*The Korean Journal of Applied Statistics* (2018) DOI: https://doi.org/10.5351/KJAS.2018.31.4.497
**31**(4), 497–505

# A study on variable selection and classification in dynamic analysis data for ransomware detection

Seunghwan Lee[a] · Jinsoo Hwang[a,1]

[a]Department of Statistics, Inha University

(Received May 31, 2018; Revised July 5, 2018; Accepted July 25, 2018)

**Abstract**

Attacking computer systems using ransomware is very common all over the world. Since antivirus and detection methods are constantly improved in order to detect and mitigate ransomware, the ransomware itself becomes equally better to avoid detection. Several new methods are implemented and tested in order to optimize the protection against ransomware. In our work, 582 of ransomware and 942 of normalware sample data along with 30,967 dynamic action sequence variables are used to detect ransomware efficiently. Several variable selection techniques combined with various machine learning based classification techniques are tried to protect systems from ransomwares. Among various combinations, chi-square variable selection and random forest gives the best detection rates and accuracy.

Keywords: ransomware, classification, variable selection, machine learning

## 1. 서론

2011년 이후 동유럽에서 부터 서유럽, 미국 및 캐나다로 빠르게 퍼져나가기 시작한 랜섬웨어는 (O'Gorman과 McDonald, 2012), 현재 전세계 PC 사용자에게 공포의 대상이 되었다. 정보사회가 급속도로 발전하는 만큼 시스템의 약점을 이용하도록 악성 프로그램 또한 동시에 진화하고 있기 때문이다. 최근 한국랜섬웨어침해대응센터(RanCERT)의 발표에 의하면 지난 2017년 한 해 동안 새롭게 발견된 악성코드의 약 70%가 랜섬웨어로 판명되고 있다. 이는 기존의 탐지 방법의 맹점을 이용하는 다양한 변종 랜섬웨어들이 빠르게 증가하고 있음을 의미하고 있다. 침입하려는 패킷들의 시그니쳐 등의 프로그램에 내장된 정보를 이용하는 정적 분석은 악성코드의 탐지에 흔히 사용되었지만 탐지망을 피해 지능적으로 진화하고 있는 랜섬웨어를 탐지해내기에 한계가 있다. 따라서 실제 행동을 통하여 랜섬웨어 여부를 판단하고 프로그램의 작동을 저지하는 동적 분석으로 요즘 연구가 진행되는 추세이다. 본 연구에서는 쿠쿠 샌드박스 가상환경을 이용한 동적 분석 결과 수집된 고차원 자료에 여러 변수선택 방법을 적용하여 랜섬웨어 분류에 유의한 변수를 파악하고 선택된 변수를 이용한 기계학습 모형을 적합하여 분류 성능이 가장 높았던 변수 선택법과 모형의 조합을 확인하고자 한다.

This research was supported by the National Research Foundation (NRF) (NRF-2017R1E1A1A03070865).
[1]Corresponding author: Department of Statistics, Inha University, 100, Inha-ro, Nam-gu, Incheon 22212, Korea. E-mail: jshwang@inha.ac.kr

## 2. 랜섬웨어

### 2.1. 랜섬웨어의 정의 및 특징

랜섬웨어(ransomware)는 몸값(ransom)과 제품(ware)의 합성어로써, 감염된 컴퓨터의 기능을 제한하거나 컴퓨터 내의 중요 파일을 암호화하여 이용할 수 없도록 만든 뒤에 복호화의 대가로 금전을 요구하는 것이 특징인 악성 프로그램의 일종이다(Kim 등, 2017a). 웹사이트 접속이나 메일을 통한 첨부파일 전송, 광고 배너 클릭 등 다양한 경로로 전파되는 것이 특징이다. 2017년 한국랜섬웨어침해대응센터(RanCERT)의 조사 결과, 2015년부터 2016년까지 국내 피해 사례의 약 70%가 웹사이트 링크를 통한 침투에 의한 것으로 밝혀졌다.

