ISSN 2586-1816
Journal of Convergence for Information Technology e-ISSN 2586-4440
Vol. 8. No. 6, pp. 165-171, 2018 DOI : https://doi.org/10.22156/CS4SMB.2018.8.6.165

싸이킷런과 사이버위협 데이터셋을 이용한 사이버 공격 그룹의 분류

김경신[1*], 이호준[1], 김성희[2], 김병익[3], 나원식[4], 김동욱[5], 이정환[6]
[1]청강문화산업대학교 모바일IT스쿨 교수, [2](주)디지털트윈 연구원, [3]한국인터넷진흥원 보안기술R&D팀 연구원,
[4]남서울대학교 컴퓨터소프트웨어학과 교수, [5](주)엔코디 연구원, [6](주)에이아이 연구원

Clasification of Cyber Attack Group using Scikit Learn and Cyber Treat Datasets

Kyungshin Kim[1*], Hojun Lee[1], Sunghee Kim[2], Byungik Kim[3], Wonshik Na[4], Donguk Kim[5], Jeongwhan Lee[6]
[1]Professor, School of Mobile IT Tech, ChungKang College of Cultural Industries,
[2]Researcher, DigitalTwin Com. Ltd,
[3]Researcher, Dept. of Security Tech. R&D Team Korea Internet & Security Agency,
[4]Profirssor, Division of Computer Science, NamSeoul Univ.,
[5]Researcher, Ncodi Com. Ltd, 6Researcher, AI Com. Ltd

요 약 최근 IT보안의 화두가 되고 있는 가장 위협적인 공격은 APT공격이다. APT공격에 대한 대응은 인공지능기법을 활용한 대응이외에는 방법이 없다는 것이 현재까지의 결론이다. 여기서는 머신러닝 기법을 활용한 사이버위협 데이터를 분석하는 방법, 그 중에서도 빅데이터 머신러닝 프레임웍인 Scikit Learn를 활용하여 사이버공격 사례를 수집한 데이터셋을 이용하여 사이버공격을 분석하는 머신러닝 알고리즘을 구현하였다. 이 결과 70%에 육박하는 공격 분류 정확도를 보였다. 이 결과는 향후 보안관제 시스템의 알고리즘으로 발전가능하다.

주제어 : 머신러닝, 싸이킷런, 사이버위협, 사이버공격그룹, 사이버공격데이터셋

Abstract The most threatening attack that has become a hot topic of recent IT security is APT Attack.. So far, there is no way to respond to APT attacks except by using artificial intelligence techniques. Here, we have implemented a machine learning algorithm for analyzing cyber threat data using machine learning method, using a data set that collects cyber attack cases using Scikit Learn, a big data machine learning framework . The result showed an attack classification accuracy close to 70%. This result can be developed into the algorithm of the security control system in the future.

Key Words : Machine Learning, Sci-kit Learn, Cyber Treat, Cyber Attack Group, Cyber Attack Datasets

1. 서론

1.1 연구 배경

바야흐로 인공지능, 그 중에서도 머신러닝의 중요성과 그 필요성이 극도로 강조되고 있는 현실이다.[1,2] 알파고라는 바둑 프로그램에서 온 세계인의 이목을 집중시킨 머신러닝은 이제 인간의 지능을 흉내 내는 단계를 넘어 인간의 지능을 뛰어넘는 존재로 부각되고 있다. 특히 이

*This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2017-0-00158, Development of Cyber Threat Intelligence(CTI) analysis and information sharing technology for national cyber incident response)
*Corresponding Author : Kyung-shin Kim(kskim@ck.ac.kr)
Received October 12, 2018 Revised November 13, 2018
Accepted December 20, 2018 Published December 31, 2018

인공지능이 더 많이 더 빨리 적용되어야 한다고 요구되는 여러 활용분야중의 하나가 정보보안 분야라는 것은 이미 알려진 일이다. 이 분야는 인간의 판단 능력과 유사한 능력을 가진 프로그램이 처리할 수 있는 분야이기도 하고, 악성코드의 발생 빈도와 그 처리능력이 인간의 한계를 뛰어넘는 분야이기 때문에 이미 머신러닝이 제일먼저 도입되어야 할 분야라는데 이견이 없기 때문이다.[3,4]

