JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 27 No. 5, pp. 149-156, May 2022
https://doi.org/10.9708/jksci.2022.27.05.149

OLE File Analysis and Malware Detection using Machine Learning

Hyeong Kyu Choi*, Ah Reum Kang*

*Student, Dept. of Information Security, Pai Chai University, Daejeon, Korea
*Professor, Dept. of Information Security, Pai Chai University, Daejeon, Korea

[Abstract]

Recently, there have been many reports of document-type malicious code injecting malicious code into Microsoft Office files. Document-type malicious code is often hidden by encoding the malicious code in the document. Therefore, document-type malware can easily bypass anti-virus programs. We found that malicious code was inserted into the Visual Basic for Applications (VBA) macro, a function supported by Microsoft Office. Malicious codes such as shellcodes that run external programs and URL-related codes that download files from external URLs were identified. We selected 354 keywords repeatedly appearing in malicious Microsoft Office files and defined the number of times each keyword appears in the body of the document as a feature. We performed machine learning with SVM, naïve Bayes, logistic regression, and random forest algorithms. As a result, each algorithm showed accuracies of 0.994, 0.659, 0.995, and 0.998, respectively.

▸ **Key words**: OLE, malware, Microsoft Office, shellcode, VBA macro, random forest

[요 약]

최근 전 세계적으로 사용되는 Microsoft Office 파일에 악성코드를 삽입하는 문서형 악성코드 사례가 증가하고 있다. 문서형 악성코드는 문서 내에 악성코드를 인코딩하여 숨기는 경우가 많기 때문에 백신 프로그램을 쉽게 우회할 수 있다. 이러한 문서형 악성코드를 탐지하기 위해 먼저 Microsoft Office 파일의 형식인 OLE(Object Linking and Embedding) 파일의 구조를 분석했다. Microsoft Office에서 지원하는 기능인 VBA(Visual Basic for Applications) 매크로에 외부 프로그램을 실행시키는 쉘코드, 외부 URL에서 파일을 다운받는 URL 관련 코드 등 다수의 악성코드가 삽입된 것을 확인했다. 문서형 악성코드에서 반복적으로 등장하는 키워드 354개를 선정하였고, 각 키워드가 본문에 등장하는 횟수를 feature 로 정의했다. SVM, naïve Bayes, logistic regression, random forest 알고리즘으로 머신러닝을 수행하였으며, 각각 0.994, 0.659, 0.995, 0.998의 정확도를 보였다.

▸ **주제어**: 복합 파일 이진 구조, 악성 파일, ms 오피스, 쉘코드, vba 매크로, 랜덤 포레스트

• First Author: Hyeong Kyu Choi, Corresponding Author: Ah Reum Kang
*Hyeong Kyu Choi (1684040@pcu.ac.kr), Dept. of Information Security, Pai Chai University
*Ah Reum Kang (armk@pcu.ac.kr), Dept. of Information Security, Pai Chai University
• Received: 2022. 03. 18, Revised: 2022. 04. 29, Accepted: 2022. 04. 29.

Copyright © 2022 The Korea Society of Computer and Information
http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

Microsoft Office 문서 파일이나 HWP(Hangul Word Processor) 문서 파일의 사용량은 점점 증가하고 있다. 대부분의 사람들은 관공서나 국가 기관 홈페이지에 업로드 되는 파일들 또는 지인이 보낸 메일을 의심하지 않고 파일을 저장하고 열어보게 된다. 악의적인 사용자들은 이 점을 이용해 사회공학적 공격을 행하며, 사용자는 항상 쓰던 익숙한 프로그램이기 때문에 컴퓨터가 악성코드에 감염된 사실을 인지하지도 못한 채 계속해서 악성코드의 공격을 받고 있다.

정치적인 목적으로도 문서형 악성코드들이 많이 사용되고 있다. 우리나라의 정보가 타국가로 유출된다면 큰 문제를 야기할 수 있고, 국가 간 문제뿐만 아니라 개인정보유출, 랜섬웨어 등 여러 가지 위험한 문제점이 있다.

2020년 12월[1]에는 북한 정부 후원 해킹조직인 “금성121”이 HWP 문서로 APT(Advanced Persistent Threat) 공격을 한 사례가 있다. “참가신청서”라는 문서에 내용을 작성하려면 클릭을 해야 하는데, 사각형 투명 오브젝트가 전체화면 크기로 들어가 있는 상태에서 사용자가 클릭을 하도록 유도하는 것이다. 클릭할 경우 악성코드가 실행되는데 xor 인코딩 키로 악성코드를 숨겨 백신 프로그램에서 탐지되지 않는다. 2021년 7월[2]에는 무역 배송 관련 표로 위장한 엑셀이 파일 내부에 악성코드를 숨기고 사용자가 “콘텐츠 사용”을 누르면 악성코드에 감염되게 만들었다. 2021년 8월[2]에는 인도네시아어로 작성된 이메일에 파워포인트 형식(.ppam)의 파일을 첨부해 “매크로 포함”을 누르면 악성코드에 감염되게 만들었다. 2021년 9월[2]에는 법률 동의서라는 제목의 doc 파일이 메일로 유포되었고 이 또한 “콘텐츠 사용”을 누르면 악성코드에 감염되는 프로그램이었다. 2021년 10월[3]에는 코로나19 긴급재난지원금 신청 관련 HWP 문서에 악성 스크립트를 삽입한 사례도 있다.