### 2.2. 랜섬웨어 분석 방법

랜섬웨어를 비롯한 악성 프로그램(malware) 분석 방법으로는 크게 정적 분석(static analysis)과 동적 분석(dynamic analysis) 방법이 있다. 정적 분석은 실행 파일에 내장된 문자열 헤더정보 등의 시그니처를 기반으로 악성 프로그램 여부를 확인하는 방법이며, 악성코드를 직접 실행하지 않고도 분석이 가능하여 안전한 반면에 실행되는 모든 프로세스를 서명 D/B와 비교해야 하여 시스템에 상당한 부하를 유발하는 문제가 있으며 특히 난독화, 최적화 등의 기법을 적용한 악성코드에는 적용이 어려워 새로운 형태의 악성코드를 판별해 내는 것에 한계가 있다 (Moser 등 2007; Kim 등, 2017b). 동적 분석은 샌드박스 등의 가상 환경에서 파일을 직접 실행시켜 레지스트리, 파일시스템, 프로세스, 네트워크 활동 등을 확인하며 분석하는 방법이다. 파일의 실제 활동에 따른 시스템 변화 분석이 가능하다는 장점이 있지만, 실제 탐지를 진행함에 있어 랜섬웨어가 시스템 내에서 활동하도록 허용해야 한다는 단점을 가지고 있다 (Zhang 등, 2019). 이 외에도 일반 악성코드와는 다른 랜섬웨어의 특징을 이용한 분석 방법으로 미끼 기반 탐지 기법(decoy based detection)이 있다. 이는 랜섬웨어의 주 타깃이 되는 특정 위치의 폴더와 특정 확장자를 가진 파일을 미끼로 두어 이를 대상으로 하는 비정상적 행동을 확인하여 랜섬웨어를 판별하는 방법이다. 그러나 이 방법은 탐지에 규칙성을 가지게 되어 랜섬웨어 공격자가 우회할 수 있는 여지가 있다. 위와 같이 기존에 대표적으로 사용하고 있는 랜섬웨어 분석 기법은 각각의 장점과 단점을 갖고 있기 때문에, 각 분석 기법의 취약점을 파악하고 보완하고자 하는 Moser 등 (2007), Kim 등 (2017b)과 같은 연구가 활발히 진행되고 있다.

### 2.3. 랜섬웨어 탐지

랜섬웨어 탐지에는 정적분석을 통한 탐지 방법이 가장 많이 활용되고 있다. 시그니처 기반 탐지 패턴을 제작하여 자동화하는 Lee 등 (2017) 연구가 있으며, 최근에는 랜섬웨어 프로그램의 Opcode로부터 추출된 특징적인 N-gram sequence를 기계학습 모형의 독립변수로 활용하는 Zhang 등 (2019)의 연구가 있다. 그러나 정적분석만을 활용한 랜섬웨어 탐지는 프로그램에 내장된 정보가 특정 패턴에 매칭되는 프로그램만을 랜섬웨어로 판별하기 때문에 새로운 패턴에 대한 대처가 미흡할 수 있다. 또한 본 연구와 같이 랜섬웨어 탐지에 기계학습 모형을 활용하는 연구 또한 진행되고 있다. 대표적으로 정적 분석과 동적 분석을 함께 이용하는 하이브리드 분석 방법과 기계학습 모형을 이용한 탐지 방법을 제안한 Kim 등 (2017a)의 연구가 있다. 기계학습 모형을 이용하는 랜섬웨어 탐지 연구라는 점에서 본 연구와 유사성을 갖지만, 랜섬웨어 분석에 Opcode 정보와 같은 정적분석 결과를 함께 활용하고 있다는 점과 동적분석 변수로 API 호출 정보만을 독립변수로 이용하였다는 점에서 다양한 동적분석 변수를 활용한 본 연구와 차이가 있다. Sgandurra 등 (2016)의 연구에서 동적분석을 통해 생성된 자료를 파싱해 이항변수로 구성된 고차원 자료를 수집하여 기계 학습 모형으로 랜섬웨어를 탐지하는 자동화 시스템을 제안하였으며, 본

