국민대학교
소프트웨어융합대학
소프트웨어학부

캡스톤 디자인 I
종합설계 프로젝트

프로젝트 명	DREAM(Detecting in Real-timE mAlicious document using Machine Learning)
팀 명	*Do it!*
문서 제목	결과보고서

Version	1.4
Date	2019-MAY-26

팀원	문다민 (조장)
	김기환
	김현석
	정혜리
	방유한
지도 교수	윤명근 교수님

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 및 소프트웨어학부 개설 교과목 캡스톤 디자인**I** 수강 학생 중 프로젝트 **DREAM**을 수행하는 팀 **Do it!**의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 **Do it!**의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역

Filename	8조_결과보고서.docx
원안작성자	문다민
수정작업자	문다민 김현석 김기환 정혜리 방유한

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2019-05-16	문다민	1.0	초안 작성	초안 작성
2019-05-18	김현석	1.1	내용 추가	개발 내용 추가
2019-05-20	김기환	1.2	내용 추가	자기 평가 추가
2019-05-22	정혜리	1.3	내용 추가	부록 추가
2019-05-24	방유한	1.3.1	내용 수정	프로젝트 개요 수정
2019-05-26	문다민	1.4	최종 수정	최종 수정

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

목 차

1 개요 5
1.1 프로젝트 개요 5
1.2 추진 배경 및 필요성 7
1.2.1 최근 악성코드 현황 8
1.2.2 현재 기술 시장 현황 10
1.2.3 현재 기술 시장의 문제점 및 개선 방향 13
1.2.3.1. 기술 시장 문제 1 13
1.2.3.2. 기술 시장 문제 2 14
1.2.3.3. 기술 시장 문제 3 14
2 개발 내용 및 결과물 15
2.1 목표 15
2.2 연구/개발 내용 및 결과물 15
2.2.1 연구/개발 내용 15
2.2.1.1. 데이터 수집 15
2.2.1.2. 데이터 라벨링 16
2.2.1.3. 특징 추출(Feature Extraction) 및 기계 학습 17
2.2.1.3.1. PDF 17
2.2.1.3.2. MS Word 21
2.2.1.3.3. 딥 러닝(Deep Learning) 23
2.2.1.3.4. 문서형 악성코드 탐지 엔진 26
2.2.1.3.5. 웹 26
2.2.1.3.5.1. 시연용 웹 27
2.2.1.3.6. 데이터 공유 웹 30
2.2.2 시스템 기능 요구사항 32
2.2.3 시스템 비기능(품질) 요구사항 36
2.2.4 시스템 구조 및 설계도 38
2.2.4.1. 유즈 케이스(Use Case) 39
2.2.4.2. 시퀀스 다이어그램(Sequence Diagram) 42
2.2.5 활용/개발된 기술 44
2.2.6 현실적 제한 요소 및 그 해결 방안 45
2.2.6.1. 현실적 제한 요소 45
2.2.6.2. 해결 방안 46
2.2.7 결과물 목록 46

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

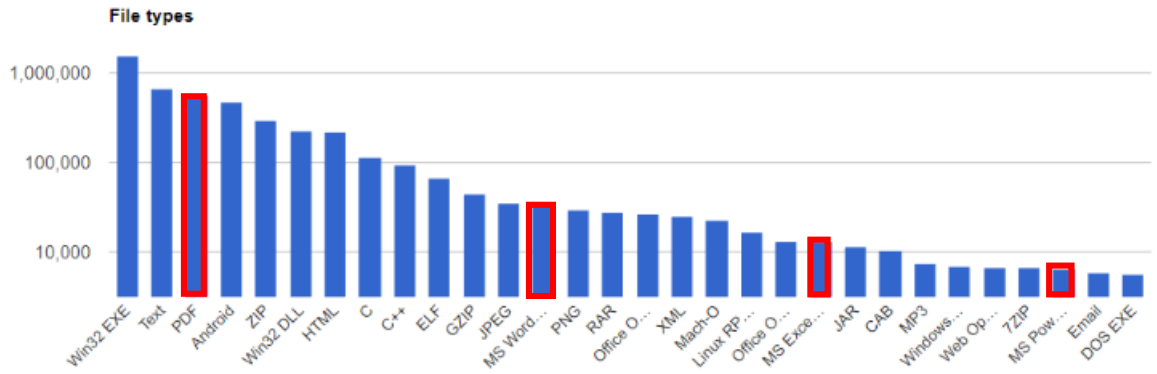
2.3 기대효과 및 활용방안 46
2.3.1 기대효과 46
2.3.2 활용방안 46
3 자기평가 47
4 참고 문헌 47
5 부록 49
5.1 설치 매뉴얼 49
5.2 사용자 매뉴얼 49
5.3 테스트 케이스 51