일상에서 많이 사용하는 문서나 프로그램에 악성코드를 삽입하는 경우가 많기 때문에 출처가 분명하지 않은 파일은 실행하지 않고, 신뢰할 수 있는 사람이나 기관일 경우에도 메일 주소를 한 번 더 확인해보는 등 보안 수칙을 지켜야 한다[2].

본 논문에서는 이러한 문제들을 해결하기 위해 문서형 악성코드의 구조를 분석했다. OLE(Object Linking and Embedding) 문서 구조는 파일과 폴더 형식과 유사한데, 그 중 파일 부분에 해당하는 스트림을 추출해 냈고, 스트림 별로 키워드를 추출하여 기계학습의 feature로 사용했다. 문서 구조를 분석하면서 키워드를 직접 확인하였고 악성행위와 직접적으로 연관되는 키워드들을 정리했으며 크리티컬 키워드로 정의했다. 악성행위와 직접적으로 연관되지는 않지만 악성파일에서 반복적으로 등장하거나 키워드가 두 개 이상 결합하여 등장할 때도 크리티컬 조건으로 추가했다.

Microsoft Office 문서 파일에서 파일 당 추출한 키워드의 등장 횟수를 CSV(Comma-Separated Values) 파일로 정리하여 데이터셋을 구축했다. SVM(Support Vector Machine), naïve Bayes, logistic regression, random forest 등의 기계학습 모델을 만들었다.

2장에서는 문서형 악성코드에 관한 선행 연구를 조사했다. 3장에서는 Microsoft Office 파일의 구조, OLE와 압축 포맷 그리고 추출한 키워드에 대한 설명을 포함했다. 4장에서는 데이터 수집 방법, 키워드별 통계, 기계학습 모델별 성능을 정리했다.

II. Preliminaries

1. Related works

문서형 악성코드는 취약점을 이용하지 않고 난독화가 되어 쉽게 판별하기 어렵다. 이러한 악성코드는 사용자가 인식하지 못하고 감염이 되기 때문에 기계학습을 이용한 정확한 판단이 필요하다. 문서형 악성코드 분류를 통해 감염의 확산을 감소시키기 위해 기계학습으로 여러 가지 유형의 파일을 탐지하는 방법에 대한 연구가 진행되었다.

Jafar 등[4]의 연구는 머신러닝이 인간과 같이 상호 작용, 수행된 작업등을 학습 후 다음 과제에서도 수행할 수 있고, 문제의 복잡성과 적응성의 필요성을 이해하기 위해 지속적으로 학습과 경험을 기반으로 컴퓨터가 인간의 개입 없이 작업을 정교하게 수행할 수 있는 머신러닝이 필요하다고 연구했다.

Batta Mahesh[5]의 연구는 머신러닝이 데이터 문제를 해결하기 위해 다양한 알고리즘으로 존재하고, 사용되는 알고리즘의 종류는 해결하려는 문제의 종류, 변수의 수에 따라 적합한 모델의 종류가 달라진다고 연구했고 다양한 알고리즘에 대해 연구했다.

Kang 등[6]의 연구는 PDF 문서의 사용량의 비례함에 따라 증가하는 문서형 악성코드가 PDF의 구조, javascript 콘텐츠, 기능들에 포함되어 있는 것을 연구했고, 학습 모델을 만들어 정상파일과 악성파일을 분류하는데 random forest가 가장 좋은 결과를 보였다고 연구했다.

Jeong 등[7]과 Woo 등[8]의 연구는 우리나라에서 많이

사용되는 HWP의 악성코드에 대해 연구했다. 탐지를 위해 새로운 CNN(Convolutional Neural Network) 모델을 구축하고, 스트림을 입력받아 malware 여부를 판단했다. 성능 평가로는 spap 레이어와 stretch padding 방법을 사용할 때 다른 모델에 비해 효율적임을 연구했다.

Kang 등[9]의 연구에서는 PDF(Portable Document Format) 악성코드 탐지를 위해 악성파일과 정상파일을 비교 분석하여 탐지 및 분류해주는 합성곱 신경망을 구축했다. PDF 파일로 연구했지만, 스트림이 포함되어 있다면 다른 파일 구조에서도 적용할 수 있다고 판별했다.