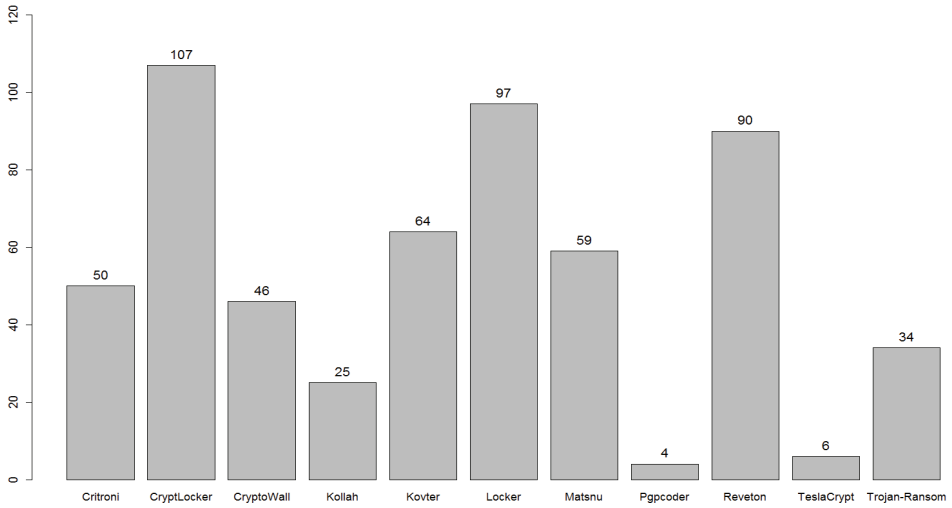


**Figure 3.1.** Ransomware Family

연구에서는 위 연구에서 수집된 동적분석 자료를 기반으로 연구를 수행하였다. 본 연구에서는 위 연구에서 한 단계 더 나아가 고차원 자료에 다양한 변수 선택법을 적용하였을 때 모형의 성능 향상을 확인하고자 한다. 또한 Aragorn 등 (2016)의 연구와 같이 분류 문제에 각광 받고 있는 딥러닝을 활용한 랜섬웨어 탐지 연구가 진행되고 있다.

## 3. 분석 자료 설명

### 3.1. 동적 분석 자료

분석에 활용된 동적 분석 자료는 Sgandurra 등 (2016)의 연구에서 수집된 것으로, 11종(Critroni, CryptLocker, CryptoWall, Kollah, Kovter, Locker, Matsnu, Pgpcoder, Reveton, TeslaCrypt, Trojan-Ransom)의 랜섬웨어 프로그램 582개와 정상 프로그램 942개를 쿠쿠 샌드박스(cuckoo sandbox) 가상환경 상에서 실행시킨 뒤 발생되는 다양한 정보를 추출하고, 전처리 과정과 파싱을 통해 특정 행동 또는 조건 충족 여부를 이항 변수로 기록한 자료이다.(http://rissgroup.org/ransomware-dataset) Figure 3.1에 동적 분석에 사용된 11종의 랜섬웨어의 명칭과 해당 랜섬웨어 프로그램 샘플의 수를 나타내었다.

### 3.2. 자료의 구성 및 특징

자료는 프로그램 고유 ID, 랜섬웨어 여부를 나타내는 이항변수, 랜섬웨어 종류를 나타내는 범주형변수, 그리고 7가지 행동 범주(API stats, Registry Keys Operations, Files Operations, Strings, File Extensions, Directory Operations, Dropped Files Extensions)에 속하는 30,967개의 이항변수로 구성된 $n = 1524, p = 30970$인 고차원 자료이며, Table 3.1에 각 범주의 정의와 포함된 변수의 수를 나타내었다.

