탐지에 사용되는 데이터는 매우 다양할 수 있으며
일반적으로 우리가 정해 놓은 침입유형 데이터의 특징과 정상적인 유형의 데이터의 특징이 나열되어 있음

네트워크 기반 침입 탐지 시스템을 만들기 위해서는
네트워크의 패킷 정보가 주를 이루는 데이터 모델을 만들 수도 있을 것이다.

보안 전문가들은 이런 데이터 모델을 만들기 위해 이전부터 연구해왔으며,
본 논문에서 현재까지 진행되고 연구된 데이터 모델 몇 가지에 대해 소개하고 분석한 뒤
기계학습을 활용한 침입 탐지 모델을 구현하여 분석하고자 한다

침입 데이터 모델

침입 데이터 모델

고려사항 데이터를 분류하는 모델의 구조를 어떻게 분류되도록 구성하고 입력과 출력에 어떤 데이터를 제공 할 것인가

기계학습 기반 침입 탐지 시스템은 일반적으로 공격 유형과 정상 유형을 분류하는 지도 학습으로 진행

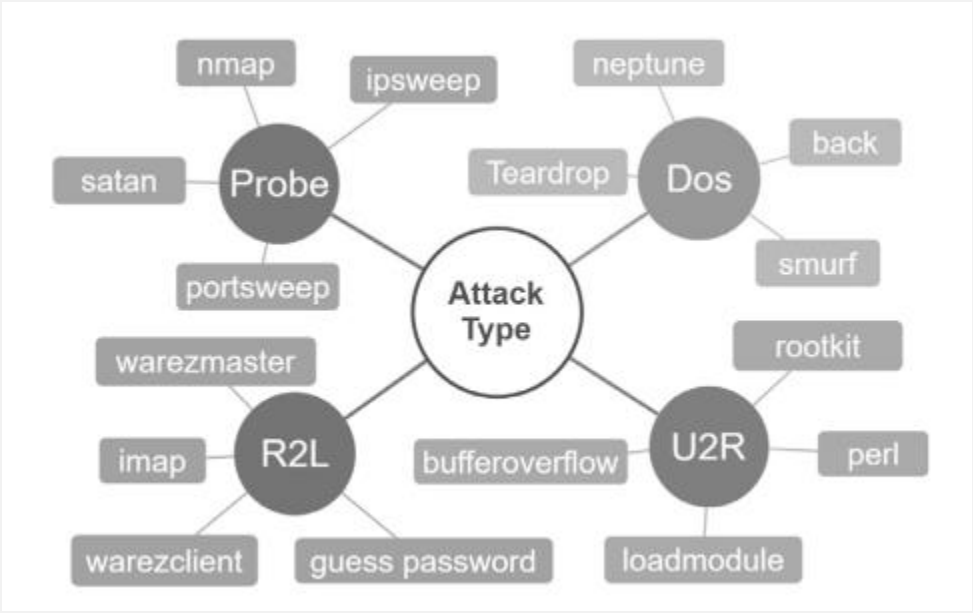
특징(Feature)과 대상 또는 라벨(Label)을 지정

- 대상 또는 라벨(Label)이 공격 유형 또는 정상 유형의 종류를 나타냄

탐지해야 할 대상 : 공격 유형

여러 공격 유형이 존재함에 따라 대응책이 다름

공격방법에 따른 공격 유형



침입 데이터 모델

침입 탐지 시스템을 생성할 프로그래밍 툴이
gz확장자로 이루어져 있는 Dataset을 읽어올 수 있는 함수를 내장하고 있다면

URL주소를 입력하여 인터넷에서 바로 다운로드 되어 시스템 상에 표현이 가능하지만,

그렇지 않다면 gz확장자로 압축된 Dataset의 압축을 풀고 csv확장자로 바꿔 Excel로 확인하는 방법을 사용한다.

이 때도 역시 csv파일을 읽어올 수 있는 프로그래밍 툴을 사용하여 침입 탐지 시스템을 생성하는 것이 바람직하다.
아래 그림 2는 Python 언어를 기반으로 하는 Spyder 프로그래밍 툴의 Keras 라이브러리를 이용하여
kddcup.data_10_percent.gz를 바로 프로그램 내부에 읽어 들인 모습을 나타낸다.

