Python 기반 AI 연구 소개

성균관대학교
소프트웨어대학
정윤경
aimecca@skku.edu

Outline

- 소개
- AI 기반 Data mining 과정 개요
- Predator Detection
- ML기반의 네트워크 이상행위 탐지와 관련된 기술 개요
- Sentiment Analysis of Movie Script

소개

- 연구 분야: Narrative Generation, Game AI, Data mining, NLP
- 성균관대학교 소프트웨어대학 조교수 (2015-현재)
- IT University of Copenhagen, Denmark (post-doc, 2010-2014)
- 삼성전자 종합기술원 (2008-2010)
- LG전자 (1998-2001)
- Ph.D in CS at North Carolina State University, USA
- 성균관대학교 정보공학과 학/석사

Data mining Procedure

Data Mining Research Process

01 **Data Collection**
- Web Crawling
- Repository (e.g., Kaggle)
- Collection

02 **Data Understanding**
- Domain/data Understanding
- Statistical Analysis
- Feature Selection

03 **Preprocessing**
- Missing information, Filtering bad examples
- **normalization**
- **Train/Test Preparation**
  - **Train data not in the Test data**
  - **Class distribution**

04 **Train**
- Model Selection
- Meta parameter Tuning
- Enough training until performance converges

05 **Test**
- Evaluation Metrics
- Baseline

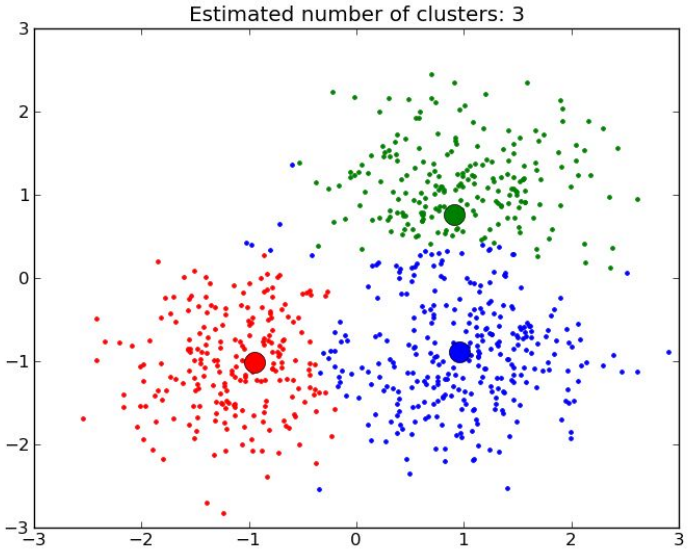
기계 학습 (Machine Learning)

- 기계 학습 알고리즘이 하는 일은 몇 개의 범주 (category)로 데이터가 분류되어 있을 때, 주어진 데이터를 해당 클래스에 맞도록 분류할 수 있는 기준 (분류기)을 찾는 기술
- 기계 학습은 학습 + 추론(분류) 을 수행
- 활용: 스팸 이메일 필터링, 신용카드 비정상적 거래, 음성 인식, 필기체 인식, 정보 검색, 오피니언 마이닝 등 패턴 인식이 문제 해결에 핵심인 분야

학습 알고리즘	설명
결정트리	특정 사례가 어떤 클래스에 속하는지 분류하는 과정을 트리의 형태로 표현. 훈련 데이터가 주어지면 트리를 자동으로 생성. 유일하게 학습 과정의 지식을 도출할 수 있는 알고리즘
SVM	다차원 공간에서 서로 다른 클래스를 분류하는 support vector를 주어진 데이터로부터 결정하는 알고리즘. 가장 효율이 높은 것으로 알려짐.
K-means clustering	다차원 공간에서 특정 사례를 표현했을때, 입력된 사례와 가장 가까운 mean값이 해당하는 클래스로 예측
베이지안 네트워크	지식과 추론을 조건부 확률 네트워크로 표현
신경망	이진 결과를 출력하는 노드의 집합으로 입력과 출력 정보를 표현하고, 입력층과 출력층간 연결을 담당하는 은닉층으로 설계된 네트워크 구조

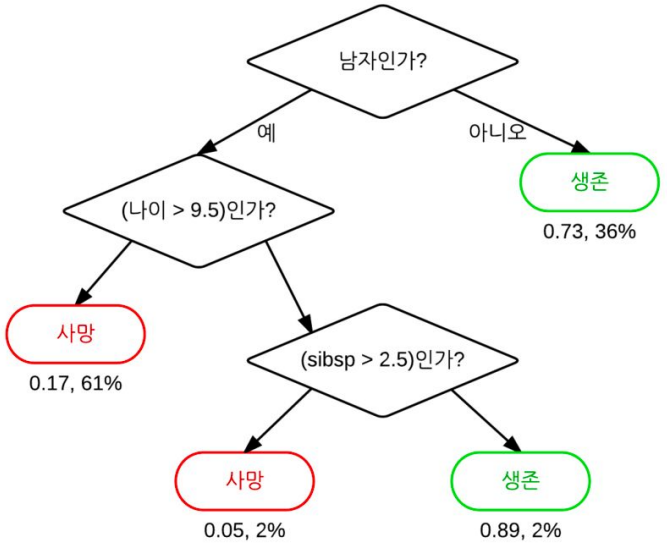
Unsupervised Learning

- 학습 데이터에서 label 정보가 없음
- 새로운 종류의 공격을 탐지하는 데에 활
- Clustering algorithm
- KNN, K-means



Decision Tree

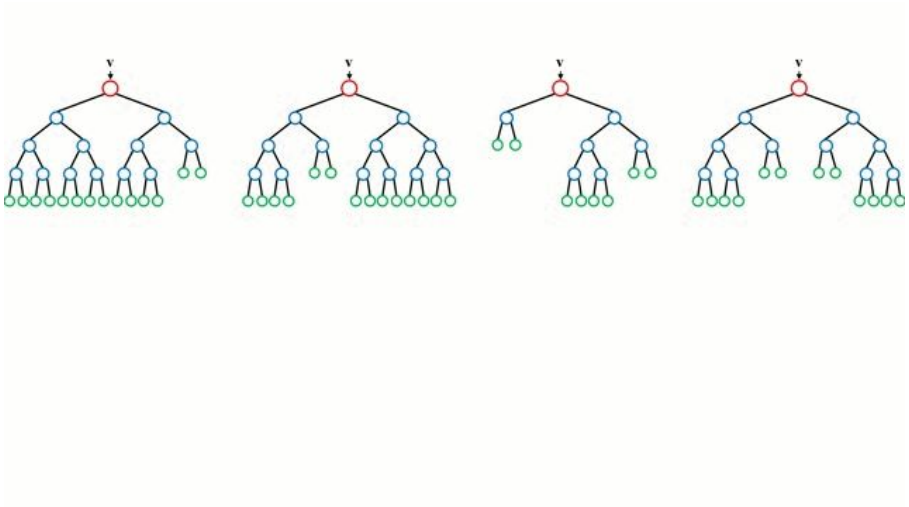
- 각 노드는 feature를 의미
- Branch마다 속성에 대한 값
- Information Gain이 최대가 되도록 자식 노드를 생성