본 연구에서 진행된 변수 선택 및 기계 학습 모형 구축 등은 Python의 scikit-learn 및 XGBoost 모듈을 활용하였으며, 전체 1,524개의 자료를 80%는 훈련용 자료로, 20%는 시험용 자료로 나누어 분석에 활용

**Table 3.1.** Features

| 범주 | 정의 | 변수의 개수 |
|---|---|---|
| API stats | 특정 API 호출 여부 | 232 |
| Registry Keys Operations | 특정 레지스트리 키 작업 여부 | 6,622 |
| Files Operations | 특정 파일 작업 여부 | 4,141 |
| Strings | 특정 임베디드 문자열 포함 여부 | 16,267 |
| File Extensions | 파일 작업에 관련된 특정 확장자에 대한 조작 여부 | 935 |
| Directory Operations | 특정 파일 디렉터리 작업 여부 | 2,424 |
| Dropped Files Extensions | 특정 파일 확장자 삭제 여부 | 346 |
| Total | | 30,967 |

하였다. 변수 선택 및 교차 검증을 통한 모형 선택 등 모든 과정이 훈련용 자료에서 이루어졌으며, 시험용 자료에서는 오직 모형의 분류 성능 평가만을 수행하였다. 또한 위의 과정을 100회 반복하여 각 변수 선택 방법과 기계학습 모형 조합의 평균적인 성능을 확인하였다.

## 4. 변수 선택

고차원 자료를 이용할 때 발생할 수 있는 과적합을 방지하고 랜섬웨어 탐지에 유의한 변수를 확인하기 위해 변수 선택을 적용하였으며, 모든 변수를 이용하는 모형을 대조군으로 두어 변수 선택의 효과를 비교하였다.

### 4.1. 변수 선택법

**4.1.1. 카이제곱검정** 첫 번째로 종속변수와 독립변수가 모두 범주형 변수인 경우에 사용할 수 있는 카이제곱검정 변수 선택법을 적용하였다. 분할표에서 기대 도수와 관측 도수의 값의 차이를 이용하는 방법으로 각각의 범주에 대한 비율이 모든 모집단에 대하여 동일하다는 귀무가설과 동일하지 않다는 대립가설을 세우고 검정을 수행하게 된다. $C$개의 범주를 가진 변수와 $I$개의 범주를 가진 변수가 주어졌을 때 두 변수를 이용한 분할표의 $i$번째 행과 $j$번째 열에 해당하는 칸의 관측 도수를 $N_{ij}$, 기대 도수를 $E_{ij}$라고 하면, 카이제곱 통계량은 다음과 같다.

$$\chi^2 = \sum_{i=1}^{C}\sum_{j=1}^{I}\frac{(N_{ij}-E_{ij})^2}{E_{ij}}.$$

귀무가설이 참이라면, 검정 통계량은 자유도가 $(C-1)(I-1)$인 카이제곱분포를 따르고, 검정 통계량 값이 클수록 이는 특정 범주에 속하는 비율이 높다는 것을 의미하게 된다. 결과적으로 종속변수와 독립변수의 카이제곱검정 결과 $p$-값이 작을수록 해당 독립변수는 정상 프로그램과 랜섬웨어를 잘 분류할 수 있는 변수임을 뜻하며, 본 논문에서는 변수 선택의 기준이 되는 $p$-값의 임계값으로 0.05, 0.01, 0.001 세 값을 적용하여 임계값보다 작은 $p$-값을 가지는 변수만을 모형 적합에 이용하였다.

**4.1.2. 상호 정보** 두 번째로 Sgandurra 등 (2016)의 선행 연구에서 사용된 상호 정보(mutual information)순위를 이용한 변수선택법을 적용하였다. 상호 정보를 이용하면 변수 간의 상호 관련성을 확인할 수 있는데, $X$와 $Y$가 결합 확률 질량 함수 $p(x,y)$와 주변 확률 질량 함수 $p(x)$, $p(y)$를 갖는 두 개의 이산 확률 변수라고 가정하였을 때 상호 정보 $I(X;Y)$는 다음과 같이 두 주변분포의 곱과 결합 분포의