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

1 개요

1.1 프로젝트 개요

정보화 시대에 들어서며 전 세계적으로 악성코드의 수는 급격히 늘어나고 있다. 최근에는 악성코드 중 문서형 악성코드의 수가 증가하고 있고, 특히 이 문서형 악성코드의 유포 방법이 화두가 되고 있다. 대표적으로 2018년 1월부터 등장한 갠드크랩(GandCrab) 랜섬웨어는 사람들의 이목을 끌 수 있는 문서 파일로 위장하여 유포된다. 그리고 2018년 3월에 전 세계적으로 등장한 시그마(Sigma) 랜섬웨어는 이력서로 위장하여 유포된다. 이처럼 문서형 악성코드는 점점 지능적이고 정교하게 발전하고 있다.



<그림 1> 일주일 동안 바이러스토탈에 유입되는 파일의 종류와 수
(출처 = https://www.virustotal.com/en/statistics/)

바이러스 토탈(Virus Total)은 의심스러운 파일 및 URL을 분석하고 모든 종류의 악성 코드를 탐지하는 서비스이다. <그림 1>은 일주일 동안 유입된 파일의 유형별 수에 대한 바이러스 토탈의 그래프이다. 그래프에 따르면 PDF는 55만 개로 3번째를 차지하고 있으며 이 외에도 MS Word, MS Excel 등 우리가 자주 사용하는 문서형 파일이 속해있다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

한 기사에 따르면 작년 10월, '국가 핵심 인력 등록 관리제 등 검토 요청.hwp'라는 파일명으로 문서형 악성코드가 유포되어 국내에서 실제 감염 피해가 발생한 바 있다. 바이러스 토탈에 이 문서형 악성코드가 처음 업로드된 날짜는 10월 23일이었지만 10월 24일에 57개의 안티바이러스 중 3개 안티바이러스만이 탐지하였으며 이어 25일에는 6개 안티바이러스, 26일에는 13~16개 안티바이러스, 11월 2일에는 56개의 안티바이러스 중 절반가량인 25개 안티바이러스에서만 탐지되었다. 즉 상당수의 안티바이러스가 이 문서형 악성코드를 탐지하지 못하였다.

<그림 2> 상당수의 안티바이러스가 문서형 악성코드를 탐지 못하는 사례
(출처= https://www.boannews.com/media/view.asp?idx=75093)

문서형 악성코드로 인한 사회적 피해가 지속해서 발생하고 있지만, 문서형 악성코드를 전문적으로 탐지하는 안티바이러스는 많지 않다. 이는 문서형 악성코드가 쉽게 유포될 수 있어 인터넷 사용자들의 PC 에 감염되어 사회적 문제가 발생할 수 있다.

본 프로젝트에서는 문서형 악성코드를 탐지할 수 있는 엔진을 제작하여 문서형 악성코드의 유포와 그로 인한 피해가 생기는 것을 막고자 한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

1.2 추진 배경 및 필요성

최근 해커들의 주요 공격 중 하나는 사회 공학(Social Engineering)적 공격이다. 사회 공학적 공격은 시스템이 아닌 사람의 취약점을 공략하는 공격이다. 대상자의 성향, 동향, 추세 등을 파악하여 정보를 수집하고 그 정보를 바탕으로 정부 기관이나 회사 또는 지인으로 속여 대상자의 흥미를 유발할 수 있는 키워드로 내용을 작성한다. 해커들은 메일, SMS, 웹 게시물 등에 이러한 내용에 악성코드가 삽입된 문서형 악성코드를 첨부하여 대상자 또는 불특정 다수가 의심 없이 첨부파일을 실행하게 유도하는 것이 특징이다.

<그림 3> 이메일을 활용한 문서형 악성코드 유포 사례

(출처= http://www.seoul.co.kr/news/newsView.php?id=20190107800022)