타이타닉호 탑승객의 생존 여부를 나타내는 결정 트리. (“sibsp”는 탑승한 배우자와 자녀의 수를 의미한다.) 잎 아래의 숫자는 각각 생존 확률과 탑승객이 그 잎에 해당될 확률을 의미

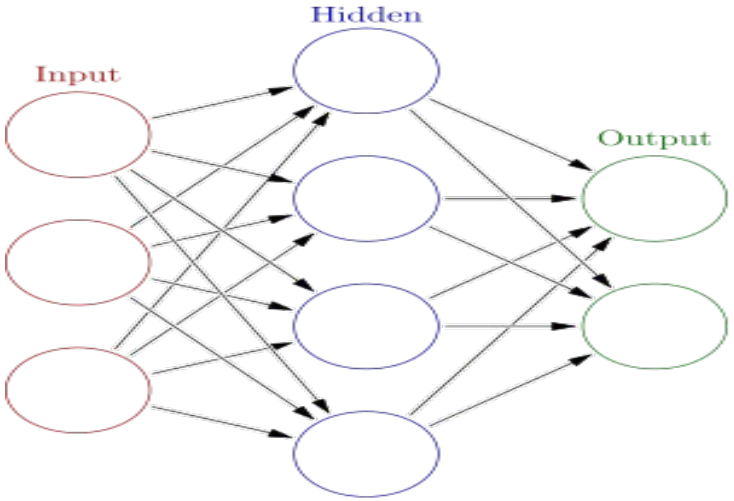
Random Forest

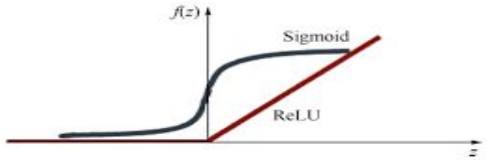
- 훈련 데이터를 분할하여 여러 트리를 학습
- 주어진 데이터의 결과를 평균 혹은 투표로 결정



Artificial Neural Network

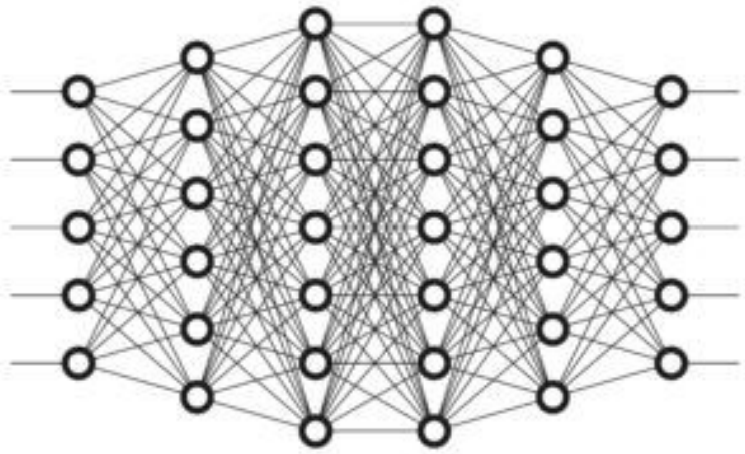
- 사람의 뉴런 동작 방식을 모사
- 노드는 입력받은 신호의 가중치가 반영된 총합을 구하고, 그 합이 임계치 이상인 경우 1을, 아닌 경우 0을 출력. 1을 출력하는 것을 활성화(activation)
- 입력/은닉/출력층 노드간의 연결 강도를 나타내는 가중치를 최종 출력 결과가 좋도록 최적 값을 찾는 것이 학습
- 활성화 함수
  - Sigmoid
  - ReLU(Rectified Linear Unit)





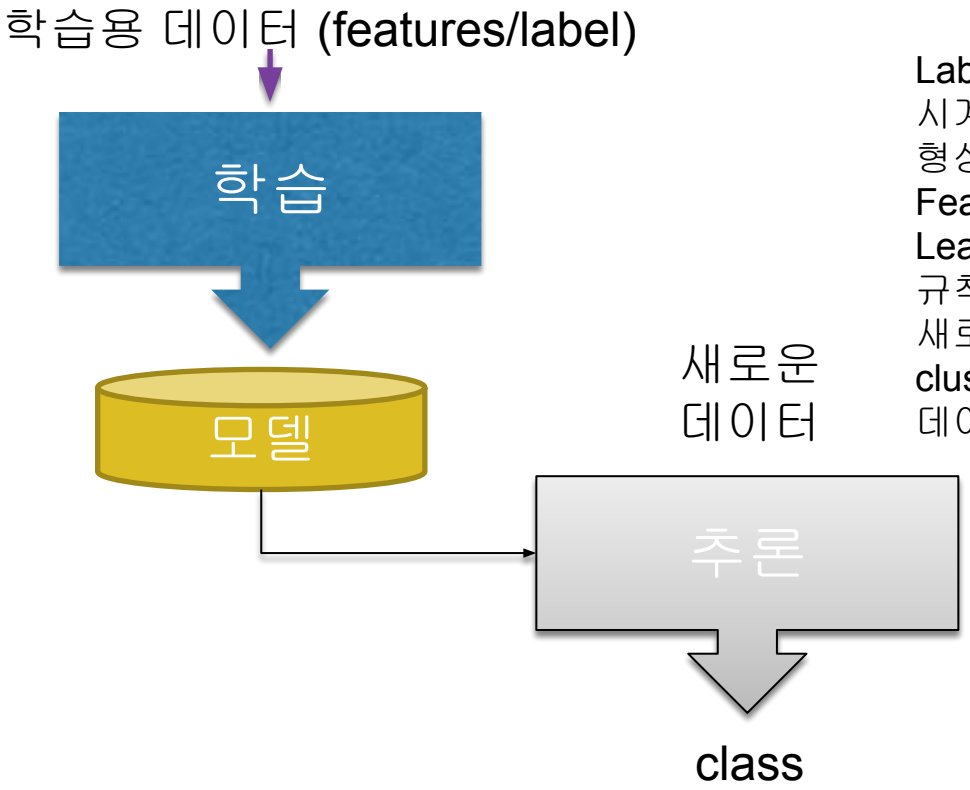
Deep Learning

- 신경망의 은닉층수를 높인 확장된 기술
- 기존 단일 컴퓨터의 파워로 계산할 수 없던 것이 GPU를 병렬로 활용하면서 가능해짐
- 특징 선택이 자동화 되면서 문제 도메인이 여 문제를 풀 수 있게 됨