**Table 4.1.** Average percentage of the relevant variables for each class

| Variable class | Chi-square test | | | Mutual information | Lasso |
|---|---|---|---|---|---|
| | $p < 0.05$ | $p < 0.01$ | $p < 0.001$ | Top 400 | |
| API Stats | 9.24% | 15.94% | 21.58% | 12.51% | 33.39% |
| Drop File Extensions | 1.97% | 2.77% | 2.11% | 1.79% | 1.04% |
| Registry Keys Operations | 55.32% | 49.06% | 49.69% | 29.68% | 23.11% |
| Files Operations | 6.34% | 6.98% | 6.08% | 10.54% | 2.33% |
| File Extensions | 6.70% | 8.96% | 8.18% | 6.02% | 3.74% |
| Directory Operations | 9.36% | 6.25% | 3.58% | 6.66% | 2.64% |
| Strings | 11.08% | 10.04% | 8.78% | 32.81% | 33.75% |

상대 엔트로피로 정의될 수 있다 (Cover와 Thomas, 2006; Huh와 Choi, 2009).

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}.$$

상호 정보 값이 높은 변수일수록 상대적으로 종속변수와 관련성이 높은 변수임을 의미한다. 본 연구에서는 선행연구에서 제안한 변수의 개수인 상호 정보 순위 상위 400개를 선택하였다.

#### 4.1.3. Lasso

$$\max_{\beta_0, \beta} \left\{ \sum_{i=1}^{N} \left[ y_i \left( \beta_0 + \beta^T x_i \right) - \log \left( 1 + e^{\beta_0 + \beta^T x_i} \right) \right] - \lambda \sum_{j=1}^{p} |\beta_j| \right\}.$$

마지막으로 $L_1$ 정규화 로지스틱 회귀 모형을 적합시켰을 때 선택된 변수들을 확인하고자 하였다. $\lambda$는 훈련용 자료에서의 교차 검증(cross validation)을 통해 오분류율이 가장 낮게 나타난 값으로 결정하였고, 최종적으로 선택된 모형에서의 평균적인 분류 성능을 확인하였다.

### 4.2. 변수 선택 결과

각 변수 선택법을 통해 선택된 변수 군의 비율을 Table 4.1에 나타내었다. 가장 먼저 모든 변수 선택 방법에서 공통적으로 레지스트리 키 작업 관련 변수들이 매우 높은 비율을 차지하는 것을 확인할 수 있었다. 또한 상호 정보와 Lasso 두 방법에서 공통적으로 임베디드 문자열 변수가 비교적 높은 비율을 차지하고 있는 것을 확인할 수 있었으며, 특히 Lasso를 적용하였을 때에 다른 방법들과 달리 API 호출 변수군이 상당히 높은 비율을 보임을 확인할 수 있었다. 결과적으로 선택 방법에 따라 비율의 차이는 있었지만, 정상 프로그램과 랜섬웨어 프로그램이 API 호출, 레지스트리 키 작업, 임베디드 문자열 변수 패턴에서 큰 차이를 보인다는 것을 알 수 있었다.

## 5. 모형 적합 및 분류 성능 평가

### 5.1. 기계학습 모형 적합

동적 분석 자료에 변수 선택법을 적용한 후 정상 프로그램과 랜섬웨어 감염 프로그램을 분류하기 위해 사용한 기계학습 모형으로는 능형 로지스틱 회귀 모형(ridge regression; Ridge), 랜덤 포레스트(random forest; RF), 익스트림 그래디언트 부스팅(extreme gradient boosting; XGBoost), 서포트 벡터 머신(support vector machine; SVM)이다. 각 모형에 사용되는 모수는 $[2 \times 10^{-5}, 2 \times 10^{1}]$ 범