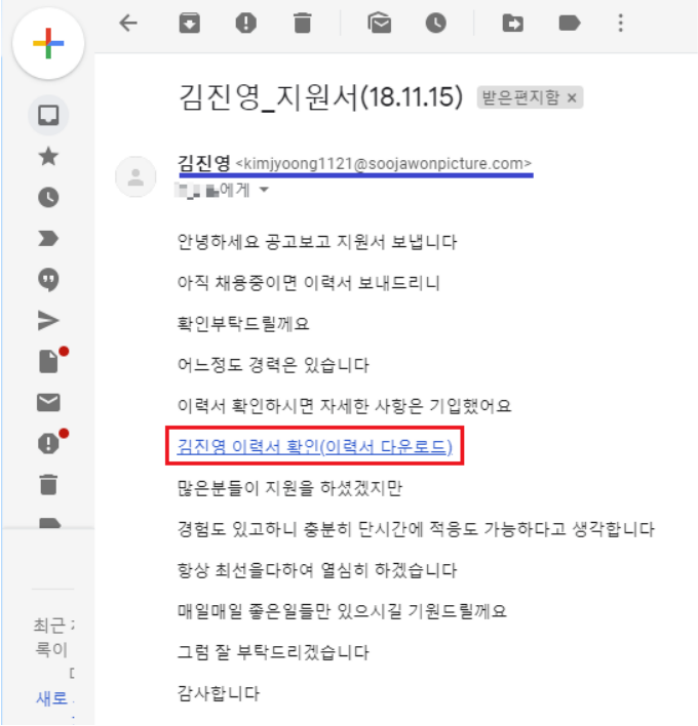
한 가지 사례로 2019 년 1 월, 통일부를 출입했던 언론사 취재기자들에게 통일부로 사칭하여 일괄적으로 ‘TF 참고.zip’라는 제목의 메일이 배포됐다. 메일의 내용은 ‘TF 참고되시길 ~, 언론사별 브랜드 관련해서 관리 잘해주시고~. 비번은 “tf”’라며 첨부된 문서형 악성코드의 실행을 유도하는 문구가 포함되어 있었다. 압축파일 안에는 pdf 파일과 hwp 파일이 있었으며 개인정보를 수집하고 해킹을 시도하려는 문서형 악성코드가 발견됐다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

1.2.1 최근 악성코드 현황

1) 갠드크랩(GandCrab) 랜섬웨어

2018년 11월 15일 이력서를 가장한 갠드크랩 랜섬웨어는. 이메일을 통해 유포되고 있다. 아래 <그림 4>는 11월 15일에 유포된 이메일의 내용이다.



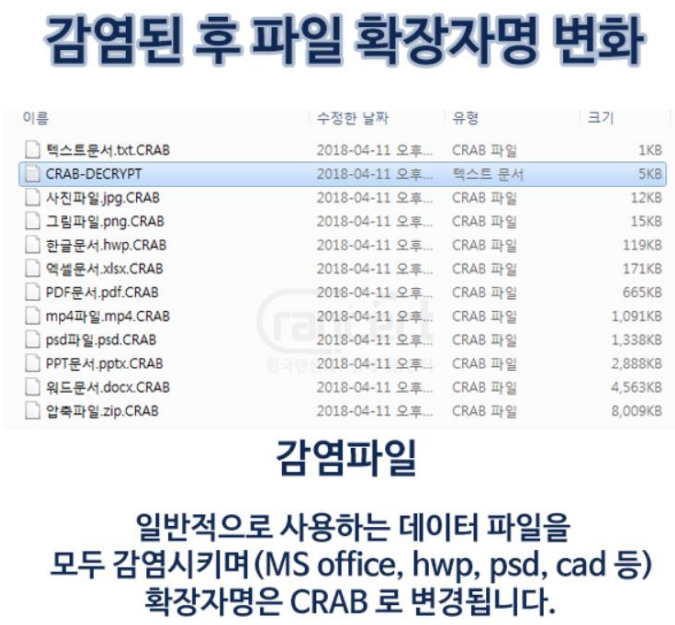
<그림 4> 이력서를 가장한 갠드크랩 랜섬웨어

출처= https://www.rancert.com/bbs/bbs.php?bbs_id=case&mode=view&id=92

메일을 보낸 사람은 "kimjyoong1121@soojawonpicture.com"으로 되어있으며, 실제 서비스되는 도메인이 아닌 것으로 확인되었다.

위와 같은 이력서로 위장된 갠드크랩은 피해자가 실행해보도록 사회 공학적 기법을 사용하여 접근한다. 아래 <그림 5>는 갠드크랩에 감염된 상황이다.