1.2 연구 동향

국외는 물론 국내에서도 이 분야의 연구가 활발히 진행되고 있다. KISA에서도 사이버 침해대응에 대한 신기술 적용방안에 대한 연구를 활발히 진행하고 있었고 최근의 연구과제로 2014년 12월 <악성코드 분석>, 2017년 1월의 <머신러닝 기반의 침해사고 공격분석 방안 연구>가 있었고, 2017년 9월에는 텐서플로우로 악성코드의 학습과 분류 알고리즘을 개발한 <머신러닝 기반 악성코드 분석 알고리즘 적합성 연구>가 수행되었다.[5,6] 빅데이터를 활용한 정보보안 분야에서 스플렁크(Splunk Enterprise)는 빼놓을 수 없다. 스플렁크는 빅데이터, 머신데이터, IoT, 정보보안 등 다양한 IT분야에서 글로벌한 영향력을 미치고 있다. 특히 정보보안 분야에서 발생하는 로그 및 실시간 이벤트 데이터와 다양한 장비 데이터를 수집하고 모니터링하며 검색, 분류, 분석할 수 있는 엔진을 제공하는 빅데이터 분석에 중점을 쏟고 있다. 모든 소스의 머신 데이터(Machine Data)를 실시간으로 수집하고 인덱싱하며, 이를 통해 데이터를 검색, 모니터링, 분석 및 가상화하여 새로운 통찰력과 인텔리전스를 얻는 방식을 쓰고 있다.

2. 연관분석 관련연구

2.1 하둡 활용

Apache Hadoop은 컴퓨터 클러스터 시스템을 위한 분산 데이터 저장과 클러스터 전체에 분산된 데이터 처리를 위해 기초단계부터 설계된 오픈소스 소프트웨어 프레임워크이다. Hadoop의 대표적인 구조는 HDFS(Hadoop Distributed File System)과 MapReduce이다.

2.1.1 HDFS

대용량 파일을 분산된 서버에 저장하고, 데이터를 빠르게 처리할 수 있는 파일 시스템이다. 또한 저사양의 서버를 이용해서 스토리지를 구성할 수 있어 기존의 file system에 비해 장점을 가진다.

2.1.2 MapReduce

간단하게 생각해서 처리할 데이터를 여러 개로 나누어서 처리하는 것으로 생각하면 된다. 이들은 map, reduce, shuffle, sort와 같은 4개의 스테이지를 가지는데, 이것은 아래 Fig. 1과 같다.

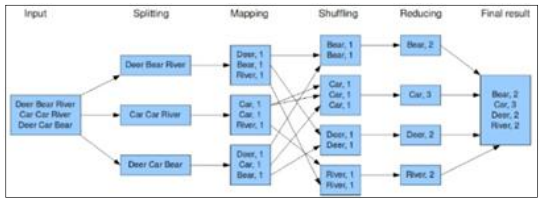


Fig. 1. MapReduce Structure

2.1.3 하둡의 설치

설치 환경은 ubuntu 18.04, Hadoop-2.6.5이다. Hadoop을 설치하기 전에 설정해두어야 할 것이 있다. Hadoop의 경우 ssh를 사용하여 접속하게 되고, java를 기반으로 하기 때문에 jdk와 ssh를 설치해야 한다. 또한 Hadoop은 server로 사용되므로 Fig. 2와 같이 'openssh-server'를 설치해야한다.

```
dino@ubuntu:~$ sudo apt-get install openssh-server
[sudo] password for dino:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
```

Fig. 2. Hadoop Install - openssh-server

2.2 스파크 활용

2.2.1 스파크 구조

Apache Spark는 Hadoop과 동일한 분산 처리 시스템(클러스터 시스템)이지만, 대용량을 처리하는 hadoop과는 다르게 비교적 적은 데이터를 빠르게 처리하는 플랫폼이다. 다양한 처리 방식을 가지고 있기 때문에 다른 언어와도 사용하기에 매우 좋은 플랫폼이다. Spark에서 지원하는 대표적인 라이브러리 4개는 다음와 같다.

- Spark SQL:SQL지원(scala, java, python, R)
- Spark Streaming:실시간 스트리밍 처리 지원(scala,

java, python)
- Spark MLlib:머신러닝 application 개발 지원(scala, java, python, R)
- Spark GraphX:그래프 처리, 알고리즘을 위한 라이브러리(scala)

2.2.2 스파크 설치

설치의 경우 spark.apache.org/downloads.html에서 가능하다. 현재 사용할 버전은 2.0.0버전이다. 설치하기 전에 jdk1.8이상의 java를 먼저 설치해주어야 한다. Fig. 3과 같이 설치를 마무리하고 spark의 설치 경로를 환경변수에 등록해주어야 한다.

```
export SPARK_HOME=/usr/local/bin/spark-2.0.0-bin-hadoop2.7
export PATH=$PATH:$SPARK_HOME/bin
export JAVA_HOME=/Library/Java/JavaVirtualMachines/jdk1.8.0_171.jdk/Contents/Home
```