침입 데이터 모델

```
Read 494021 rows.
   duration protocol_type   ...    dst_host_srv_rerror_rate  outcome
0         0           tcp   ...                         0.0  normal.
1         0           tcp   ...                         0.0  normal.
2         0           tcp   ...                         0.0  normal.
3         0           tcp   ...                         0.0  normal.
4         0           tcp   ...                         0.0  normal.
5         0           tcp   ...                         0.0  normal.
6         0           tcp   ...                         0.0  normal.
7         0           tcp   ...                         0.0  normal.
8         0           tcp   ...                         0.0  normal.
9         0           tcp   ...                         0.0  normal.

[10 rows x 42 columns]
```

시스템에 나타낸 KDD Cup 1999 Dataset의 일부

494021개의 정상 및 공격 유형 데이터 중 10개만 나타냄

데이터의 특징은 **Duration**부터 **Dst_host_srv_rerror_rate** 까지 총 41개

마지막열의 이름인 Outcome은 이 데이터가 분류되어야 할 Label을 나타냄

Outcome의 종류는Normal, Neptune, Smurf를 비롯해서 총 23개

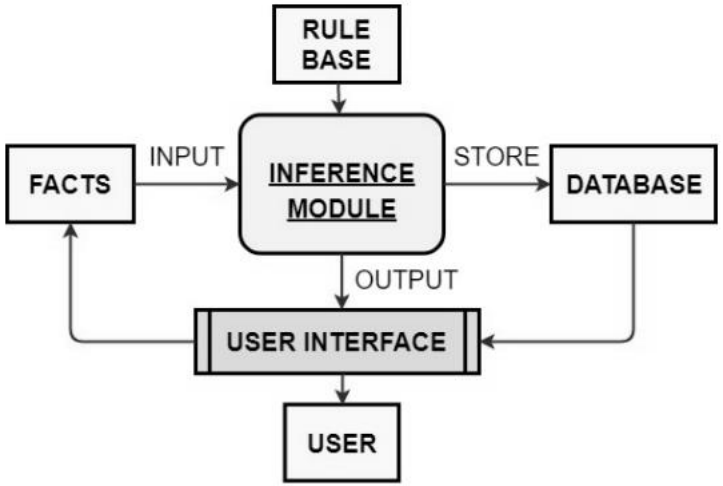
Raw Dataset은 침입 탐지 시스템을 구현할 때 언제든지 적합하도록 다시 가공

침입 탐지 기술 연구 동향

규칙 기반 시스템 (Rule-Based System)

분야의 전문가들이 직접 찾아내고 생성한 규칙을 기반으로 이루어지는 시스템
주로 If-Then 규칙을 이용하여 문제 해결을 진행하며 규칙을 시스템이 자동으로 추론하는 규칙 기반 기계학습
일반적으로 규칙 기반 시스템이 아닌 것으로 간주

Inference Module 실제 데이터의 정보를 이용하여 규칙을 만드는 과정



규칙 기반 시스템의 구조

전문가들이 규칙을 생성하는 과정을 모두 거치면 Database에 저장

⬇

침입 탐지 시스템의 경우
기존 공격 데이터에서 찾아낸 규칙이 들어오는 데이터의 규칙과 일치

⬇

공격 분류

규칙 기반 시스템 (Rule-Based System)

규칙 기반 전문가 시스템(Rule-Based Expert System)을 활용한 DIDS(Distributed Intrusion Detection System) 침입 탐지 모델을 제시

DIDS에서는 전문가 시스템을 활용하면서 전반적인 규칙들은 바뀌지 않지만, 각 규칙에 관계된 Rule Value(RV)라는 변수를 변경하도록 설정

Rule Value는 어떤 규칙이 침입 탐지에 있어서 얼마나 유용한 지를 나타내는 지표로써 사용

생성된 DIDS 모델은 기존 침입 탐지 시스템과는 달리 LAN의 크기와 수가 커질수록
네트워크 환경에서 생성된 침입 탐지 시스템이 영향을 받는 것에 대해 고려

단일 호스트 침입 탐지 시스템의 단점을 개선하기 위해 LAN을 통해 연결된 다중 호스트들로 목표 환경을 일반화할 수 있는 연구 결과 보여줌

ANN (Artificial Neural Network)

ANN에서 필요로 하는 함수적인 필요조건들과 진행과정을 설정

⬇

C++에서 사용되는 오픈 소스 라이브러리인 CBackProp 를 사용하여 학습과 평가를 진행하는 연구

ANN으로 학습하고 평가하기 위 한 데이터 셋으로 kddcup.data_10_percent를 사용하였으며,
침입 탐지를 위해 공격과 정상을 분류하는 이진 분류(Binary Classification) 기법을 이용