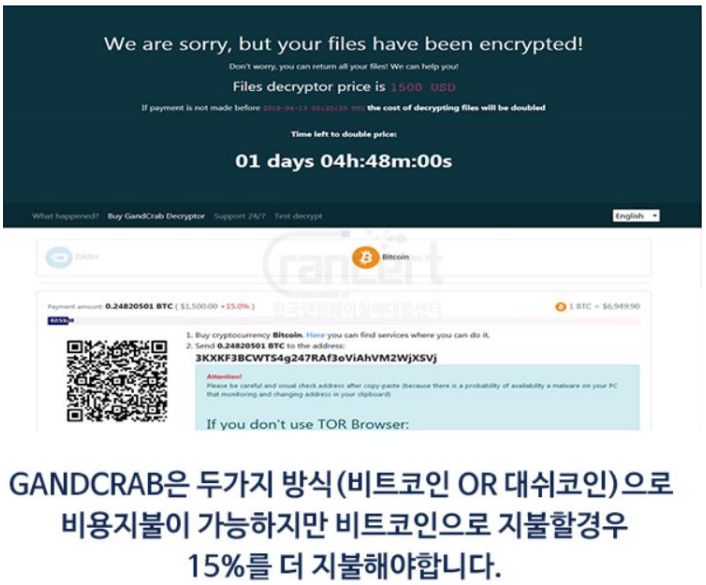
All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.



<그림 5> 갠드크랩 랜섬웨어 감염 예시

출처= https://www.rancert.com/bbs/bbs.php?bbs_id=case&mode=view&id=92

아래의 <그림 6>과 같이 감염 시 금전을 요구하게 된다. 갠드크랩 랜섬웨어뿐만 아니라 여러 문서형 악성코드들이 사회적 문제를 발생시키고 있다.



<그림 6> 랜섬웨어 감염 시 금전요구 화면

출처= https://www.rancert.com/bbs/bbs.php?bbs_id=case&mode=view&id=92

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

1.2.2 현재 기술 시장 현황

1) 안랩(AhnLab) MDS

<그림 7> 안랩 MDS 장비

(출처 = https://www.ahnlab.com/kr/site/product/productView.do?prodSeq=68)

안랩(AhnLab)은 한국 정보 보안 업체 중 하나로, 안티바이러스인 V3 로 잘 알려져 있다. 안랩은 V3 외에도 다른 소프트웨어 및 하드웨어 보안 솔루션, 모바일 보안, 정보보안 컨설팅, 기타 산업용 제품 보안 등 다양한 분야에서 보안 사업을 하고 있다.

안랩의 보안 솔루션 MDS 는 네트워크 샌드박스 및 전용 에이전트를 통해 다양한 경로를 통해 유입되는 위협을 신속하게 수집하며, 시그니처 기반, 평판 기반, 비 시그니처(signature-less) 기반, 동적 행위 분석 등 멀티 엔진을 기반으로 기존 방식의 위협(Known), 알려지지 않은(Unknown) 신 · 변종 위협까지 정확하고 효율적으로 탐지 및 대응한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2) 시만텍(Symantec)

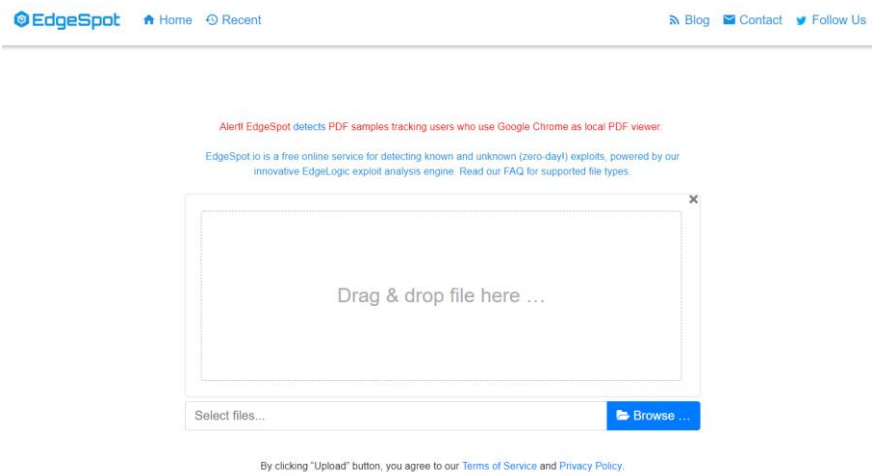
시만텍은 미국의 보안 소프트웨어 회사이다. 시만텍이 개발하고 배포하는 제품인 노턴 안티바이러스는 악성 코드 방지 및 제거 기능을 제공한다. 또한 스팸 메일 필터링과 피싱 보호 기능이 있으며 2007 년에 바이러스 검사 소프트웨어 시장에서 61%를 차지했다.

또한 시만텍은 시그니처의 한계를 제시하며 Symantec Endpoint Protection 14 를 개발하고 있다. Symantec Endpoint Protection 14 는 시그니처에 의존하지 않고 기계 학습(Machine Learning) 및 행위 분석을 통해 보안 효과를 극대화하고 오탐을 최소화한다.