Cho 와 Lee의[10] 연구에서는 악성코드를 Microsoft Office 문서의 미할당 영역에 삽입하거나 매크로 기능에 실행코드를 삽입하는 형태로 사용하기 때문에 메타데이터와 파일 포맷 구조를 분석해 악성코드가 존재할 수 있는 모든 영역을 확인하고 분석함으로써 악성파일과 정상파일을 분류하는 기법을 연구했다. 기존에 있던 900가지 특징으로 연구했으며 VBA 매크로 기능은 MS Visio, AutoCAD 등 여러 가지 프로그램에서도 사용되기 때문에 다른 파일 포맷에도 적용할 수 있다고 연구했다.

Kim 등[11]의 연구에서는 PDF, HWP 등 문서 파일을 이용하는 사회공학적 공격은 웹, SNS 등에 정치나 사회적 문제로 업로드 하는 파일이나 직장 동료로 가장한 스팸 메일 전송 등으로 행해지고 있고, 이러한 위협일 경우 자동으로 악성코드를 분석하고 분류해주는 시스템을 연구했다. 정적분석은 스크립트 및 shell 코드를 추출하고 동적 분석은 로그를 생성하는 결론을 내었다. 샘플에서는 뛰어난 성능을 보였지만 정확도를 더 높이기 위해서는 새로운 샘플들이 다수 필요하다고 제안했다.

Song 등[12]의 연구에서는 사이버 공격은 공격자들이 방법을 바꾸어가며 우회하기 때문에 예전의 방식으로는 식별하기 어렵다고 했다. 따라서 인공 지능 기술을 도입하였고 악성 powershell 스크립트를 탐지하기 위해 스크립트를 정적으로 분석했다. 토큰, AST(Abstract Syntax Tree) 기반의 메서드로 데이터를 스케일링 했다. 토큰을 추출해 냈고 자주 나타나는 유형과 특정 행동을 취할 수 있는 토큰을 조합하여 비교 실험한 결과 탐지 율은 98% 이상이었으며 5-token 3-gram, AST 3-gram에서 가장 좋은 결과가 나왔고 각각 ML과 DL 모델이라고 연구했다.

Balal 등[13]의 연구에서는 매크로 기반 malware는 문서의 내부에 내장되어 컴퓨터를 감염시키며, 악성파일은 일반적으로 이메일을 통해 전송된다고 연구했다. 또한 실행 파일이 아닌 텍스트 파일 형식으로 기존의 백신 프로그램들이 바이러스를 감지할 수 없다고 했다. 이 연구에서는 공격을 받은 후 피해자가 직면할 문제와 감염을 완화시킬 수 있는 방법에 대해 연구했다. Word, Excel, PowerPoint와 같은 Microsoft Office 문서에 내장되어 있는 악성코드는 탐지할 수 있지만 PDF, jpeg 등의 다른 파일 포맷에서는 탐지할 수 없었다.

Kim 등[14]의 연구에서는 malware 작성자가 피싱 공격이나 안티바이러스 탐지를 피하기 위해 난독화를 한다고 연구했고 그중 VBA 매크로와 관련된 난독화에 대해 연구했다. 난독화된 매크로 코드 검출 방법과 기계학습을 통한 분류로 다른 연구와 비교하여 23% 이상 향상되었으며, VBA 매크로를 분석한 결과 악성파일은 98.4%가 난독화되어 있었고, 정상파일은 1.7%만 난독화가 되어있었다고 연구했다.

Mikus 와 Nicholas[15]의 연구에서는 API를 사용하여 OLE파일을 추출할 수 있고 알고리즘은 헤어 블록을 읽는다. 헤더에서 OLE문서의 FAT(File Allocation Table)를 작성할 수 있다. FAT 구문 분석이 되면 디렉터리 목록을 조사한 다음 각 항목을 읽어 애플리케이션별 정보를 추출한다. Chicago프로젝트를 위해 만들어진 ole-dump라는 프로그램으로 출력했다.

Kim 등[16]은 의사결정 트리에 훈련 샘플 및 특징 선택 등 배깅을 기반으로 한 결정 트리 결합 모델인 random forest 알고리즘을 사용하여 3~5%의 성능 향상을 보였다.

Lee 등[17]의 연구에서는 생체 신호 데이터를 이용해 저혈압 발생 환자를 예측하는 방법에 대해 연구했는데, 네 가지 피쳐 셋을 분류해 피쳐에 따른 random forest 정확도를 연구했고, 가장 좋은 정확도로 75%의 높은 정확도를 보였다.

다른 연구들에서 언급을 잘 하지 않았던 키워드 관련한 부분도 본 논문에서 다뤄볼 예정이다.