Fig. 3. Spark Install - Variable Edit

spark action 중에서 가장 대표적인 액션인 map,reduce,flatMap을 살펴본다.
- map(fn) : RDD의 모든 데이터(lambda로 설정한 것)를 순회하면서 전달 받은 함수(fn)를 적용하고 함수가 리턴하는 output을 선택한다.
- reduce(fn) : 함수 fn을 RDD의 모든 데이터에 적용하고 최종 결과는 함수 정의에 의해 계산한다. 함수는 반드시 두개의 parameter를 가지고 있어야 하고, 결과는 1개를 내보내야 한다.
- flatMap(fn) : RDD의 모든 데이터를 순회하면서 전달받은 함수(fn)을 적용하고 반환하는 output을 선택한다.

설명을 보면 map, filter, flagtMap의 경우 전달받은 함수인 fn을 적용하고 결과를 반환한다는 공통점을 가진다. 우선 filter와 map류와 다른 점은 filter는 함수 fn에 대해서 true, false가 되는 조건을 넣어야 하며, true인 결과를 반환한다는 차이점을 가지고 있고, map과 flatMap의 차이점은 아래 Fig. 4를 보면 알 수 있다.

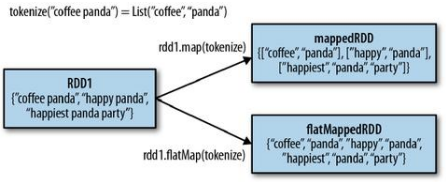


Fig. 4. Function Structure

2.3 문제점 및 차별성

앞에서 살펴본 솔루션과 소프츠웨어의 경우 특정 회사의 저작권이 보호되고 특히 비용이 지불되어야 하는 문제를 가지고 있다. 이 연구에서는 오픈소스를 이용한 알고리즘의 개발과 그 검증을 통해 빅데이터 머신러닝을 이용한 보안위협 분석을 실시한다.

3. 사이버위협 데이터셋 분석

3.1 스파크 싸이킷런 활용

여기서는 KDD99데이터셋을 이용하여 각 사이버 rydrur에 대한 클러스크링을 실시하였다. 먼저, Fig. 5와 같이 스파크 세션을 만든다.

```
from pyspark import SparkContext, SparkConf
sc = SparkContext(conf=SparkConf().setAppName("KDDCup99"))
```

Fig. 5. Create Session

그 이후에 해야 할 내용은 Fig. 6과 같이 사용할 data를 불러오는 것이다. 여기서 사용된 dataset은 KDD99 dataset이다.[7]

```
from collections import OrderedDict
from time import time
data_file = "./kddcup.data"
raw_data = sc.textFile(data_file)

labels = raw_data.map(lambda line: line.strip().split(",")[-1])
label_counts = labels.countByValue()
```

Fig. 6. Loading Datasets

먼저 data를 읽어온 후에 각 label에 대한 정보를 불러오는 것을 알 수 있다. 이제 label들을 갯수에 맞춰서 정렬해야 한다. 정렬을 하고난 후엔 아래 Fig. 7과 같은 결과를 얻을 수 있다.

```
smurf. 2807886
neptune. 1072017
normal. 972781
satan. 15892
ipsweep. 12481
portsweep. 10413
nmap. 2316
back. 2203
warezclient. 1020
teardrop. 979
pod. 264
guess_passwd. 53
buffer_overflow. 30
land. 21
warezmaster. 20
imap. 12
rootkit. 10
loadmodule. 9
ftp_write. 8
multihop. 7
phf. 4
perl. 3
spy. 2
```

Fig. 7. Result of Sorting

3.2 클러스트링과 결과치 획득

Data의 경우 Raw Data상태보다는, 이를 더 효율적으로 사용하기 위한 Scaler를 사용해서 상대적인 값들로 변환해주는 것이 좋다. 변환 후에는 KMeans를 이용해서 clustering을 진행한다. Fig. 8은 클러스트링 결과이다.