딥러닝 기술	설명	오픈소스
DNN (deep neural network)	은닉층이 깊고 속성 선택 단계가 없는 신경망 구조	Keras Theano Caffe TensorFlow Microsoft CNTK Torch Deeplearning4j
CNN (convolutional neural network)	입력의 차원 정보를 유지하는 딥러닝 네트워크 구조로 주로 이미지 인식에 활용	
RNN (recurrent neural network)	은닉층이 과거의 상태를 저장하여 시퀀스 및 시계열 데이터에 활용되는 딥러닝 네트워크 구조	
LSTM(Long Short Term Memory)	오래된 과거의 정보를 저장할 수 있는 구조	
Autoencoder	입력 정보와 동일한 정보를 출력하는 단일 은닉층 신경망 구조. 즉, 입력을 복원하는 기능을 수행	
GANs(Generative Adversarial Networks)	주어진 훈련 데이터에서 유사한 가짜 데이터를 만들어 내는 생성기와, 진짜 데이터와 가짜 데이터를 식별하는 분류기를 학습	

Model 선택시 고려 사항



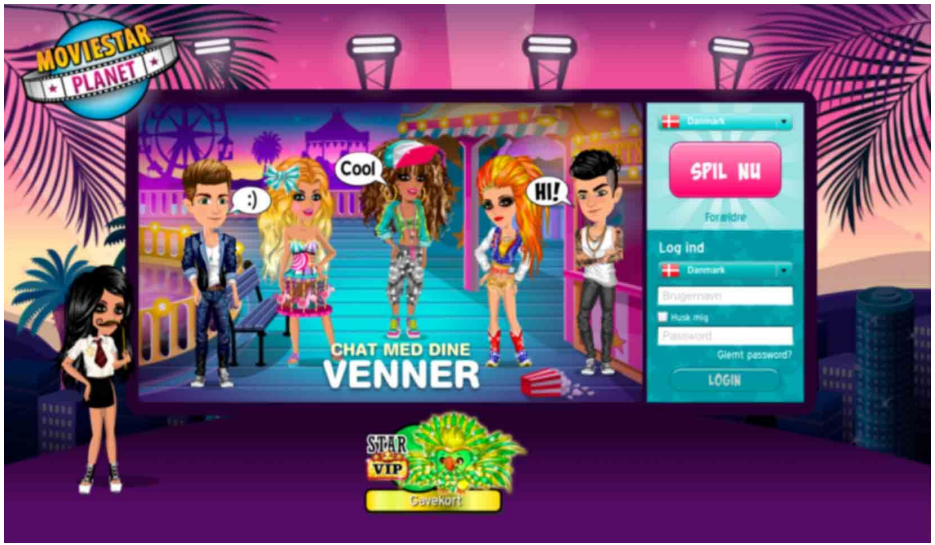
Label이 있는지? => 지도/비지도 학습
시계열 데이터인지? => 신경망/RNN/LSTM
형상 데이터인지?=>CNN
Feature를 추출해야 하는지? => 신경망/Deep Learning
규칙을 도출하고 싶은지?=>결정 트리
새로운 유형의 공격을 탐지하고 싶은지? => clustering
데이터가 큰지?=> splunk등 빅 데이터 프레임워크

Detecting Predatory Behavior in Game Chats

Cheong, Y.-G., Jensen, A.K.,Gudnadottir, E.R., Bae., B.-C., Togelius, J.
IEEE Transactions on Computational Intelligence and AI in Games, Volume:7, no: 3, pp. 220 - 232, 10 September 2015

MovieStarPlanet

- A social online game where each player controls an actor/actress character
- Children aged between 8 and 15 years are the target user group
- Aims to raise a movie start and produce a movie in the game collaborating with other game users



https://moviestarplanet.com/
Popular in UK and USA

Research Question

Can text classification algorithms be used to label sexual predators in the specific context of MovieStarPlanet game, using chatlogs from each user during a certain time frame?

- The context of an online game for children
- Our definition of a sexual predator
  - Anyone who initiates sexually suggestive language (i.e. ”Let’s have sex” or ”What does your underwear look like?”).
  - Anyone who welcomes this type of language, and responds with similar language.
  - Anyone who tries to gain physical access to other users of the game. (i.e. ”Let’s meet in real life”).

Dataset from MSP

- Predator Data
  - 1 file per predator (over 600 total), outgoing and incoming.
  - Divided into categories 0, 1, 3, 7, 99999.
  - Type of offence unknown. After sorting according to type of offence, 58 sexual predators found.
  - Timespans can be short or very long (up to three months).

- Non-predator Data
  - 65000 rows.
  - Containing 15 minutes of gameplay across entire UK site
  - Essentially unlabeled.

Chat Example

Sexual harassment chat

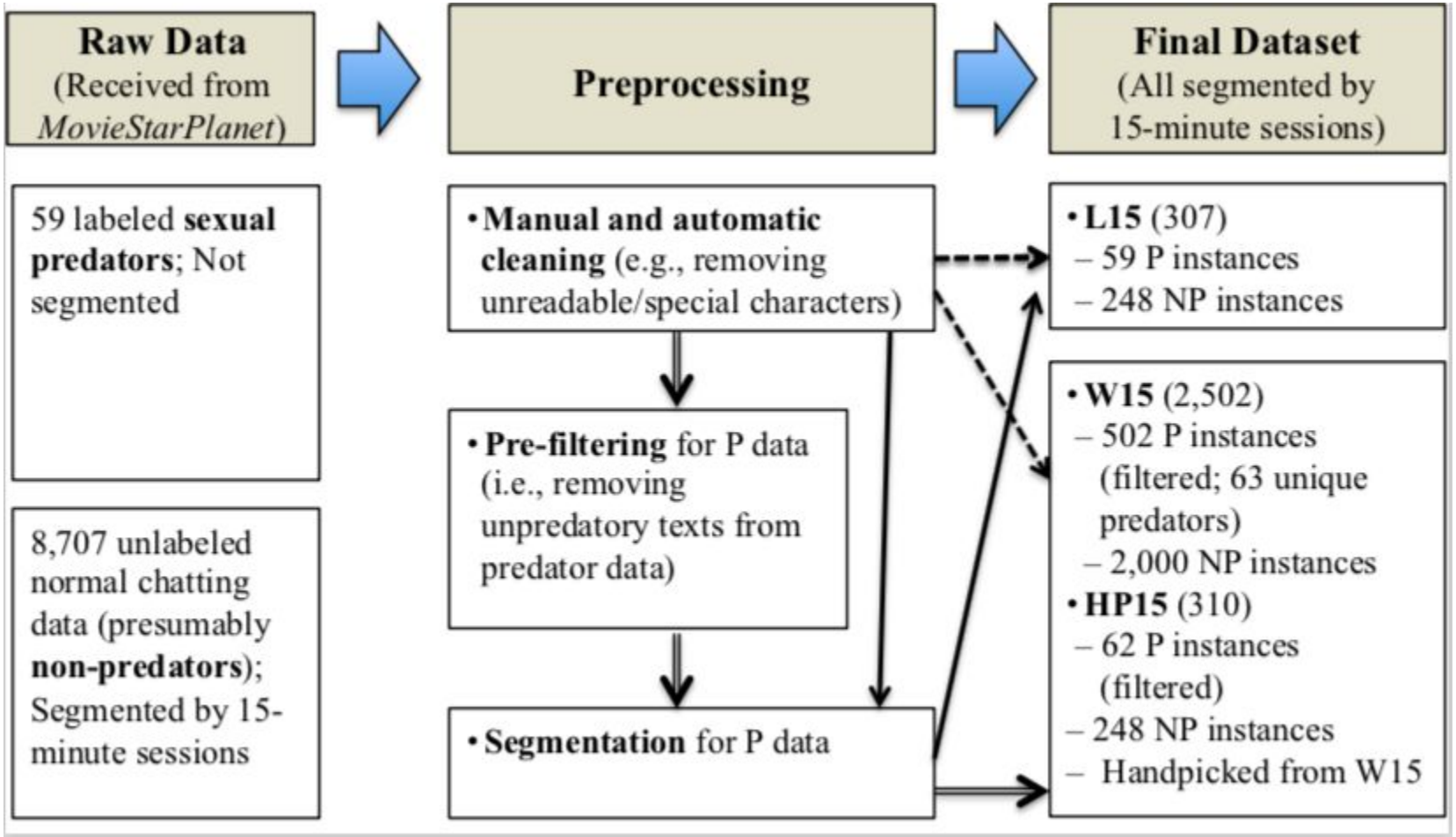
A.12662280	A.13403615	-1	2-2-13 11:15	285381198	S	NULL	
A.12662280	A.13403615	-1	2-2-13 11:15	285381198	E	NULL	
A.12662280	A.13403615	-1	2-2-13 11:15	285381198	X	NULL	
A.12662280	A.13403615	-1	2-2-13 11:15	285381198	WITH A 5 YEAR OLD	NULL	
A.12662280	A.12375332	-1	2-2-13 11:15	285381198	U WANNA MEET ME IN RL	NULL	
A.12662280	A.13403615	-1	2-2-13 11:16	285381198	HAVE IT		99999
A.12662280	A.13403615	-1	2-2-13 11:16	285381198	WOULD U WANNA BABY IF WE MET	NULL	
A.12662280	A.13403615	-1	2-2-13 11:17	285381198	WOULD U PEE ON ME IF WE MET	NULL	