1

2개의 공격 유형만 포함한 데이터를
정상 유형 데이터와 같이 학습

평가 시 더 많은 공격 유형을 포함한 데이터로 평가

2

평가 진행 시, 전체 데이터 셋을 사용

3

최적의 특징을 찾아 학습시키는 방법 사용

**향후 침입 탐지 모델이 향상될 수 있도록 영향을 미치는 관련 파라미터들을
더 잘 응용할 수 있도록 하는 연구를 보여줌**

SVM (Support Vector Machine)

데이터의 일부만 분류에 사용하는 기법

데이터의 일부는 데이터를 분류하는 결정적인 분류자 즉, 초평면(Hyperplane)에 가장 근접하는 데이터

NN기법과 SVM기법을 모두 이용해서 침입 탐지 모델을 생성

Dataset 안에 들어있는 특징 값들의 세부사항과 그 특징 값들이 범주형 데이터인지 또는 연속형 데이터인지를 분석

SVM 기법을 적용시키기 위해서 정규화 등에 필요한 하이퍼 파라미터들을 결정

SVM 기법은 이진 분류로만 가능하지만 학습시간은 적게 듬

특이한 점은 **기본 특징 값들이 보안이벤트 분류에 적합하지 않은 값들이라고 보고, 신규 특징 값 추가한 점**

DT (Decision Tree)

Decision Tree는 나무에서 자라는 나뭇가지처럼 의사결정을 해 나가도록 학습하며 분류하는 기법

Decision Tree 기법 안에는 **Best Frist Tree, C4.5 Tree, ID3** 등 다양한 방식의 Decision Tree가 존재

1) OCNL(Oak Ridge National Laboratories)라는 기관에서 만든 데이터 셋을 가지고 학습 진행

2) OCNL Dataset은 KDDCUP'99 Dataset과 비슷하게 크게 Dos, R2L, U2R, Probing 으로 이루어진 공격 유형 데이터 소유

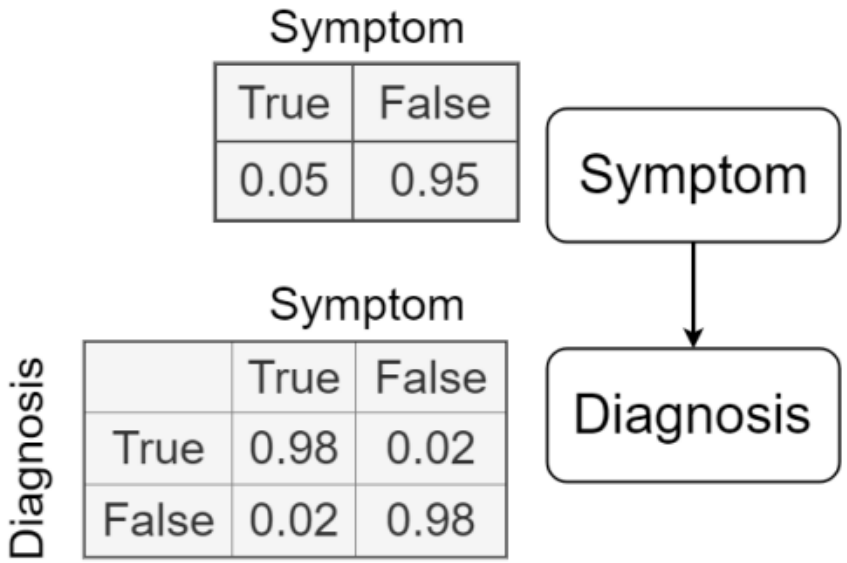
3) Dataset의 비율을 변화시켜가며 학습하는 과정을 거쳐 SVM 기법보다 더 나은 정확도를 보여주는 모델을 생성

DT 기법을 적용하는 과정에서 분류되어지는 방식을 시각화할 시 , 사람이 쉽게 식별할 수 있을 정도의 규칙 적용

→ Decision Tree에서 적용되는 규칙이 규칙 기반 시스템에 공유될 수 있고
해당 규칙을 규칙 기반 침입 탐지 시스템을 구현하는 과정에 참고할 수 있음

BN (Bayesian Network)

랜덤 변수(Random Variable)간 에 DAG(Directed Acyclic Graph)를 만들어 확률 그래프 모델을 만드는 기법



Bayesian Network의 그래프 모델

BN에서 증상(Diagnosis)와 진단(Symptom) 변수 이용하여
' **증상이 있다면 의사가 진단을 옳게 내릴 확률이 몇인가?** '를
구할 수 있는 결합 확률 함수를 그래프 모델로 나타낸 것