**Table 5.1.** Classification performances

| Variable selection | Avg. # of variables | Model | Accuracy | FPR | Detection rate |
|---|---|---|---|---|---|
| Chi-square test | 1,550($p < 0.05$) | Ridge | $0.9727 \pm 0.0093$ | $0.0185 \pm 0.0099$ | $0.9584 \pm 0.0210$ |
| | | SVM | $0.9720 \pm 0.0090$ | $0.0172 \pm 0.0097$ | $0.9545 \pm 0.0205$ |
| | | RF | $0.9804 \pm 0.0074$ | $0.0164 \pm 0.0092$ | $0.9753 \pm 0.0142$ |
| | | XGBoost | $0.9794 \pm 0.0075$ | $0.0143 \pm 0.0078$ | $0.9693 \pm 0.0172$ |
| | 770($p < 0.01$) | Ridge | $0.9716 \pm 0.0084$ | $0.0205 \pm 0.0093$ | $0.9588 \pm 0.0194$ |
| | | SVM | $0.9716 \pm 0.0078$ | $0.0193 \pm 0.0099$ | $0.9569 \pm 0.0175$ |
| | | RF | $0.9811 \pm 0.0069$ | $0.0151 \pm 0.0085$ | $0.9749 \pm 0.0140$ |
| | | XGBoost | $0.9787 \pm 0.0077$ | $0.0150 \pm 0.0078$ | $0.9686 \pm 0.0174$ |
| | 480($p < 0.001$) | Ridge | $0.9717 \pm 0.0091$ | $0.0204 \pm 0.0103$ | $0.9591 \pm 0.0195$ |
| | | SVM | $0.9716 \pm 0.0091$ | $0.0205 \pm 0.0106$ | $0.9589 \pm 0.0187$ |
| | | RF | $0.9797 \pm 0.0072$ | $0.0154 \pm 0.0087$ | $0.9717 \pm 0.0142$ |
| | | XGBoost | $0.9782 \pm 0.0078$ | $0.0156 \pm 0.0083$ | $0.9681 \pm 0.0175$ |
| Mutual information | 400 | Ridge | $0.9370 \pm 0.0160$ | $0.0443 \pm 0.0194$ | $0.9066 \pm 0.0300$ |
| | | SVM | $0.9476 \pm 0.0140$ | $0.0285 \pm 0.0152$ | $0.9089 \pm 0.0325$ |
| | | RF | $0.9547 \pm 0.0155$ | $0.0288 \pm 0.0134$ | $0.9279 \pm 0.0325$ |
| | | XGBoost | $0.9505 \pm 0.0135$ | $0.0305 \pm 0.0150$ | $0.9197 \pm 0.0286$ |
| Lasso | 116 | Logistic Reg. | $0.9772 \pm 0.0089$ | $0.0174 \pm 0.0110$ | $0.9684 \pm 0.0162$ |
| None | 30,967 | Ridge | $0.9757 \pm 0.0083$ | $0.0196 \pm 0.0108$ | $0.9680 \pm 0.0170$ |
| | | SVM | $0.9743 \pm 0.0087$ | $0.0150 \pm 0.0093$ | $0.9570 \pm 0.0194$ |
| | | RF | $0.9645 \pm 0.0108$ | $0.0362 \pm 0.0146$ | $0.9658 \pm 0.0168$ |
| | | XGBoost | $0.9761 \pm 0.0071$ | $0.0205 \pm 0.0081$ | $0.9705 \pm 0.0161$ |

FPR = false positive rate; Ridge = ridge regression; SVM = support vector machine; RF = random forest; XGBoost = extreme gradient boosting.

위에서 그리드 서치 방법을 이용하였으며, 훈련용 자료를 이용한 교차 검증 결과 오분류율이 가장 낮게 나타나는 모수를 지정하였다.

### 5.2. 각 모형의 평균 분류 성능 비교

자료의 분할과 변수 선택, 기계학습 모형 적합, 교차 검증을 이용한 모형의 선택 그리고 성능 평가의 과정을 100회 반복한 결과 나타난 평균적인 성능은 Table 5.1과 같다.

정분류율 측면에서 카이제곱검정 결과 $p$-값이 0.01보다 작은 변수만을 이용한 랜덤 포레스트 모형이 98.11%로 가장 우수한 성능을 보였으며, 랜섬웨어 프로그램을 랜섬웨어로 올바르게 분류하는 비율인 탐지율 측면에서도 97.49%로 가장 우수한 성능을 보였다. 정상 프로그램을 랜섬웨어로 잘못 분류하는 비율인 false positive rate (FPR) 측면에서 가장 우수한 모형은 1.43%로 $p$-값이 0.05보다 작은 변수만을 이용한 XGBoost 모형이었다. 변수 선택의 효과를 살펴보았을 때, 능형 로지스틱 회귀모형과 서포트 벡터 머신의 경우 변수 선택을 선행한 뒤에 오히려 성능이 하락하는 모습을 보였으나 랜덤 포레스트와 XGBoost는 성능이 상승하는 것을 확인할 수 있었다. Lasso를 이용한 로지스틱 회귀 모형의 경우에 선택된 변수가 가장 적었고 97.72%의 정분류율과 96.84%의 탐지율로 준수한 성능을 보였으나, 카이제곱 검정을 이용한 변수 선택법을 거친 랜덤 포레스트 모형이 상대적으로 더 높은 성능을 보였다.