Regular chat

A.11413710	A.14798585	-1	4-3-13 12:03	366252878	Can u check my playlists
A.14958259	Dressup Game	-1,6E+009	4-3-13 12:03	1,26E+009	i'll give u a auto
A.14955887	User Room	14835309	4-3-13 12:03	250741342	i been here 4 ages but the brides late lol
A.14908675	Park	339697135	4-3-13 12:03	-1,1E+009	wolfy roll
A.5470223	A.7533606	-1	4-3-13 12:03	583794000	
A.2024018	Mall	645864018	4-3-13 12:03	-7,5E+008	u blocked me >.>
A.14821652	A.13640251	-1	4-3-13 12:03	1,27E+009	in msp
A.14902284	User Room	14958286	4-3-13 12:03	203584606	i dont

MSP’s efforts to provide a safe and secure environment for children

- (Real human) Moderators and automated systems monitor the chat to locate offensive language
- Keep list of block/alert words
- Provide User Interface to instantly report perpetrators


Raw Data
(Received from *MovieStarPlanet*)
Preprocessing
Final Dataset
(All segmented by 15-minute sessions)
59 labeled **sexual predators**; Not segmented
8,707 unlabeled normal chatting data (presumably **non-predators**); Segmented by 15-minute sessions
• **Manual and automatic cleaning** (e.g., removing unreadable/special characters)
• **Pre-filtering** for P data (i.e., removing unpredatory texts from predator data)
• **Segmentation** for P data
• **L15** (307)
– 59 P instances
– 248 NP instances
• **W15** (2,502)
– 502 P instances (filtered; 63 unique predators)
– 2,000 NP instances
• **HP15** (310)
– 62 P instances (filtered)
– 248 NP instances
– Handpicked from W15

Dataset for train/test

	Number of Users	*Number of lines*	*Unique words*	*Misspelled words*
W15-P	63	9,138	3,193	1,350
W15-NP	2,000	20,699	7,989	4,090
L15-P	59	1,063	926	249
L15-NP	248	3,388	2,532	882
HP15-P	62	1,220	1,086	339
HP15-NP	248	3,388	2,532	882

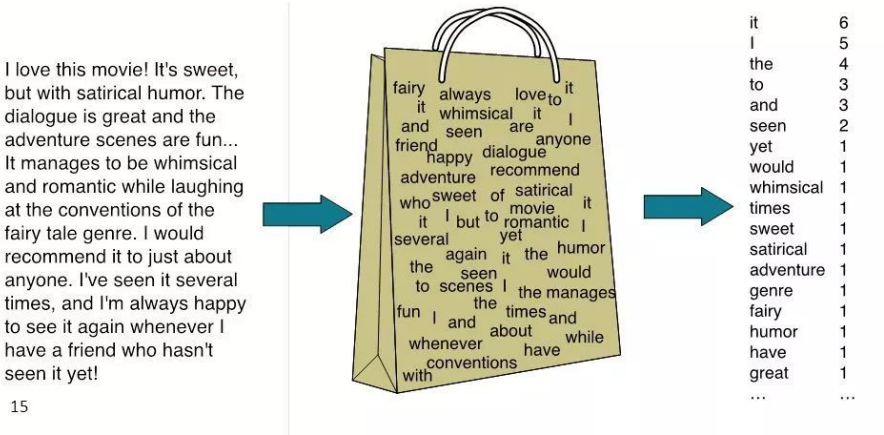
Features

- Bag of Words (BOW)
- Sentiment Features
  - Emoticons: Positive, Neutral, Negative
  - Sentiments: Positive, Negative
- Rule Breaking Features
  - Linguistic Features: One letter lines, One word line, Lines, Spaces, Non letter word, Consecutiveletters, Misspellings
  - Blacklist and alert words

Feature type	Sub-category	Size of the feature vector
BoW	No sub-category	dynamically generated
Sentiment	Emoticons	3
	Sentiment scores	2
Rule Breaking	Behavioural Features	7
	Blacklist/Alert list features	2

BOW (Bag-of-words)

- A sentence or a document is represented as the bag of its words, disregarding grammar and word order but keeping multiplicity
- Used for document classification where the (frequency of) occurrence of each word is used as a feature for training a classifier



(1) John likes to watch movies. Mary likes movies too.
(2) John also likes to watch football games.