III. The Proposed Scheme

3.1 Microsoft Office structure analysis

Microsoft Office 파일은 복합 파일 이진 구조(Compound File Binary Format)인 97-2003 버전과 OOXML(Open Office XML) 구조인 2007 버전 이상으로 나누어진다[10].

본 논문에서는 복합 파일 이진 구조를 대상으로 연구를 진행했다. Microsoft에서 제작한 복합 파일 이진 구조는 OLE라고도 불리며, 하나의 작은 파일 시스템과 같은 구조를 지니고 있다. 내부에서 파일과 폴더의 개념을 OLE 파

일 포맷에서는 각각 storage와 stream이라고 부른다. 파일과 폴더의 개념을 가지고 있고 상/하위 버전에 대해 뛰어난 호환성을 가지고 있다. 호환성이 좋아서 문서 열람은 가능하지만, 상위 버전의 기능을 하위 버전에서 실행할 경우 기능을 사용하지 못한다.

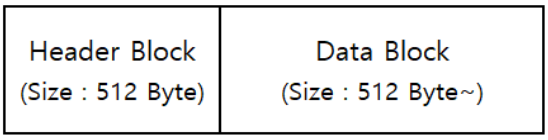


Fig. 1. OLE File Structure

OLE 파일은 크게 Fig.1처럼 헤더 블록과 데이터 블록으로 나누어진다. 헤더 블록은 512byte 크기이며, 데이터 블록은 512byte 이상을 가진다.

헤더 블록은 OLE 파일 전체의 주요 정보들을 가지고 있으며, 데이터 블록은 주로 property, stream data, BBAT(Big Block Allocation Table), SBAT(Small Block Allocation Table)등 네 가지 정보를 갖는다.

Property는 드라이브의 파일이나 폴더에 대한 정보를 지닌다. Stream data는 OLE 파일에서 가장 큰 비중을 차지하는 데이터이다. BBAT는 OLE 내부의 스트림 위치 정보를 포함하는 링크 형태의 구조로 OLE 파일이 커질수록 증가한다. SBAT는 문서를 입력할 때 작은 영역의 데이터를 저장한다. 이 중에서 데이터 블록의 대부분은 stream data가 차지하고 있다.

OLE 헤더 블록은 OLE 파일의 주요 정보들을 담고 있는 블록이다. OLE 파일을 열어 512byte 만큼 읽으면 헤더 블록을 읽은 것이 되는데 제일 첫 번째 블록은 -1 블록이다. Fig.2는 데이터 블록의 예시이며 -1 블록이 헤더 블록이고 0에서 n까지의 블록이 데이터 블록이다.

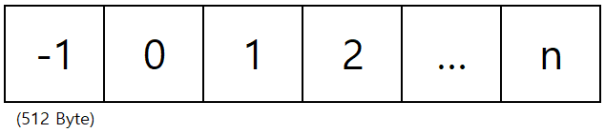


Fig. 2. OLE Header Block

Fig.2는 데이터 블록의 예시이며 -1 블록이 헤더 블록이고 0에서 n까지 블록이 데이터 블록이다.

헤더 블록을 읽었을 때 Fig.3에서 붉은색 박스로 표시한 부분이 magic id로 위치는 0x00에서 0x07까지 8byte이다. D0 CF 11 E0 A1 B1 1A E1로 정상적인 OLE 파일임을 나타내는 signature이다.

3.2 Ms office feature extraction

OLE 파일의 구조 분석을 통해 354개의 feature를 추출했다. Table 1은 중요 키워드를 정리한 것으로 파일 기본 정보를 나타내는 키워드 354개 중 3개, 중요 키워드 354개 중 78개, 중요 크리티컬 키워드 36개로 키워드의 활용 용도와 치명적인 정도에 따라 정리했다.

Microsoft Office는 문서 작성 시 반복적인 작업을 효율적으로 할 수 있게 하는 VBA(Visual Basic for Application) 매크로 기능을 제공한다. VBA 매크로 기능으로 파일의 생성과 삭제, 실행이 가능하다.

또한 VBA 매크로 기능으로 레지스트리, 사용자 프로필, 암호, 보안 관련 시스템 파일에도 접근이 가능하기 때문에 문서형 악성코드에서 많이 활용되고 있다. 실제로 Microsoft Office 문서형 악성코드를 분석한 결과 악의적인 행위를 수행하는 기능을 VBA 코드 내부에 삽입한 것을 확인할 수 있다.