BN의 구조

- **노드** Node — 랜덤 변수를 나타냄
- **엣지** Edge — 노드 간의 의존성을 나타냄

더 많은 변수들을 가지고 실제 문제에 적용하여 공격과 정상을 분류하는
BN 기반 침입 탐지 모델 구현

BN을 적용시키는 과정에서 KDDCUP'99 Dataset을 정제한 NSL-KDD Dataset을 사용
NSL-KDD에 있는 특징들을 다 사용하지 않고 특징 선택 기법을 이용하여, 특징 수 감소

BN 기반 분류기(Classifier)를 개선시키기 위해 K2 Search Algorithm을 적용하여 분류

Naive Bayes, C4.5 등을 이용한 침입 탐지 모델과 성능 비교를 하여 우수 성능 보임

RF (Random Forest)

데이터 셋에서 랜덤으로 추출한 데이터를 가지고 만든 여러 개의 Decision Tree를 이용하여
과적합 (Overfitting)을 줄이고 예측하는 성능을 높인 기법

RF모델 생성하기 위해서 두 가지 파라미터 고려

1. Mtry로 각각의 트리 노드에서 이루어지는 소분류를 고려하기 위해 랜덤으로 선택된 특징의 수
2. RF모델 내에서 사용 된 트리의 개수를 나타내고, **Ntree** 라고 표현

이 두가지 파라미터를 이용해 N개의 트리를 가지는 RF 기법 기반 침입 모델을 생성한 뒤
어떤 트리가 가장 잘 분류하는 지를 테스트 데이터를 이용해 검증하여 최종적인 모델을 결정

분석 및 평가

SVM, NN, DT를 이용하여 침입 탐지 모델을 생성 및 비교, 평가

분석방법

침입 탐지 모델은 공격(침입)을 탐지하기 위한 것이므로 기본적으로 공격 유형의 데이터가 분류될 때 이진 분류로 분류

이진 분류를 하게 되면 원래 데이터 셋의 라벨(Label)과 모델에 넣었을 때 분류되는 예측 값(Predicted Value)을 비교하여 올바르게 분류되었는지 아닌지를 판별 해야함

TP(True Positive), TN(True Negative), FP(False Positive), FN(False Negative)을 행렬로 간단하게 표현

혼동행렬

Predicted Value

Label (Target Value)	**Positive**(=1)	**Negative**(=0)
Positive(=1)	True **Positive**	False **Negative**
Negative(=0)	False **Positive**	True **Negative**

혼동 행렬에서 True와 False를 결정하는 기준

: Predicted Value와 라벨(Label)의 값이 같은지 아닌지의 여부

값이 같다면 True | 값이 다르다면 False

Positive와 Negative는 Predicted Value의 값에 의해서 결정됨

분석방법

Predicted Value 가 Positive의 값을 가지고 그 값이 Label과 똑같다면

혼동 행렬에서의 값 = True Positive

$$\text{탐지율}(Detection\ Rate) = \frac{TP}{TP + TN} \quad (1)$$

$$\text{정확도}(Accuracy) = \frac{TP + TN}{TP + FP + TN + FN} \quad (2)$$

$$\text{오경보율}(False\ Alarm\ Rate) = \frac{FP}{FP + TN} \quad (3)$$

정확도 전체 데이터 중에서 정상유형이나 공격유형으로 올바르게 잘 분류한 데이터의 비율

탐지율 원래 Positive로 분류되어야 하는 것 중에서 Predicted Value가 얼마나 잘 분류된 지를 나타낸 척도

오경보율 원래 Negative로 분류되어야 하는 데이터 중 Predicted Value가 얼마나 잘못 분류되었는지를 나타낸 척도

성능평가

사용 프로그래밍 언어는 Python

전처리(Preprocessing)을 위해 Keras, Numpy 라이브러리 사용

기계학습 모델 생성을 위해서는 Tensorflow와 Scikit-Learn 라이브러리 사용

DT 기반 그리고 SVM기반 침입 탐지 모델을 구현하는 과정에서 Scikit-Learn 라이브러리에는 DT 기법과 SVM 기법을 이용할 수 있는 함수가 마련되어 있으므로 Scikit-Learn을 활용