특이사항으로 상호 정보를 이용한 변수 선택을 적용한 뒤 적합한 모형들의 성능이 Table 5.2에 나타낸 선행 연구의 모형들에 비해 상대적으로 낮은 경향을 보였는데, 이는 선행연구에서 자료 분할 전 변수 선

**Table 5.2.** Classification Performances in preceding research (Sgandurra *et al.*, 2016)

| Variable selection | Avg. # of variables | Model | Accuracy | FPR | Detection rate |
|---|---|---|---|---|---|
| Mutual information | 400 | Logistic Reg. | $0.9762 \pm 0.0090$ | $0.0161 \pm 0.0088$ | $0.9634 \pm 0.0215$ |
| | | SVM | $0.9579 \pm 0.0108$ | $0.0199 \pm 0.0107$ | $0.9219 \pm 0.0244$ |
| | | Naive Bayes | $0.9200 \pm 0.0118$ | $0.0958 \pm 0.0162$ | $0.9453 \pm 0.0212$ |

FPR = false positive rate; SVM = support vector machine.

택을 수행하여 보다 더 많은 정보를 사용하였기 때문인 것으로 확인되었다.

## 6. 결론 및 향후 과제

이항변수만으로 구성된 고차원 자료를 이용한 분류라는 특수한 경우에 카이제곱검정을 이용한 변수 선택법이 종속변수와 관련성이 낮은 독립변수를 제거함으로써 일부 모형의 성능 향상에 도움을 줄 수 있는 것을 확인하였다. PC에 새로운 프로그램이 설치됨에 따라 동적 분석을 통한 변수 추출과 랜섬웨어 분류가 진행되는 백신 프로그램에 본 연구에서 제안된 변수 선택법 및 모형이 이용될 경우, 탐지 성능 면에서 많은 향상이 있을 것으로 기대한다.

또한 본 연구에서 이용한 동적분석 자료의 수가 1,524개로 많은 수가 아니며 랜섬웨어의 종류도 11종으로 다양하지 않기 때문에 보다 많은 수와 다양한 종류의 표본을 이용한 연구가 필요다고 생각되며, 프로그램의 특정 행동 횟수나 순서 등 다양한 정보를 활용하여 변수 선택과 랜섬웨어를 탐지할 수 있는 방법에 대한 연구가 필요하다. 추가적으로 본 연구에 활용된 자료에 8개의 은닉층을 가진 딥러닝 모형(deep neural network; DNN)을 적용하여 탐지 성능을 확인한 결과, 변수 선택법에 따라 최소 93%에서 최대 97%의 정분류율을 보임을 확인하였으며, 향후 딥러닝 모형의 성능을 좀 더 보완하는 연구와 다양하게 진화하는 랜섬웨어의 변화에 능동적으로 대처할 수 있는 위계적 탐지 방법에 대한 연구를 추가로 진행하려고 한다.

## 감사의 글

랜섬웨어 동적 분석 자료의 사용을 흔쾌히 허락해주신 Daniele, Luis, Rahib, Emil에게 감사드립니다.

## References

Aragorn, T., YunChun, C., YiHsiang, K., and Tsungnan, L. (2016). Deep Learning for Ransomware Detection, *IEICE Technical Report*, **116**, 87–92.

Cover, T. M. and Thomas, J. A. (2006). *Elements of Information Theory*, John Wiley & Sons, New York.