대표적으로 외부 url에 접속하여 파일을 다운로드 하는 Urlmon, UrlDownloadToFile 등 url 관련 악성코드, 윈도우 시스템에 접속하여 비정상 행위를 하는 root, userprofile, password 등 시스템 관련 악성코드, 쉘코드, kernel32.dll을 사용하기 위한 각종 Windows API 함수인 CreateThread, VirtualAlloc, RtLmoveMemory 등을 포함하는 악성코드가 발견되었다. 이를 주요 크리티컬 키워드로 정리하였고 정상파일과 악성파일을 분석하여 잘 알려진 악성코드는 아니지만 악성파일에서 반복적으로 나타나는 키워드들 또한 크리티컬 키워드로 정리했다.

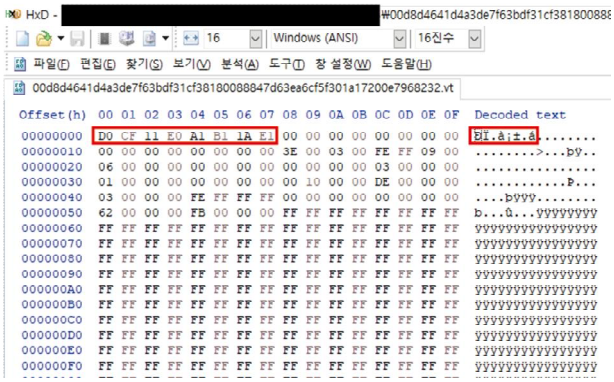


Fig. 3. OLE Header Structure

Table 1. Keywords in OLE Files

File basic information	filename, size, stream
Important keywords	xlfn, PNG, IHDR, RESDLL, ssbB, sheet, Visual, Basic, Object, Library, Color, ColorMode, ThisDocument, Win16, Win32, Win64, MSWORD, Windows, http, exe, Source, sleep, application, Gloval, Temp, xml, PPT, Arkusz, VBA6, VBE6, Workbook, Encode64, AutoOpen, shell32, urlmon, regsvr32, urldownload, password, kernel32, documentopen, lyskovick, shellexecute, hkey_local, open, binary, copyfile, shell, createthread, xmlhttp, kill, createtextfile, writetext, savetofile, mkdir, shellrun, end, Environ, var, root, Delete, Control, Print, xmlPK, threadid, Attributes, StackSize, Runtime, Token, manifest, assembly, Exit, Unit, bax, hashSize, blockSize, Algorithm, auto_close, lookup
Important critcal keywords	binary, filecopy, shell, shellrun, shellexecute, powershell, shell32, savetofile, createtextfile, createobject, createthread, mkdir, urlmon, urldownloadtofilea, urldownload, arkusz, documentopen, regsvr32, vba, vbe, lyskovick, runtime, autoclose, autoopen, password, kernel32, popupkiller, reset, silent, keylogger, token, encode64, root, userprofile, www, bax

IV. Experiment Results

4.1 Dataset

본 연구에서는 공공데이터포털에서 수집한 3,000개의 OLE 구조 정상파일과 국내 백신사에서 제공받은 10,000개의 악성파일 중 7,830개의 OLE 구조 악성파일을 추출했고, 7,830개의 악성파일 중 임의의 7,000개의 파일을 사용했다. 총 정상파일 3,000개, 악성파일 7,000개로 10,000개의 파일로 연구를 진행했다. 10,000개의 파일 중 70%인 7,000개를 학습에 사용하였고 30%인 3,000개의 파일을 테스트에 사용했다.

4.2 Feature statistics

악성파일 7,000개를 Table2와 같이 키워드별로 빈도를 구해 CSV 파일로 정리했다. Open, Environ, ED, GD, User, Document, VBA, Common, VBA7, VBE7, DLL, Visual, For, tlb, OLE, binary, Filecopy, Regsvr32, Shell32, Shellexecute, Auto_open, Label, lyskovick, UrlDownloadToFile, Mydecode, Accent, Downloadfile, Encode64, Password 등의 키워드들이 악성파일에서 많이 등장하는 것을 볼 수 있었다.

Table 2. OLE Keywords Statistics

file name	mal.	open	C:	VBA	Crea
0011b	1	3	3	4	3
0015c	1	3	3	4	8
00282	1	0	0	1	0
...	...	...	...	...	...
e9e0d	1	3	3	2	2
e9ec4	1	1	2	1	1
e9f83	1	0	3	4	5

4.3 Training algorithm

추출한 키워드 리스트와 크리티컬 키워드 리스트를 활용하여 머신러닝을 진행했다. 대표적인 기계학습 알고리즘인 SVM, naïve Bayes, logistic regression, random forest로 학습 모델을 만들고 성능을 비교했다.

SVM은 패턴 인식과 자료 분석을 위한 학습 모델이다. 주로 회귀 분석과 분류에 사용된다. SVM은 주어진 데이터 집합을 바탕으로 새로운 데이터가 들어왔을 때 비확률적 이진 선형 분류 모델을 만든다. SVM의 장점으로는 범주나 수치 예측 문제에 사용 가능하고, 오류 데이터가 있더라도 영향을 받지 않는다. 또한 과적합의 경우가 적고 신경망보다 사용하기가 쉽다는 장점이 있지만 학습 속도가 느리고, 블랙박스 형태로 되어 있어 해석이 어렵다는 단점이 있다[18][19].