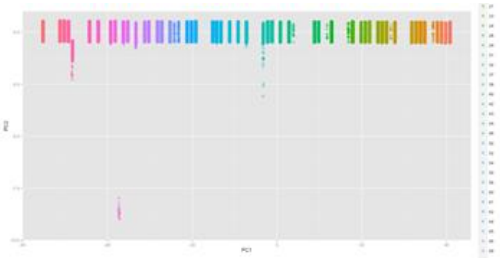


Fig. 8. Result of Clustrings

4. 공격그룹 분석 결과

4.1 데이터셋의 정의

실제로 분석에 사용한 데이터는 KISA R&D팀에서 제공한 malware-cve_description을 사용하였고, 알고리즘은 KNN을 사용하였다.[8,9]

KNN 알고리즘은 K-근접 이웃 알고리즘이라고 한다. 머신러닝의 분류 기법에 쓰이는 대표적인 알고리즘으로 여러 분류기에 사용된다. KNN은 train data가 충분히 존재해야 사용할 수 있는 알고리즘이다. 알고리즘의 Flow는 Fig. 9와 같다.

Fig. 9. Algorithm Flow

KNN은 이처럼 알고리즘 구현에 있어서 매우 간단하며 분산, 밀집이 고르고 좋은 데이터일수록 좋은 결과를 볼 수 있다. 하지만 데이터가 많아지면 분류 속도가 느려질 수 있고, 다차원 데이터에서는 계산량이 많아지는 단점을 가지고 있다. KISA R&D 팀에게 받은 cve 데이터는 cve_code를 label로 생각했을 때 분포가 매우 고르다는 점에서 KNN을 사용하게 되었다. KNN은 라이브러리로 Sci-kit learn에 KNeighborsClassifier로 존재한다. 먼저 데이터로 지도 학습이 가능한지에 대한 여부를 확인해보았다. 주어진 데이터의 column중 'cve_code'라는 column이 다른 것에 비해 분포가 좋았다.

```
CVE-2013-0422    2835
CVE-2012-4681    2826
CVE-2011-2140    2820
CVE-2011-3544    2817
CVE-2014-6332    2752
CVE-2014-0569    1826
CVE-2015-0336    1581
CVE-2015-5119     848
CVE-2015-2419     651
CVE-2010-2884     571
CVE-2016-0189     369
CVE-2015-0311     222
CVE-2015-3133     212
CVE-2013-0074     151
CVE-2014-0322     126
CVE-2013-2551     125
CVE-2016-7200      76
CVE-2014-8439       2
CVE-2013-0634       2
CVE-2015-0313       2
CVE-2015-3113       2
CVE-2014-0556       1
CVE-2015-3043       1
CVE-2016-1019       1
CVE-2013-3897       1
CVE-2012-1889       1
Name: cve_code, dtype: int64
```

Fig. 10. Datasets Classfy

4.2 공격 데이터의 분류

Fig. 10과 같은 데이터의 구분표는 cve_code에 대한 분포를 나타내는 것인데, 마지막 부분 소수의 elements를 제외하면 고른 분포를 얻을 수 있다. 그 후에 data의 column 중 공통된 이름을 가지는 것들에 대해서 Fig. 11처럼 mapping했다. Mapping의 대상은 'malware_type', 'exploit_kit'이 된다.

```
1     가상통화 채굴
2     금융사이트 파밍
3     기타
4     다운로더
5     드롭퍼
6     랜섬웨어
7     백도어
8     비정상파일
9     애드웨어
10    원격제어
11    정보유출
12    키로깅
```

Fig. 11. Keywords

먼저 pandas를 사용하여 data processing을 진행, 학습을 위해 Sci-kit learn의 KNeighborsClassifier을 사용하였다. 먼저 dataset에 대해서 문제가 생겼던 것은 string에 대한 것이었다. String은 학습에 적절하지 않아서 float범위의 수치로 encoding을 해주어야 한다. 이 Data에 대해서 train data와 test data의 비율은 9:1로 사용하였고, 이에 대해서 KNeighborsClassifier를 사용한 결과 약 40%의 결과를 얻을 수 있었다.

4.3 데이터셋의 보완과 실험 결과

Fig. 9의 데이터셋의 분포에 대한 재설정 필요성을 알 수 있다. 옆의 데이터에서 분포가 고르지 않은 cve_code에 대해서 제외시키고 학습 혹은 cve code를 연관성이 있는 것들끼리 재설정후 지도학습 모델을 생성하였다. 현재에는 classifier 1개만을 사용하였지만, KMeans, CNN, KNN 등 여러 알고리즘을 실험 후 적절 model 생성한다. 또한 문자열은 적절하지 않은 자료형(한글 포함). 이를 mapping혹은 encoding을 거쳐서 hashing보다 신뢰성을 가지는 encoding 필요하다.[10-14]

데이터의 분포를 변경(가장 분포가 좋은 8개 label을 사용) 하였다. Fig. 12는 선택된 데이터의 모습이다.