Huh, M. Y. and Choi, B. S. (2009). Variable selection based on mutual information, *Communications of the Korean Statistical Society*, **16**, 143–155.

Moser, A., Kruegel, C., and Kirda, E. (2007). Limits of Static Analysis for Malware Detection, *23rd Annual Computer Security Applications Conference*.

Kim, J., Ji, S., and Kim, S. (2017a). A machine learning based ransomware detection model using a hybrid analysis, *Journal of Security Engineering*, **14**, 263–280.

Kim, J. H., Park, K. S., and Park, Y. H. (2017b). A study of vulnerability analysis of ransomware detection techniques, *The Korean Institute of Communications and Information Sciences 2017 Summer Conference*, 590–591.

Lee, H., Seong, J., Kim, Y., Kim, J., and Gim, G. (2017). The automation model of ransomware analysis and detection pattern, *Journal of the Korea Institute of Information and Communication Engineering*, **21**, 1581–1588.

O'Gorman, G. and McDonald, G. (2012). Ransomware: a growing menace, *Symantec Security Response*.

Sgandurra, D., Munoz-Gonzalez, L., Mohsen, R., and Lupu, E. C. (2016). Automated Dynamic Analysis of Ransomware: Benefits, Limitations and use for Detection. *arXiv preprint arXiv:1609.03020*.

Zhang, H., Xiao, X., Mercaldo, F., Ni, S., Martinelli, F., Sangaiah, A., K.(2019). Classification of ransomware families with machine learning based on N-gram of opcodes, *Future Generation Computer Systems*, **90**, 211–221.

# 랜섬웨어 탐지를 위한 동적 분석 자료에서의 변수 선택 및 분류에 관한 연구

이승환[a] · 황진수[a,1]

[a]인하대학교 통계학과

(2018년 5월 31일 접수, 2018년 7월 5일 수정, 2018년 7월 25일 채택)

**요 약**

최근 랜섬웨어는 일반 PC 사용자에 비해 상대적으로 수준 높은 보안 체계를 갖추고 있는 기업과 정부 기관에 침입하여 상당한 피해를 입히는 등 기존 보안 체계의 허점을 찾아 진화하는 모습을 보이고 있다. 이처럼 계속해서 변화하는 랜섬웨어를 탐지하기 위해 랜섬웨어의 특징을 파악하는 정적 분석과 동적 분석과 관련된 연구가 활발히 이루어지고 있다. 본 연구에서는 582개의 랜섬웨어 샘플과 942개의 정상 샘플 프로그램을 쿠쿠 샌드박스 가상환경 내에서 실행시킨 뒤, PC에서 이루어지는 30,967가지의 행동 여부를 기록한 동적 분석 자료를 활용하여 랜섬웨어 분류에 유의한 변수를 탐색하기 위한 여러 변수 선택 방법의 적용과 랜섬웨어 분류를 위한 기계학습 모형들을 구축하고자 하였다. 변수 선택법으로 LASSO와 이항변수 만으로 이루어진 고차원 자료라는 특성을 활용하기 위한 카이제곱검정을 이용한 변수 선택, 선행 연구에서 이용된 방법인 상호정보를 이용한 변수 선택법을 적용하였으며 기계 학습 모형으로는 능형 로지스틱 회귀, 서포트 벡터 머신, 랜덤 포레스트, XGBoost가 활용되었다. 연구 결과, 정상 프로그램과 구별되는 랜섬웨어 프로그램만의 특징적인 행동을 확인할 수 있었으며 여러 변수 선택법과 기계학습 분류 모형들의 조합 중, 주어진 자료에서 카이제곱검정을 이용한 변수 선택법과 랜덤 포레스트 모형의 조합이 가장 높은 탐지율과 정분류율을 보이는 것을 확인하였다.

주요용어: 랜섬웨어, 분류, 변수선택, 기계학습

본 연구는 한국연구재단의 지원을 받아 진행되었음 (NRF-2017R1E1A1A03070865).

[1]교신저자: (22212) 인천광역시 남구 인하로 100, 인하대학교 통계학과. E-mail: jshwang@inha.ac.kr