Naïve Bayes는 feature와 label을 사용해 지도학습을 진행한다. 새로운 데이터가 들어왔을 때 학습된 머신러닝 모델은 feature를 기반으로 label을 판단한다. 장점으로는 간단하고 computation cost가 작아 빠르며, 큰 데이터셋에 적합하다. 단점으로는 feature 간에 독립성이 있어야 하지만 실제 데이터셋에서 feature 간 독립되어 있는 경우는 매우 드물기 때문에 큰 단점이라고 볼 수 있다[18][20].

Logistic regression은 회귀를 사용하여 데이터가 어떤 범주에 속할 확률을 0과 1사이의 값으로 예측한다. 확률이 더 높은 범주에 속하는 것으로 분류하는 지도학습 모델이다[21].

Random forest는 분류와 회귀 분석에 사용되는 앙상블 학습 방법이다. 다수의 결정 트리를 학습하고 결과를 종합하여 예측한다. 각 트리를 학습시킬 때 배깅(bagging) 방식을 사용하는데, 이 배깅 방식은 예측 모델의 안정성을 향상시켜준다. 장점으로는 아주 높은 정확도를 가지고 있고 data scaling을 할 필요가 없으며 과적합이 잘 되지 않는다는 장점이 있다. 단점으로는 훈련 시 메모리 소모가 크고, train data를 추가해도 성능 개선이 어렵다는 단점이 있다[22].

Table 3. Confusion Matrix

SVM		predicted	
		malware	normal
actual	malware	879	21
	normal	4	2096
naïve Bayes		predicted	
		malware	normal
actual	malware	860	40
	normal	1010	1090
logistic regression		predicted	
		malware	normal
actual	malware	889	11
	normal	4	2096
random forest		predicted	
		malware	normal
actual	malware	891	9
	normal	4	2096

Table 3은 예측 값(predictive value)과 실제 값(actual values)을 통해 training 예측 성능을 측정할 수 있는 confusion matrix이다.

Confusion matrix의 TP(True Positives, 참의 값을 참이라고 예측한 경우)와 FP(False Positives, 거짓 값을 참이라고 예측한 경우)를 확인해 봤을 때 random forest가 가장 좋은 성능을 보여주고 있다.

Table 4에서 알고리즘에 따른 accuracy, precision, recall, F1-score를 확인해 봤을 때 random forest의 accuracy가 0.996, precision이 0.996, recall이 0.998, F1-score가 0.997로 가장 좋은 성능을 보였다.

Table 4. Algorithm Performance

algorithm	acc.	pre.	recall	f1-score
SVM	0.992	0.990	0.998	0.994
naïve Bayes	0.650	0.965	0.519	0.675
logistic regression	0.995	0.995	0.998	0.996
random forest	0.996	0.996	0.998	0.997

Jeong 등[7]의 연구는 534개의 HWP 파일을 분석한 결과 spap 레이어의 F1-score가 92.86%으로 위양성과 위음성 사이의 적절한 모델임을 연구했는데, 본 연구에서는 random forest의 F1-score가 0.997로 향상되었음을 보였고, 정확도 또한 향상되었음을 보였다.

Random forest 같은 트리 형태의 모델은 특질을 파악해 우회할 수 있는데 이러한 우회를 막기 위해 SVM 같은 black box형태의 알고리즘도 중요하다. Table 4에서 보이듯 SVM 또한 정확도 0.992, F1-score 0.994로 좋은 성능임을 보여주고 있다.

4.4 Feature importance

Fig.4에서 random forest 학습에 주요하게 사용된 변수를 시각화했다.

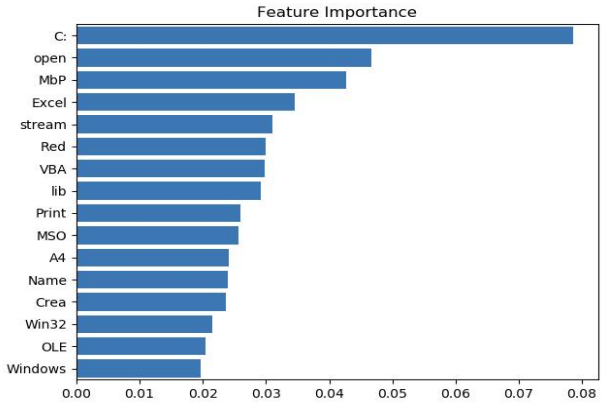


Fig. 4. Feature Importance

Feature importance에서는 Print, Temp 등 악성코드와 크게 관련이 없는 단어도 나왔지만 open, C:, VBA, Crea, lib, Win32 등 크리티컬 키워드가 많이 포함되어 있는 것을 알 수 있다.