<그림 8> 시만텍 로고

(출처 = https://www.symantec.com/)

3) EdgeSpot



<그림 9> EdgeSpot 화면

(출처 = https://edgespot.io/)

Edge Spot은 알려지거나 알려지지 않은 (Zero Day) 공격에 대해 탐지 기능을 제공하는 무료 온라인 웹 서비스이다. PDF, Microsoft Office 파일 등 문서형 파일을 업로드하면 탐지 결과를 총 4 가지(Malicious, Suspicious, Information, No threat found)로 분류하여 사용자에게 보여준다. Edge Spot은 정적 분석 및 동적 분석, 기계 학습과 같은 기술을 사용하여 분석한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

4) 지란지교시큐리티 SaniTOX

<그림 10> 지란지교시큐리티 SaniTOX

(출처 = https://www.jiransecurity.com/products/sanitox)

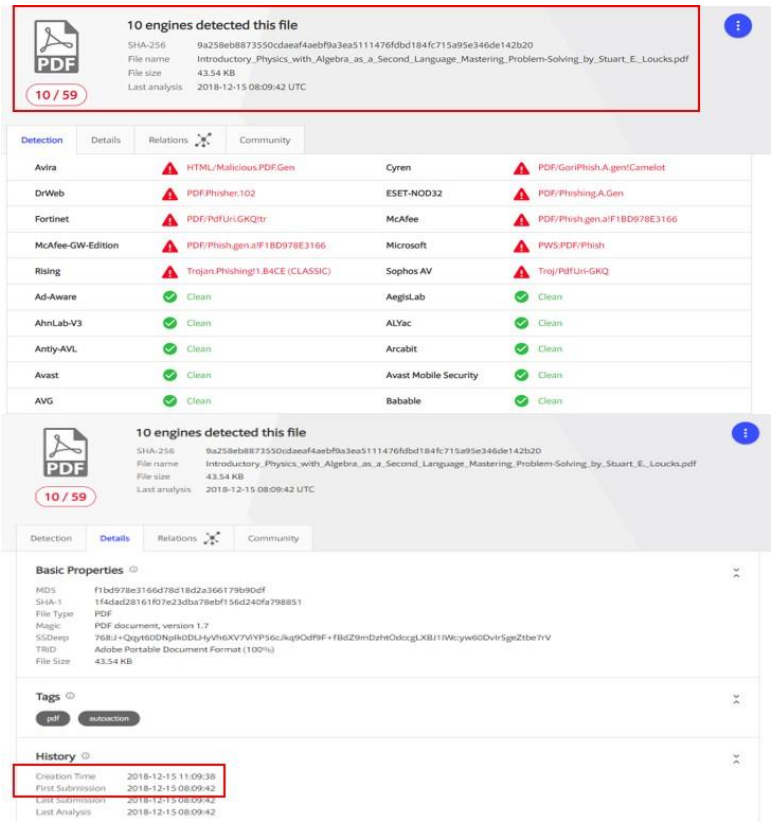
지란지교시큐리티는 문서보안, 암호화폐 보안과 모바일보안 등 소프트웨어 보안 전문 기업이다. 최근에 지란지교시큐리티가 자체 개발한 CDR (Content Disarm & Reconstruction) 기술을 이용하여 콘텐츠 악성코드를 무해화하는 새니톡스(SaniTox) 솔루션을 공개하였다. CDR 은 파일 내 잠재적 보안 위협 요소를 탐지하여 제거한 뒤에 안전한 파일로 재조합하여 악성코드 감염 위험을 사전에 방지할 수 있는 기술이다.

새니톡스 솔루션은 2 가지의 형태로 제공이 된다. 첫 번째, 새니톡스 어플라이언스는 별도의 소프트웨어 설치나 설정이 필요 없는 일체형 장비로 쉽게 도입이 가능하다. 그리고 Content Prevention Engine(Anti-Virus + CDR) 기반의 알려진 위협에 대한 1 차 필터링과 문서 기반의 표적형 악성코드에 대한 2 차 예방적 보안을 통해 전방위 위협에 대응한다. 두 번째, 새니톡스 SDK 는 소프트웨어 개발사 및 서비스 제공 기업이 새니톡스 CDR 엔진을 자체 소프트웨어, 하드웨어 혹은 서비스에 통합할 수 있도록 API 를 제공한다. 다양한 콘텐츠 유입 채널이 있는 제품을 통해 콘텐츠 악성코드 무해화 기능을 제공할 수 있다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