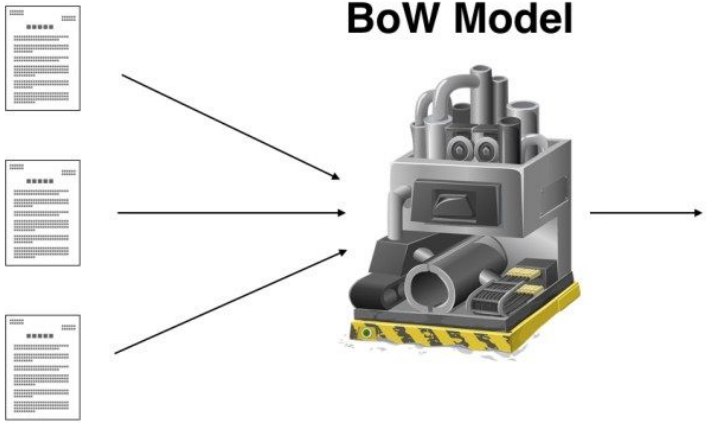
(1) [1, 2, 1, 1, 2, 1, 1, 0, 0, 0]
(2) [1, 1, 1, 1, 0, 0, 0, 1, 1, 1]

BoW1 = {"John":1,"likes":2,"to":1,"watch":1,"movies":2,"Mary":1,"too":1};
BoW2 = {"John":1,"also":1,"likes":1,"to":1,"watch":1,"football":1,"games":1};

1. I love dogs.

2. I hate dogs and knitting.

3. Knitting is my hobby and my passion.



	I	love	dogs	hate	and	knitting	is	my	hobby	passion
Doc 1	1	1	1							
Doc 2	1		1	1	1	1				
Doc 3					1	1	1	2	1	1

Sentiment Features

- emoticon scores
  - positive [e.g., :-), :*], neutral (e.g., :-P), and negative [e.g., :-(, 8(, etc.]
  - extracted from the data using regular expressions
- sentiment scores
  - positive and negative sentiment scores using AFINN-111 wordlist
  - AFINN-111 sentiment wordlist was specifically created for microblogging (i.e., twitter) and was more effective on this type of the Internet content than the standard but older ANEW list

Rule Breaking Features

Someone give me an auto and if u do i will give u one back! ;p
want to swap accounts
f* off other wise i will brske ur a*
ITS SPRING SUMMER HOLIDAYS BEACH ICE-CREAM AND I AM FREE NO SCHOOL FOR 2 WEEKS !!!!!!!
Nothing whats wrong with UR HAIR ITS WEARD

who wants to have *** with me
can i fiddle with ur body?
what is your email
it with me i can feel it now uhh uhh uhh ohh harder HARDER
if u want to be my gf tell me where u live then

Rule Breaking Features

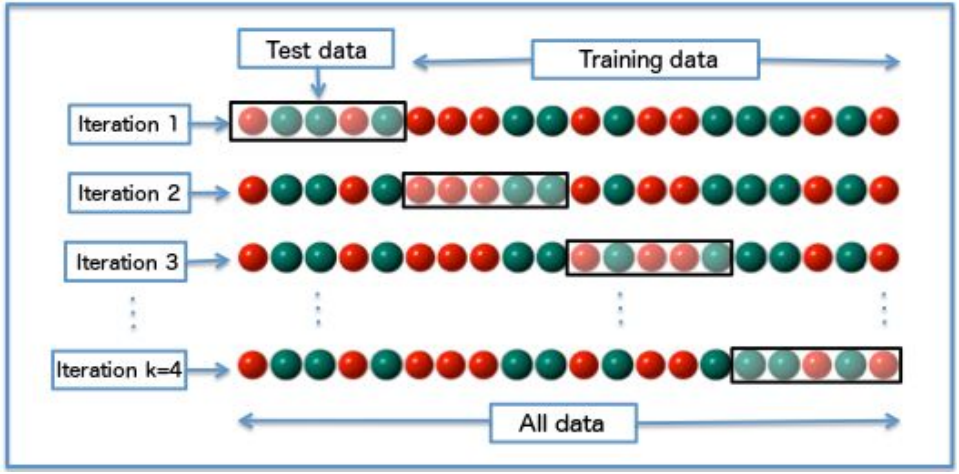
- One letter lines (LL)—user types a blacklist word, one letter per line
- One word lines (WL)—user types a forbidden phrase by typing one word at a time.
- Lines (NL)—The number of lines itself could possibly indicate suspicious behavior
- Spaces (SP)—user types blacklist words with spaces in between the letters in a word (e.g., “s e x”).
- Nonletter words (NLW)—user types blacklist words by typing symbols or numbers in place of letters, for instance “s*x.” This feature records a count of the words which contain symbols and/or numbers in addition to letters.
- Consecutive identical letters (CL)—A user types many consecutive identical letters inside a word, for example “seeeeex.”
- Misspellings (MS)—A user types misspelled words to avoid being caught

Evaluation

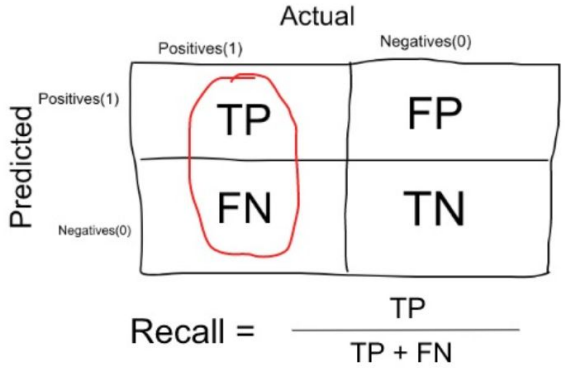
- R + Weka package
- Initial evaluation: Random Cross Validation with Naive Bayes (NB), Decision Tree (DT),Neural Network (NN), Linear Regression (LR), k-Nearest Neighbour (k-NN), Support Vector Machine (SVM)
  - All features
  - Feature Selection by Feature Set
- Second evaluation: separate the test so that the id in the test set does not appear in the training set and ran five-fold cross validation on the training set
- Testing on unlabeled data

Cross validation

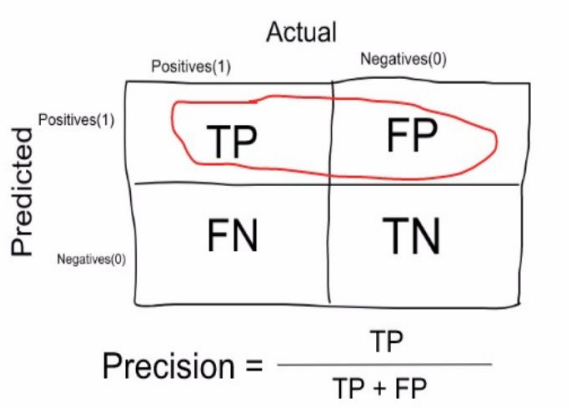
- Partition the data into k subset
- Choose one subset for each validation
- Take a caution when the data contains multiple instances from a single source