Data를 보면 md5, sha256이라는 column이 존재하는데 이는 via_url을 hashing한 column으로 보인다. 동일 의미의 column이므로 column을 제거하고 진행하였다.

```
CVE-2013-0422    2835
CVE-2012-4681    2826
CVE-2011-2140    2820
CVE-2011-3544    2817
CVE-2014-6332    2752
CVE-2014-0569    1826
CVE-2015-0336    1581
```

Fig. 12. Datasets ReAllocate

4.4 실험 결과

데이터셋의 재설정과 엔코딩 그리고 유사 컬럼을 삭제하는 등의 스케일링 작업 후 KNeighborClassifier를 사용하여 데이터를 분류했다. 그 결과 Fig. 13과 같이 66%의 정확도를 얻을 수 있었다.

```
from sklearn.metrics import accuracy_score
acc = accuracy_score(pred,labels_test)
print("accuracy : ",acc)

accuracy :  0.6689576174112256
```

Fig. 13. Result of Experiment

5. 결론

정보보안 분야에서 스플렁크를 이용한 빅데이터 데이터 연관분석은 나름대로의 정확도를 가지고 있어서 현업에서 사용할 수 있다는 평을 가지고 있으나 나름대로의 문제점을 가지고 있다.[15] 당연한 이야기 이지만 스플렁크는 고가의 상용 소프트웨어이고, 특정 회사의 제품이므로 본 연구과제에서는 오픈소스를 활용한 빅데이터 연관분석 알고리즘을 개발하고 결과를 검증하는 연구 단계를 수행할 계획을 수립하였다. 연구간 Apache Spark와 싸이킷런을 활용하였다. Hadoop과 동일한 분산 처리 시스템(클러스터 시스템)이지만, 대용량을 처리하는 Hadoop과는 다르게 비교적 적은 데이터도 빠르게 처리 가능한 플랫폼이다. 다양한 처리 방식을 가지고 있기 때문에 다른 언어와도 사용하기에 매우 좋은 플랫폼이다. Spark와 Sci-kit learn을 공공 데이터셋인 KDD99 dataset에 적용한 실험 이후 KISA R&D팀에서 제공한 용역과제 결과물중의 하나인 malware-cve_description 데이터셋을 사용하였고, 알고리즘은 K-근접 이웃 알고리즘이라는 KNN을 사용하였다. KNN은 train data가 충분히 존재해야 사용할 수 있는 알고리즘으로 알려져 있다. KISA R&D 팀의 cve 데이터는 cve_code를 label로 생각했을 때 분포가 매우 고르다는 점에서 사용하게 되었다. KNN은 라이브러리로 sci-kit learn에 KNeighborsClassifier로 존재한다. 최초, Unbalanced한 RAW 데이터를 사용하고 KNeighborsClassifier를 사용한 분류 실험 결과 약 40%의 분류 정확성을 얻었다. 그러나 데이터셋의 보완과 모델 재정립과 스케일링 그리고 실험을 거쳐 최종적으로 66%의 정확도 결과를 얻었다. 향후 데이터셋의 구성과 알고리즘의 정확성을 보완하면 90% 이상의 정확도에 도달할 수 있을 것으로 본다.

REFERENCES

[1] Malware Images: Visualization and Automatic Classification, https://vision.ece.ucsb.edu/research/signal-processing-malware-analysis

[2] S. H. Seok. (2016). Malware Family Classify of Convolution Neural Network using Imagification. *Journal of the Korea Institute of Information Security & Cryptology, 26(1).*

[3] H. J. Kim & E. J. Yoon. (2017). AI Deep Learning

protection of Malware Imagification. *Journal of The Institute of Electronics and Information Engineers, 54(2).*

[4] J. H. Kwon. (2011). Malware detection of Various code using Action Graph. *Security of Information Society Journal, 21(2).*

[5] C. K. Kong. (2011). *Malware Host Detection using Spam Mail Analysis.* Korea Internet & Security Agency Final Report.

[6] K. S. Kim. (2018). Malware Analysis Algorithm using Machine Learning. *International Journal of Engineering & Technology, 7(2.12),* 80-83.

[7] T. K. Kwon. (2016). *Malewave Various Group Classfy using Data Mining.* Korea Internet & Security Agency Final Report.