1.2.3 현재 기술 시장의 문제점 및 개선 방향

1.2.3.1. 기술 시장 문제 1



<그림 9> 바이러스토탈의 안티바이러스 별 문서형 악성코드 탐지 결과 및 파일 정보

<그림 9>는 2018년 12월에 등장했던 문서형 악성코드를 바이러스 토탈에 업로드한 화면이다. 바이러스 토탈 결과 59곳의 안티바이러스 중 오직 10곳의 안티바이러스가 악성이라고 탐지했다. 최신 문서형 악성코드를 탐지하는 안티바이러스가 적다는 것을 알 수 있다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

1.2.3.2. 기술 시장 문제 2

A 사의 솔루션은 문서 파일 내 실행 가능한 액티브 콘텐츠(매크로, 자바스크립트 등)를 원천 제거하여 문서 파일이 어떠한 동적 행위를 할 수 없는 파일로 만들기 때문에 악성 행위를 일절 차단할 수 있는 장점이 있다. 하지만 문서가 정상 파일 일지라도 문서 내에 존재하는 액티브 콘텐츠를 일절 제거하기 때문에 사용자들은 정상적으로 문서를 사용할 수 없게 된다.

본 프로젝트에서는 파일의 구조를 확인 후 정적 분석 과정으로 특징을 추출 후 기계 학습 기법으로 학습한 모델로 악성 코드를 탐지하기 때문에 위 솔루션에서의 정상 파일까지 변환되는 문제가 해결된다.

1.2.3.3. 기술 시장 문제 3

B사는 파일 탐지 주요 기술로 동적 행위 분석을 사용하는데 이때 많은 시간과 비용이 발생한다. 하루에 분석할 수 있는 데이터의 양의 한계가 있으며 대용량의 데이터를 처리하는 데 어려움이 있다. 또한 많은 시간과 비용이 발생하기 때문에 일반 사용자나 가정에서는 사용하기 어려운 단점이 있다. 본 프로젝트는 정적 분석 기반의 기계 학습 기법을 사용하여 데이터를 탐지하기 때문에 많은 양의 데이터를 대응하는데 유리하다.

따라서 본 프로젝트는 PDF나 MS Office 등의 문서형 파일이 악성 파일인지 탐지하는 엔진을 제작하여 기존 기술을 보완하고, 문서형 악성코드가 유포되는 것을 방지하고자 한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2 개발 내용 및 결과물

2.1 목표

본 프로젝트는 문서형 악성코드를 탐지하는 엔진을 개발하고자 한다. 엔진은 파일이 유포될 수 있는 서버, 메일 서버 등의 서비스와 쉽게 연동될 수 있도록 개발한다. 이를 통해 서비스에서 문서형 악성코드를 탐지할 수 있게 함으로써 문서형 악성코드 유포를 방지하는 것을 목표로 한다. 또한 오픈소스 소프트웨어로 개발하여 여러 개발자와 엔진을 보완해가며 엔진을 발전시키고자 한다.

또한 엔진 사용자와 문서형 악성코드 연구 종사자에게 문서형 악성코드를 공유할 수 있는 웹 사이트를 개발한다. 데이터를 공유함으로써 문서형 악성코드 관련 연구에 기여하고 본 프로젝트에서 개발한 엔진의 발전을 이루고자 한다.

2.2 연구/개발 내용 및 결과물

2.2.1 연구/개발 내용

2.2.1.1. 데이터 수집

기계 학습 기반의 문서형 악성코드 탐지 모델을 제작하기 위해 문서형 악성코드 데이터가 필요하다. 필요에 따라 문서형 악성코드 데이터를 공유하는 사이트에서 데이터를 수집하는 크롤러를 개발했다. 데이터를 수집한 사이트 목록은 다음 표와 같다.

<표 1> 악성 데이터를 수집할 수 있는 사이트 목록

출처 = https://chogar.blog.me/80212372093

사이트 이름	사이트 주소
Contagiodump	http://contagiodump.blogspot.kr
Kernelmode	http://www.kernelmode.info/forum/viewforum.php?f=16
Malshare	http://malshare.com
AVCaesar	http://avcaesar.malware.lu
Malwareblacklist	http://www.malwareblacklist.com/showMDL.php
Malwr	https://malwr.com
Minotaur	http://minotauranalysis.com/exetweet
Openmalware	http://openmalware.org
Secuboxlabs	http://secuboxlabs.fr
Virusign	http://www.virusign.com
Virusshare	http://virusshare.com
Google	http://www.google.com

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.1.2. 데이터 라벨링

2018년 AV-TEST 성능 평가에서 수상한 3개의 안티바이러스를 선정하여 해당 안티바이러스의 탐지 결과를 데이터 라벨링에 사용하였다. 선정한 3개의 안티바이러스는 에프시큐어(F-Secure), 카스퍼스키(Kaspersky), 시만텍(Symantec)이다. 이 3개의 안티바이러스는 AV-TEST 성능 평가에서 일반 사용자 부문, 기업용 부문 모두에서 수상한 안티바이러스다.