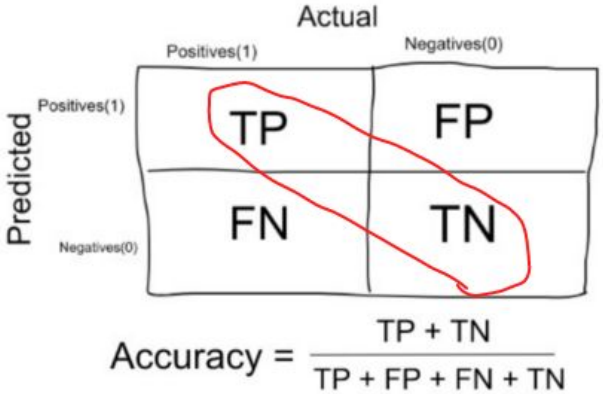
Evaluation Metrics



$$\text{Recall} = \frac{TP}{TP + FN}$$



$$\text{Precision} = \frac{TP}{TP + FP}$$



$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$

F2는 recall을 precision보다 2배 더 중요하게 고려.
F0.5는 precision를 recall을보다 2배 더 중요하게 고려

Results

Detection with All Features

Data	Alg.	Acc.	$F1$	$F0.5$	P	R
HP15	NB	.86	.67	.65	.64	.69
	DT	.86	.57	.68	.78	.45
	LR	.76	.60	.50	.45	**.89**
	kNN	.84	.34	.53	**.87**	.21
	SVM	**.89**	**.71**	**.73**	.75	.68
	MLP	.89	.70	**.73**	.66	.75
W15	NB	.79(2.20)	.57(.04)	.51(.04)	.48(.04)	.72(.07)
	DT	.91(3.99)	.74(.16)	.78(.06)	.80(.05)	**.73(.23)**
	LR	.89(3.11)	.70(.08)	.74(.09)	.76(.09)	.66(.10)
	kNN	.85(1.72)	.49(.14)	.62(.07)	.71(.06)	.40(.17)
	SVM	.92(2.51)	.77(.09)	.85(.06)	**.91(.05)**	.68(.12)
	MLP	**.93**(1.74)	**.78(.06)**	**.86(.04)**	**.91(.04)**	.70(.11)
L15	NB	.80	.46	.47	.48	.44
	DT	**.83**	**.52**	**.55**	**.57**	.48
	LR	.72	.49	.41	.38	.70
	kNN	.26	.34	.24	.20	**.97**
	SVM	.81	.47	.49	.50	.44
	MLP	.81	.40	.46	.50	.34

Results

Detection with Rule-Breaking Features

Data	Alg.	Acc.	$F1$	$F0.5$	P	R
HP15	NB	.86	.56	.65	.74	.45
	DT	.86	.57	.68	.78	.45
	LR	.86	.57	.67	.76	.45
	kNN	.82	.40	.50	.59	.31
	SVM	.84	.38	.56	**.83**	.24
	MLP	**.87**	**.63**	**.70**	.77	**.53**
W15	NB	.86(3.28)	.62(.13)	.65(.06)	.65(.05)	.65(.19)
	DT	**.91(2.46)**	.73(.1)	**.80(.04)**	**.84(.04)**	.66(.16)
	LR	.90(3.04)	.72(.13)	.77(.03)	.80(.02)	.67(.20)
	kNN	**.91(2.71)**	**.75(.09)**	**.80(.06)**	**.84(.05)**	.68(.14)
	SVM	.90(2.80)	.73(.09)	.79(.04)	.77(.05)	**.69(.15)**
	MLP	**.91(2.15)**	.73(.09)	.79(.06)	.82(.05)	**.69(.18)**

Comparisons with other work

- Dataset Types

  a) predator/victim (victim is underage)

  b) predator/pseudovictim (volunteer posing as child);

  c) predator/pseudovictim (law enforcement officer posing as child)

Participant	**Precision**	**Recall**	**F1**	**F0.5**
Villatoro-Tello et al. [19]	0.98	0.77	0.87	0.93
Parapar and el. [18]	0.94	0.67	0.78	0.87
Morris and Hirst [23]	0.97	0.60	0.75	0.87
Eriksson and Karlgren [25]	0.86	0.89	0.87	0.86
Peersman et al. [16]	0.89	0.60	0.71	0.81
Kontostathis et al. [15]	0.36	0.67	0.47	0.39
Bogdanova et al. [4]	0.03	0.22	0.05	0.03

RESULTS OF THE PAN2012 COMPETITION

The PAN2012 data set consists of chatlogs drawn from various sources in- cluding true positives (predators) from PJ (b type), negative data from IRC logs (chats about generic topics), and false positives from the Omegle repository (consensual chats about sex).
The Perverted Justice Foundation provides files containing full chatlogs of convicted sexual offenders are available for download on perverted-justice.com

Discussions

- Machine Learning algorithms successfully classified sexual predators at the accuracy of 91-93 %, F1 of .78, F0.5 of .86
- Behavioral features were as useful as BoW representation, especially when the data contains less severe predatory language
- sentiment features that we created in this work did not contribute to performance improvement

Feature type	Sub-category	Size of the feature vector
BoW	No sub-category	dynamically generated
Sentiment	Emoticons	3
	Sentiment scores	2
Rule Breaking	Behavioural Features	7
	Blacklist/Alert list features	2

Conclusions

- This study is the first work that demonstrated the feasibility of the text classification approach for detecting predators in a real game chat corpus
- Machine Learning algorithms successfully classified sexual predators at the 91-93 %, F1 of .78, F0.5 of .86
- Behavioral features were as useful as BoW representation, especially when the data contains less severe predatory language
- Limitations with small dataset (59 predators)
- For future work, cleaner and more focused data is necessary

Code

```
def read_lines2(type):
    datnam=type+"dat.txt"
    tarnam=type+"tar.txt"
    arr=[]
    arr2=[]
    with open(datnam, 'rU') as data:
        reader = csv.reader(data)
        y=0
        for row in reader:
            new = [float(x) for x in row if x != '']
            arr.append(new)
    with open(tarnam, 'rU') as data:
        dat=data.readlines()
        for line in dat:
            arr2.append(float(line))
    return {'dat':arr,'tar':arr2}
```

```
import csv
import numpy as np
rec=read_lines2("feb")
rec2=read_lines2("jan")
datset=np.array(rec['dat'])
tarset=np.array(rec['tar'])
testdat=np.array(rec2['dat'])
testtar=np.array(rec2['tar'])

from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
sc.fit(datset)
train_std=sc.transform(datset)
test_std=sc.transform(testdat)
Ml =  RandomForestClassifier(
criterion='entropy',n_estimators=10,n_jobs=2,random_state=
1)
ml.fit(train_std,tarset)
pred=ml.predict(test_std)
```

```
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report

true_labels=testtar
from sklearn import metrics

print(metrics.accuracy_score(true_labels,pred))
print(metrics.confusion_matrix(true_labels,pred))

print('총테스트개수 :%d 오류개수:%d'
%(len(testtar),(testtar!=pred).sum()))
from sklearn.metrics import classification_report

targets=['class1','class2','class3','class4','class5','class6']
print('\n',classification_report(true_labels,pred,target_names=targets))
```