V. Conclusions

문서형 악성코드는 문서 내부에 삽입되어 관공서나 국가 기관 홈페이지, 의심하지 않는 지인들이 전송하는 메일 등 백신 프로그램이 탐지하지 못하는 영역에서의 공격이 계속 되고 있다. 이런 악성코드가 포함된 파일은 주로 Microsoft Office 97-2003 버전의 복합 파일 이진 구조에서 발견되고 있으며, 사용자의 편리성을 높여주기 위한 기능인 VBA 매크로 기능에 외부 프로그램 실행 코드나 외부 url에서 특정 파일을 다운받는 기능으로 삽입되어 있었다.

본 논문에서는 OLE 파일 구조와 스트림에 포함되어있는 데이터를 분석하고 악성파일에서 354개의 키워드를 추출했다. 키워드는 파일 기본 정보를 나타내는 세 개의 키워드와 자주 등장했던 주요 키워드, 주요 키워드 중 치명적인 코드를 분리하여 크리티컬 키워드 리스트를 만들었다.

크리티컬 키워드를 기반으로 정상파일과 악성파일을 판별해주는 기계학습을 진행하였고, 기계학습의 대표적 알고리즘인 SVM, naïve Bayes, logistic regression, random forest를 사용하여 악성코드 여부를 탐지했다. 탐지 결과 decision tree 방식을 사용하는 random forest가 좋은 성능을 보였으며, random forest뿐만 아니라 SVM, logistic regression에서도 좋은 성능을 보여주었다. 성능을 평가할 수 있는 confusion matrix를 확인해 봤을 때도 세 가지 알고리즘 모두 좋은 결과가 나온 것을 확인해 볼 수 있었다.

향후 연구에서는 인터넷 크롤링과 국내 백신사를 통해 더 많은 샘플을 수집하여 최신 악성코드를 분석할 예정이다. 또한, 최신 MS Office 포맷인 OOXML의 구조를 분석하여 OLE뿐만 아니라 OOXML 포맷 파일도 지원할 것이다. 파일 스캔에서부터 문서 구조 분석, 악성 파일 탐지 및 파일 격리 기능을 제공하는 MS Office 악성코드 탐지 시스템을 구축할 것이다.

ACKNOWLEDGEMENT

This research was supported by Seoul R&BD Program(CY210066).

REFERENCES

[1] Gil Min-kwon, “In December, HWP OLE-based APT Attack by the Geumseong121”, *https://www.dailysecu.com/news/articleView.html?idxno=118508*, Dailysecu, Dec. 2020.

[2] Jonghyun Lee, “Ahnlab Urges Caution against Malicious Code Distributed as Document Files”, *https://www.ddaily.co.kr/news/article/?no=221749*, Digital Daily, Sep. 2021.

[3] Jonghyun Lee, “Disguised Document File for Hacking Related to Disaster Aid”, *https://www.ddaily.co.kr/news/article/?no=223783*, Digital Daily, Oct. 2021.

[4] Jafar Alzubi, Anand Nayyar and Akshi Kumar, “Machine Learning from Theory to Algorithms an Overview”, Journal of Physics: Conference Series, Vol. 1142, Dec. 2018. DOI:10.1088/1742-6596/1142/1/012012

[5] Batta Mahesh, “Machine Learning Algorithms - A Review”, International Journal of Science and Research, Vol. 9, Issue 1, Jan. 2019. DOI: 10.21275/ART20203995

[6] Ah Reum Kang, Young-seob Jeong, Se Lyeong Kim and Jiyoung Woo, “Malicious PDF Detection Model against Adversarial Attack Built from Benign PDF Containing Javascript”, Applied Sciences, Vol. 9, No. 22, pp. 4764, Nov. 2019. DOI: 10.3390/app9224764

[7] Young-Seob Jeong, Jiyoung Woo, SangMin Lee and Ah Reum Kang, “Malware Detection of Hangul Word Processor Files Using Spatial Pyramid Average Pooling”, Sensors, Vol. 20, No. 18, pp. 5265, Sep. 2020. DOI:10.3390/s20185265

[8] Young-Seob Jeong, Jiyoung Woo and Ah Reum Kang, “Malware Detection on Byte Streams of Hangul qord Processor Files”, Applied Sciences, Vol. 9, No. 23, pp. 5178, Nov. 2019. DOI: 10.3390/app9235178

[9] Young-Seob Jeong, Jiyoung Woo and Ah Reum Kang, “Malware Detection on Byte Streams of PDF Files Using Convolutional Neural Networks”, Hindawi Security and Communication Networks, 2019, April 2019. DOI:10.1155/2019/8485365