<그림 11> AV-TEST 성능 부문 평가에서 수상한 안티바이러스

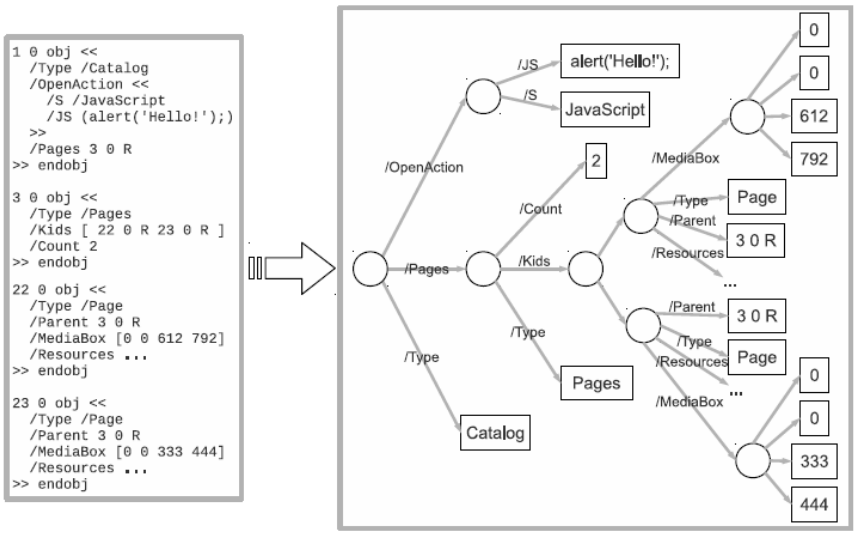
출처 = https://www.av-test.org/en/news/av-test-awards-this-is-the-elite-class-of-it-security-2018/

바이러스 토탈에서 라벨링 할 파일의 결과 리포트를 바이러스 토탈 API를 사용해서 받은 다음, 위에서 선정한 3개의 안티바이러스 중 하나 이상의 안티바이러스가 악성이라고 한다면 악성으로, 바이러스 토탈에 등록된 모든 안티바이러스가 정상으로 라벨링 하였다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.1.3. 특징 추출(Feature Extraction) 및 기계 학습

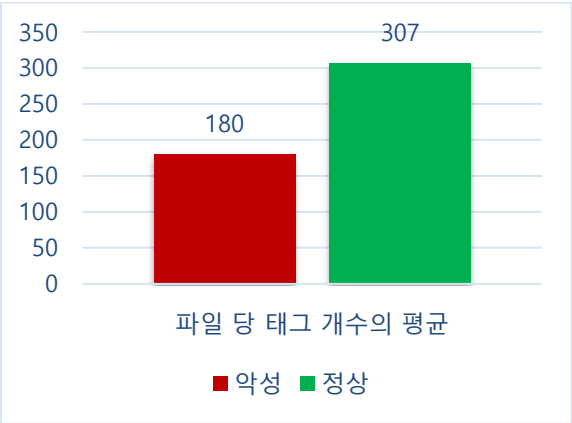
2.2.1.3.1. PDF



<그림 12> PDF 구조; 물리적인 구조(왼쪽), 논리적인 구조(오른쪽)

출처 = Nedim ˇSrndi´c and Pavel Laskov. Detection of Malicious PDF Files Based on Hierarchical Document Structure. In *Proceedings of the Network and Distributed System Security Symposium, NDSS 2013*

PDF 구조는 <그림 12> 와 같이, /Type, /Page 같은 다양한 태그(Tag)를 갖는 구조로 이루어져 있다. 실험 과정에서 악성 PDF가 갖는 문서 구조와 정상 PDF가 갖는 문서 구조가 조금 다르다는 것을 확인하였다. <그래프 1>과 같이 악성 PDF 문서 구조는 평균 180개의 태그 종류로 이루어져 있고, 정상 PDF 문서 구조는 평균 307개의 태그 종류로 이루어져 있다는 것을 확인했다.



<그래프 1> 파일 당 등장하는 태그 개수의 평균

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

그리고 <표 2>를 통해서 악성 PDF와 정상 PDF에서 다른 특징을 확인할 수 있다. <표 2>는 자주 등장하는 태그의 상위 16개를 나타낸 표이다.