네트워크 이상징후 탐지 모델

클래스 불균형 데이터에 적합한 기계 학습 기반 침입 탐지 시스템. 정윤경, 박기남, 김현주, 김종현, 현상원. 정보보호학회논문지. 27권 6호. 2017년 12월

연구 목표 및 내용

- 정상과 이상 트래픽이 불균형적으로 발생하는 상황에서 기계 학습 기반의 효과적인 침입 탐지 모델 구축
- 클래스 불균형 문제가 있는 경우에도 신뢰성 있는 탐지 성능을 보이는 네트워크 IDS를 개발
- 실제 환경에서 수집한 네트워크 로그 데이터인 Kyoto 2006+ 데이터셋으로부터 정상과 침입 클래스 간 비율이 다른 실험 데이터셋을 구축
- 기계 학습 연구 분야에서 활용되는 성능 비교 척도를 사용하여 6가지 대표적인 기계 학습 알고리즘의 탐지 성능을 분석

Kyoto 2006+ 데이터셋

- 교토 대학교에서 2006년부터 2015년까지 다양한 유형의 허니팟 시스템으로부터 수집된 네트워크 로그
- 24개의 feature
- 14개 feature는 기존 KDD CUP 99 데이터에서 학습에 유의미한 것들을 선정
  - connection duration, service, source bytes, count
- 나머지 10개의 feature는 검증 용도로도 사용될 수 있도록 추가
  - Snort 탐지 여부, 안티 바이러스 탐지 여부, 쉘 코드 탐지 여부, 위협 혹은 정상 세션인지 여부를 나타낸 레이블, 소스 IP 주소, 소스 포트 번호, 목적지 IP 주소, 목적지 포트 번호, 시작 시간, 지속 시간 정보
- 3종류의 클래스
  - 알려진 침입 (-1)
  - 쉘코드 기반의 알려지지 않은 침입 (-2)
  - 정상 서버에 들어오는 패킷은 정상 (1)

데이터

- 데이터셋 확보: 교토대 허니팟 2006+
- 실험용 데이터셋 구축
  - 테스트 데이터: 2013년 1월 3-9일
  - 훈련 데이터: 2013년 2월 19-25일
  - 침입 데이터가 1%에 해당하도록 랜덤 샘플링한 훈련 3세트 추가

Label	**Training (2013.1.3.-9)**	**Test (2013.2.19.-25)**
1 (normal)	303,929 (27.3%)	232,431 (34.9%)
-1 (abnormal)	801,767 (72.0%)	431,722 (64.8%)
-2 (shellcode)	8,548 (0.8%)	2,536 (0.4%)
Total	1,114,244	666,689

Label	Attack 1% Training A	Attack 1% Training B	Attack 1% Training C
-1 (abnormal)	3,020 (0.99%)	3,013 (0.99%)	3,009 (0.99%)
-2 (shellcode)	19 (0.01%)	26 (0.01%)	30 (0.01%)

훈련 데이터셋 수치 통계

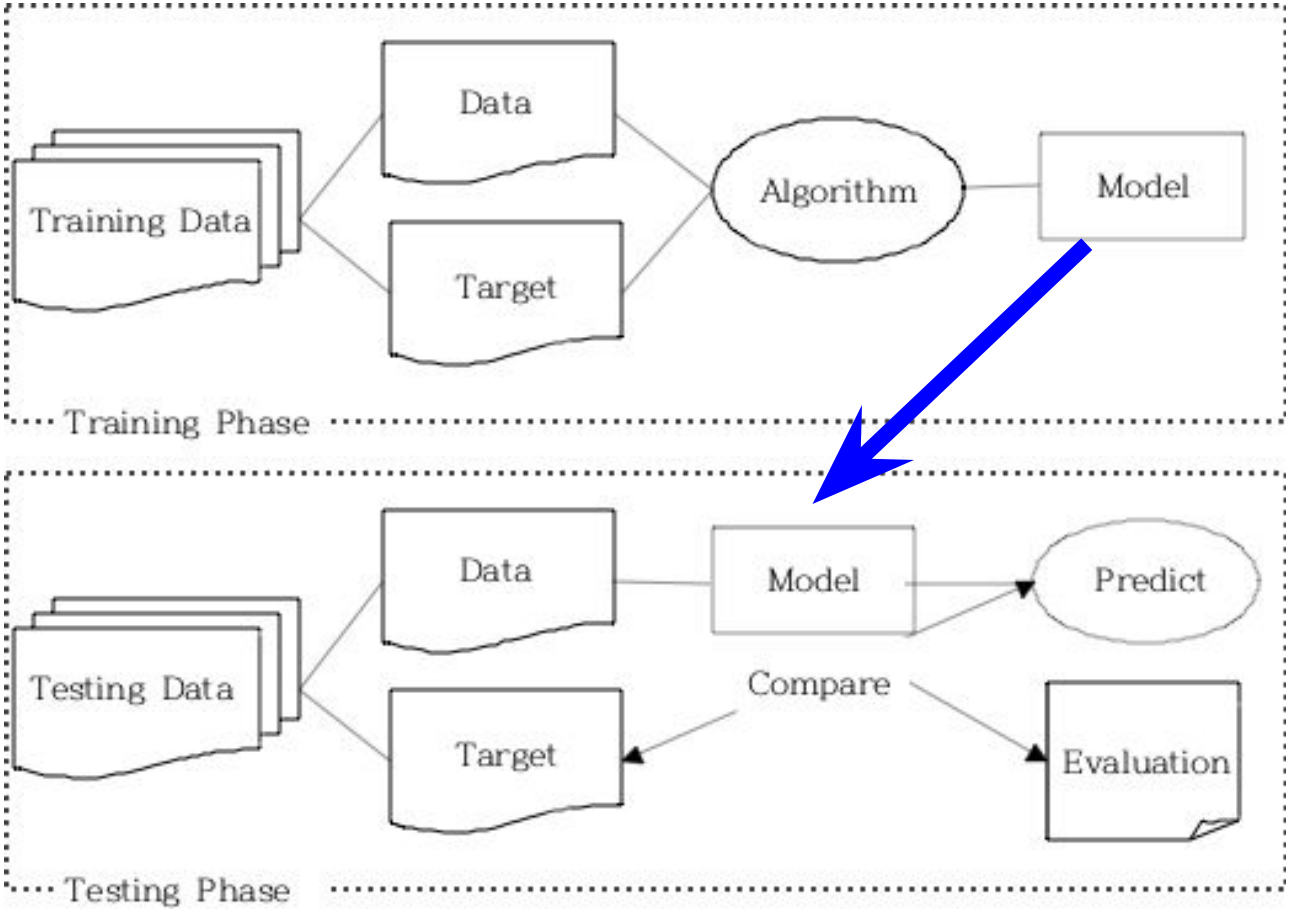
데이터 구축


Kyoto 2006+
특징 – 24개의 feature로 이루어진 labeled 데이터 셋
처리 과정
통계치 확인 – 2013~2015 사이의 월별 –2 label 수
데이터 셋 결정 – 2월 -2 label : 21357 1월 -2 label : 17147 Training set : 7일의 데이터가 가장 多 Test set : Training set의 절반
실제 데이터 환경으로 조정 – Training 셋에서 임의로 1%의 공격만 뽑아서 3개의 셋을 구성
최종 데이터 셋
Test Set 2013.1.3.~9 -2 : 2,536 -1 : 431,722 +1 : 232,431
Training Set 2013.2.19.~25 -2 : 8,548 -1 : 801,767 +1 : 303,929
1% Attack Training Set 2013.2.19.~25 -2 : 19, 26, 30 -1 : 3,020, 3,013, 3,009 +1 : 303,929 (동일)