DT기반 침입탐지 모델에서 노드가 뻗어 나가는 최대 길이 즉 Depth를 25로 설정

SVM기반 침입탐지 모델에서는 OVR(One Versus Rest)방식과 RBF(Radial Basis Function)을 이용하여 모델을 구현

NN 기법은 Tensorflow 라이브러리를 이용하여 Layer의 개수, 각 Layer의 노드의 개수, 오차함수까지 자율적으로 조정할 수 있는 침입 탐지 모델을 구현

KDDCUP'99 Dataset을 이용해 학습을 진행하여 정확도와 오경보율 그리고 탐지율을 측정

성능평가

성능 평가에 사용한 함수 종류와 구체적인 파라미터 값을 정리

DT	NN	SVM
Sk-learn	Sk-learn	Tensorflow
Depth:25	Iteration:4000	Iteration:4000
-	Gradient Descent	Decision Function : OvR
-	오차 함수 : MSE	커널 함수 : RBF
-	Input Layer Node: 123 Output Layer Node: 23 Learning Late : 0.05	Tolerance : 0.0001

Dataset을 기계학습 기법에 이용하기 위해 String형태로

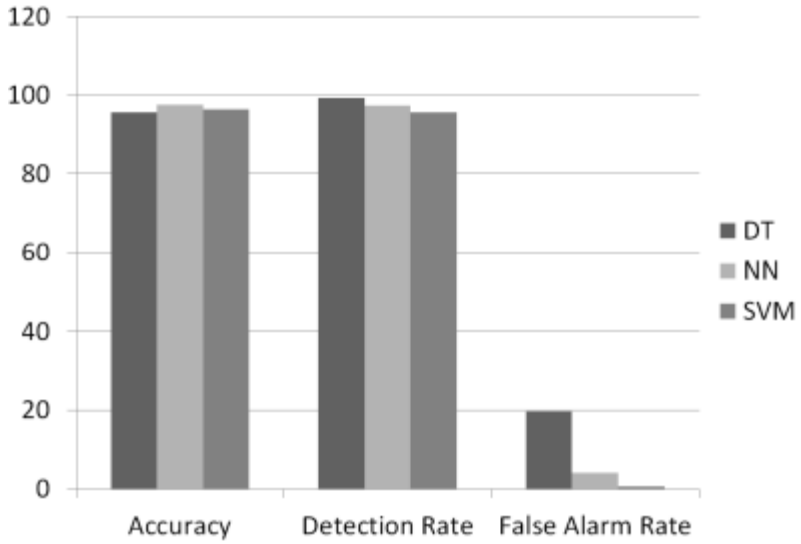
이루어진 특징 값들은 One-Hot Encoding을 통해서 이진수로 전환

41 개의 특징에 존재하는 값의 범위가

각각 다르므로 더 효율적인 학습을 위해 정규화 과정 실시

성능평가

성능 분석



정확도와 탐지율에서는 크게 차이를 보이는 부분은 없었지만,
오경보율는 DT기반 침입 탐지 모델이 다른 모델보다 높음

오경보율은 정상 유형이 얼마나 공격 유형으로 잘못 분류되었는지를 나타내는 척도
다른 두 개의 침입 탐지 모델보다 DT기반 침입 탐지 모델이 신뢰도가 떨어짐

탐지율과 오경보율이 비례관계
공격 유형이 공격 유형으로 올바르게 분류될수록 정상 유형이 공격 유형으로 잘못 분류되는 경향을 보임
한 기계학습 기법 내부에서 파라미터를 변화시켜주어도 나타나는 모습 보임

탐지율과 오경보율 사이의 관계를 잘 분석하고
탐지율은 높이면서 오경보율은 낮추는 방향으로 모델을 개선하는 것이 중요

결론

기계학습 기반 침입 탐지 모델은 공격 미탐지로 인한 위험성이 크다는 점에서 다른 분야의 분류 문제보다 훨씬 더 중요하고,
매우 철저하게 검증할 필요가 있으며 더 많은 연구가 필요하다

본 논문에서는 침입 탐지 기법을 적용한 다양한 논문들을 살펴보고, 논문에 적용된 기법들의 성능 분석 결과를 살펴봄으로써
향후 기계학습 기반 침입 탐지 모델을 구현 및 개선할 수 있다

또한 여러 기계학습 기법 중 SVM, DT, NN 기반 침입 탐지 모델을 직접 구현하고 성능을 비교함으로써
모델의 특징과 성능 분석 방법에 대한 새로운 상관관계를 도출했다