[10] Sung Hye Cho and Sang Jin Lee, “A Research of Anomaly Detection Method in MS Office Document”, KIPS Transactions on Computer and Communication Systems, Vol. 6, No. 2, pp. 87-94, Feb. 2017. DOI:10.3745/KTCCS.2017.6.2.87

[11] Hong-Koo Kang, Ji-Sang Kim, Byung-Ik Kim and Hyun-Cheol Jeong, “Development of an Automatic Document Malware Analysis System”, IT Convergence and Security Lecture Notes in Electrical Engineering, pp. 3-11, 2012. DOI: 10.1007/978-94-007-5860-5_1

[12] Jihyeon Song, Jungtae Kim, Sunoh Choi, Jonghyun Kim, Ikkyun Kim, “Evaluations of AI-based Malicious PowerShell Detection with Feature Optimizations”, ETRI Journal Wiley, Vol. 43, No. 3, pp. 549-560, Nov. 2020. DOI: 10.4218/etrij.2020-0215

[13] Balal Sohail, Ma’en Tayseer Ekrayem Alrashd, Yaseein Soubhi Hussein, Mohammad Tubishat, Shounak Ghosh, Ahmed Saeed Alabed, “Macro based Malware Detection System”, Turkish Journal of Computer and Mathematics Education, Vol. 12, No. 3, pp. 5776-5787, April 2021. DOI:10.17762/turcomat.v12i3.2254

[14] Sangwoo Kim, Seokmyung Hong, Jaesang Oh and Heejo Lee, “Obfuscated VBA Macro Detection Using Machine Learning”, IEEE/IFIP International Conference on Dependable Systems and Networks, pp. 490-501, 2018. DOI:10.1109/DSN.2018.00057

[15] Mikus and Nicholas, “An Analysis of Disc Carving Techniques”, Naval Postgraduate School Monterey CA Dept of Computer Science, March 2005.

[16] Jae Hyup Kim, Hyn Ki Kim, Kyung Hyun Jang, Jong Min Lee and Young Shik Moon, "Object Classification Method Using Dynamic Random Forests and Genetic Optimization", Journal of The Korea Society of Computer and Information, Vol. 21, No.

5, pp. 79-89, May. 2016. DOI:10.9708/jksci.2016.21.5.079

[17] Ji Hyun Lee, Ah Reum Kang, Sang Hyun Kim and Ji Young Woo, "Multi-Cutting Machine for TJ Coupler Production", Proceedings of the Korean Society of Computer Information Conference, Vol. 27, No. 1, Jan. 2019.

[18] Byengha Choi, Kyungsan Cho, “Comparison of HMM and SVM Schemes in Detecting Mobile Botnet”, The Korea Society of Computer and Information, Vol. 19, No. 4, pp. 81-90, April 2014. DOI:10.9708/jksci.2014.19.4.081

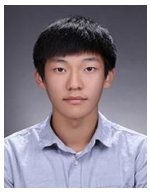
[19] Aleksander Kolcz, “Local Sparsity Control for Naive Bayes with Extreme Misclassification Costs”, Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 128-137, Aug. 2005.

[20] Yeong-Hwil Ahn, Koo-Rack Park, Dong-Hyun Kim and Do-Yeon Kim, “A Study on the Development of Product Planning Prediction Model Using Logistic Regression Algorithm”, The Korea Convergence Society, Vol. 12, No. 9, pp. 39-47, Sep. 2021. DOI:10.15207/JKCS.2021.12.9.039

[21] Pan Jun Kim, “An Analytical Study on Automatic Classification of Domestic Uournal Articles Using Random Forest”, Journal of the Korean Society for Information Management, Vol. 36, No. 2, pp. 57-77, 2019. DOI:10.3743/KOSIM.2019.36.2.057

[22] Moon Kwon Kim, Seung Ho Han, Hyun Jung La and Soo Dong Kim, “Design of Effective Inference Methods for Supporting Various Medical Analytics Schemes”, The Korean Institute of Information Scientists and Engineers, Vol. 42, No. 1, pp. 1102-1104, June 2015.

Authors



Hyeong Kyu Choi is currently pursuing the B.S. in Pai Chai University, Korea, in 2022. His research interest includes algorithm, computer security, machine learning.

Ah Reum Kang received the M.S. and Ph.D. degrees in Information Security from Korea University, Korea, in 2012 and 2016, respectively. She is a currently assistant professor at the Department of Information Security, Pai Chai University. From 2016 to 2018, she was a researcher at the Department of Computer Science and Engineering at the University at Buffalo, State University of New York, USA. She was a research professor at SCH Convergence Science Laboratory, Soonchunhyang University from 2019 to 2020. Her research interest includes computer security, privacy-preserving data mining and data science in general.