[8] E. K. Yang. (2010). *Deveop of Performance Factor and Collect of Malware Analysis.* Korea Internet & Security Agency Final Report.

[9] J. S. Moon. (2010). *Neutralization Algorithm Study using Execution Self-Compression file.* Korea Internet & Security Agency Final Report.

[10] B. I, Kim. (2018), A Study on Cyber Threat Intelligence Analysis (CTI) Platform for Proactive Detection of Cyber Attacks Based on Automated Analysis. *The Journal of Korea Telecom Society, Fall Symposium,* 578-579.

[11] B. I, Kim. (2016), A Study on the ID Management System of Cyber Threat and its Relevant Information for Cyber Threat Intelligent Analysis. *The Journal of Korea Telecom Society, Winter Symposium,* 959-960.

[12] Daesung Moon, Hansung Lee, (2014), "Feature Extraction for Host based Anomaly Detection", *The Journal of Korea Electronics Society, Summer Symposium,* 591-594

[13] D. H. Kim & K. S. Kim. (2018). DGA-DNS Similarity Analysis and APT Attack Detection Using N-gram. *The Journal of Korea Computer Secret Society, 28(5),* 591-594.

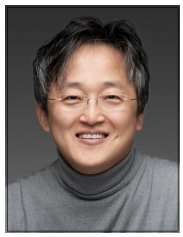
[14] D. G. Kim & C. H. Kim. (2018). Study on APT Attack Response Techniques Based on Big Data Analysis. *The Journal of Society of Convergence Knowledge, 4(1),* 29-34.

[15] Splunk Product Bries. (2018). *Splunk Enterprise Security.* https://www.splunk.com/pdfs/product-briefs/splunk-enterprise-security.pdf

김 경 신(Kim, Kyungshin) [정회원]

- 1993년 2월 : 연세대학교 전자공학과(공학석사)
- 2007년 8월 : 경희대학교 컴퓨터공학과(공학박사)
- 2000년 3월~현재 : 청강문화산업대학교 모바일IT스쿨 교수
- 관심분야 : 머신러닝, 빅데이터, 정보보안, 사물인터넷, CTI
- E-Mail : kskim@ck.ac.kr

이 호 준(Lee, Hojun) [정회원]



- 1994년 2월 : 서울시립대학교 산업디자인학과(학사)
- 2002년 2월 : 국민대학교 대학원 시각디자인전공(석사)
- 2002년 3월~현재 : 청강문화산업대학교 스마트미디어전공 교수
- 관심분야 : 3D프린팅, 사물인터넷, 산업디자인
- E-Mail : tommy@ck.ac.kr

김 성 희(Kim, Sunghee) [정회원]

- 2019년 2월 : 광운대학교 컴퓨터공학과(학사)
- 2018년 1월 ~ 현재 : ㈜디지털트윈 연구원
- 관심분야 : 머신러닝, 정보보안, 디지털트윈
- E-Mail : kkrs0512@naver.com

김 병 익(Kim, Byungik) [정회원]

- 2010년 2월 : 아주대학교 정보및컴퓨터공학과
- 2010년 7월~현재 : 한국인터넷진흥원 선임연구원
- 관심분야 : 머신러닝, 정보보안, 사이버위협분석
- E-Mail : kbi1983@kisa.or.kr

나 원 식(Na, Wonshik) [종신회원]

- 2005년 8월 : 경희대학교 컴퓨터공학과(공학박사)
- 2001년 3월 ~ 2003년 2월 : ㈜성신섬유 전산실장
- 2006년 3월 ~ 현재 : 남서울대학교 컴퓨터소프트웨어학과 교수
- 관심분야 : 네트워크보안, 무선LAN, 모바일컴퓨팅, 의료정보, 전자제어
- E-Mail : winner@nsu.ac.kr

김 동 욱(Kim, Dongwook) [정회원]

- 1993년 2월 : 금오공과대학교 컴퓨터공학과(공학사)
- 2002년 2월 : 국방대학교 전산정보학과(공학석사)
- 2013년 3월 ~ 현재 : ㈜엔코디 대표이사
- 관심분야 : 머신러닝, 정보보안, M&S, 사물인터넷
- E-Mail : dw.kim@ncodi.co.kr

이 정 환(Lee, Jeonghwan) [정회원]

- 2008년 2월 : 고려대학교 경영학과(학사)
- .2017년 3월~현재 : 주식회사 에이아이 연구원
- 관심분야 : 머신러닝, 정보보안, 사물인터넷, 3D모델링
- E-Mail : edlee@ai-korea.com