전처리

- 오류 데이터 필터링
- 문자열 -> 숫자 변환
- 0~1 수치로 정규화
- 사용할 속성 선정
- 클래스별 비율 조정

처리 과정


Data
Training Data
Algorithm
Model
Target
Training Phase
Data
Model
Predict
Testing Data
Compare
Target
Evaluation
Testing Phase

Supervised learning 알고리즘으로 학습

- SVM, Decision Tree, Random Forest, Gradient Boosting Tree, Naive Bayes, Neural Network, KNN
- Scikit-learn 모듈 사용
- 정규화 vs 비정규화

3 class 인식 결과

- 1 (정상)
- -1 (알려진 침입)
- -2 (쉘코드를 이용한 침입)

Classifier	Accuracy	Precision	Recall	F_1-Score	F_2-Score
SVM	0.95	0.95	0.95	0.95	0.95
Decision Tree	0.96	0.96	0.96	0.96	0.96
Random Forest	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**
Gradient Boosting Tree	**0.98**	**0.98**	**0.98**	**0.98**	**0.98**
Naive Bayes	0.86	0.89	0.86	0.86	0.87
Neural Network	0.98	0.98	0.98	0.98	0.98
KNN	0.98	0.98	0.98	0.98	0.98

class별 인식 결과

- 1 (정상)
- -1 (알려진 침입)
- -2 (쉘코드를 이용한 침입)
- 비정규화 결과는 -2 레이블에 대한 Decision Tree 성능이 낮았는데 정규화후 성능 향상

Label	Precision	Recall	F_1-Score	F_2-Score	#
-2	1.00	0.99	0.99	0.99	2536
-1	0.95	1.00	0.97	0.99	431722
1	0.99	0.90	0.95	0.92	232431
Avg	0.96	0.96	0.96	0.96	666689

Decision Tree

Label	Precision	Recall	F_1-Score	F_2-Score	#
-2	1.00	1.00	1.00	1	2536
-1	0.99	1.00	0.99	1	431722
1	0.99	0.98	0.99	0.98	232431
Avg	0.99	0.99	0.99	0.99	666689

Random Forest

2 class 인식 결과

- 정상: 1
- 침입: -1, -2
- -1, -2의 침입 패턴이 유사하다면 성능이 낮아지지 않을 것

Classifier	Accuracy	Precision	Recall	F_1-Score	F_2-Score
SVM	0.95	0.95	0.95	0.95	0.95
Decision Tree	0.97	0.97	0.97	0.97	0.97
Random Forest	0.99	0.99	0.99	0.99	0.99
Gradient Boosting Tree	0.98	0.98	0.98	0.98	0.98
Naive Bayes	0.85	0.89	0.85	0.86	0.86
Neural Network	0.97	0.97	0.97	0.97	0.97
KNN	0.98	0.98	0.98	0.98	0.98

class별 인식 결과

- 정상: 1
- 침입: -1, -2

Label	Precision	Recall	F_1-Score	F_2-Score	#
-1	0.96	0.99	0.98	0.98	434258
1	0.99	0.93	0.96	0.94	232431
Avg	0.97	0.97	0.97	0.97	666689

Decision Tree

1% Attack data의 필요성

- 침입이 1% 정도 발생하는 현실 상황 반영
- class imbalance 데이터를 사용하는 경우 문제점
- 지도 학습 기반의 모델을 업데이트 하여 최신 침입 패턴을 반영하려고 하는 경우, 정상/침입 데이터 비율을 현실에서 나타나는 비율 그대로 사용하지 않도록

1% Atack
2 class 인식 결과

- 정상: 1
- 침입: -1, -2

Classifier	Accuracy	Precision	Recall	F1-Score	F2-Score
SVM	0.76	0.86	0.76	0.76	0.89
Decision Tree	0.81	0.87	0.81	0.81	0.82
Random Forest	**0.92**	**0.90**	**0.90**	**0.90**	**0.90**
Naive Bayes	0.87	0.90	0.87	0.87	0.88
Neural Network	0.78	0.86	0.78	0.78	0.79
KNN	0.82	0.88	0.82	0.82	0.83

	Accuracy	Precision	Recall	F1	F2
SVM	0.76	0.85	0.76	0.76	0.78
Decision Tree	0.81	0.88	0.81	0.81	0.82
Random Forest	**0.88**	**0.91**	**0.88**	**0.89**	**0.89**
Naive Bayes	0.86	0.89	0.86	0.86	0.87
Neural Network	0.80	0.87	0.80	0.81	0.81
KNN	0.82	0.88	0.82	0.82	0.83

Training B

	Accuracy	Precision	Recall	F_1	F_2
SVM	0.76	0.85	0.76	0.76	0.78
Decision Tree	0.85	0.89	0.85	0.85	0.86
Random Forest	**0.90**	**0.92**	**0.90**	**0.90**	**0.90**
Naive Bayes	0.86	0.89	0.86	0.86	0.87
Neural Network	0.81	0.88	0.81	0.82	0.82
KNN	0.81	0.87	0.81	0.81	0.81

Training C

Random Forest 클래스별 인식 결과

- Training A, B, C 모두에 대하여 F_1 수치는 정상 클래스 0.87, 침입 클래스 0.90-0.91,
- F_2 수치는 정상 클래스 0.94-0.95, 침입 클래스 0.87-0.88 수준으로 고른 인식률
- 재현력을 더 중요시하는 F_2 수치의 경우 침입 클래스의 성능이 정상 클래스에 비해 낮음

결론

- Random Forest 선정:
  - 전 성능 비교 수치에서 최고의 인식 성능
  - 클래스별 인식률도 균일
  - 속도 빠름
  - 정규화에 영향 없음
- 훈련 데이터셋에서 침입의 비율에 따라 성능 차이
- 새로운 데이터로 훈련시 정상/이상 클래스 비율을 1:3 수준
- Naive Bayes, Decision Tree, Random Forest 순으로 빠르고 약 30초 소요
- SVM, Naive Bayes는 정규화에 영향을 받는다. 비정규화시 성능이 비균일하고 낮음