<표 2> 악성 PDF 태그 정보(왼쪽), 정상 PDF 태그 정보(오른쪽)

Tag Name	Count	Tag Name	Count
/URI	8,204,328	obj	16,470,754
endobj	5,930,353	endobj	16,462,865
obj	5,226,287	/Type	7,882,414
/Type	5,054,637	/Length	6,351,320
/www	4,407,113	stream	6,342,069
/A	4,405,322	endstream	6,280,863
/Subtype	4,355,685	/Filter	5,687,772
/S	4,355,685	/FlateDecode	5,091,152
/Contents	4,348,565	/Subtype	4,090,938
/I	4,112,098	/xmpG	3,645,833
/M	4,111,108	/S	3,579,119
/P	4,109,853	/rdf	3,579,119
/H	4,109,502	/Xobject	2,680,936
/F	4,109,445	/Font	2,370,243
/Rect	4,102,544	/P	2,243,650
/Border	4,102,524	/stEvt	2,206,310

이처럼 악성 PDF의 문서 구조 특징과 정상 PDF의 문서 구조 특징이 다른 양상을 보인다. 따라서 이 특징을 특징 벡터에 담아내고자 해싱 트릭(Hashing Trick) 기술을 사용하여 특징 벡터를 생성하여 악성 PDF를 탐지해보는 실험을 진행했다. 해싱 트릭을 사용하여 특징 벡터를 생성한다면 악성 특징 벡터와 정상 특징 벡터 간의 희소한 정도에서 차이가 생길 것이라 기대했다.

생성한 특징 벡터로 기계 학습을 진행했다. 사용한 알고리즘은 LR(Logistic Regression), SVM, KNN(k-Nearest Neighbor Algorithm), RF(Random Forest), XGBoost, LightGBM 이다. 여러 알고리즘으로 기계 학습을 진행하고 검증데이터에 대한 검증 결과로 알고리즘마다 성능을 확인해보았다. 검증에 사용한 성능 지표는 다음과 같다.

1. 정확도(Accuracy)

   계산된 값이 실제 값과 얼마나 가까운지 나타내는 척도이다.

$$\text{Accuracy} = \frac{True\ \ Positive + True\ \ Negative}{True\ \ Positive + True\ \ Negative + False\ \ Positive + False\ \ Negative}$$

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2. F1-Score

F1-Score는 Recall과 Precision을 이용하여 조화 평균을 이용한 척도이다. F1 Score는 악성 데이터와 정상 데이터 간의 불균형을 이루었을 때 성능을 확인할 수 있는 지표이다.

$$\text{F1 Score} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

첫 번째 검증 데이터로는 바이러스 사인에서 수집한 데이터를 사용했다. 학습에 사용한 데이터는 바이러스 사인에서 11월, 12월에 수집된 악성 약 7만 5천 개, 정상 PDF 약 7만 개이고 검증에 사용한 데이터는 바이러스 사인에서 1월에 수집된 악성 PDF 약 2만 5천 개, 정상 PDF 약 5만 개이다. 검증 결과는 <표 3>과 같다.

<표 3> 검증 결과 1

	SVM	KNN	RF	XGB	LGB
Accuracy	0.89	0.88	0.93	0.89	0.90
F1-Score	0.87	0.85	0.90	0.86	0.86

두 번째 검증 데이터로는 바이러스 사인과 바이러스 토탈에서 수집한 데이터를 사용했다. 학습에 사용한 데이터는 바이러스 사인에서 11월, 12월, 1월에 수집된 악성 PDF 약 10만 개, 정상 PDF 약 12만 개이고 검증에 사용한 데이터는 바이러스 토탈에서 2017년에 수집된 악성 PDF 약 1만 개와 자체 수집한 정상 PDF 약 1만 개이다. 결과는 <표 4>와 같다.

<표 4> 검증 결과 2

	SVM	KNN	RF	XGB	LGB
Accuracy	0.91	0.83	0.95	0.96	0.97
F1-Score	0.91	0.83	0.95	0.96	0.97

본 프로젝트에서 개발한 엔진의 문서형 악성코드 탐지 성능을 안티바이러스 중 안랩 안티바이러스와 비교해보았다. 테스트 샘플(Sample)은 검증 2에서 사용한 검증 데이터 중 악성 PDF 10,678개를 샘플로 사용했다. 결과는 <표 5>와 같이 안랩은 10,678개 중 3,487개를 탐지하였고 드림은 10,678개 중 10,141개를 탐지하였다.

<표 5> 안티바이러스와 드림 엔진과 탐지 결과 비교

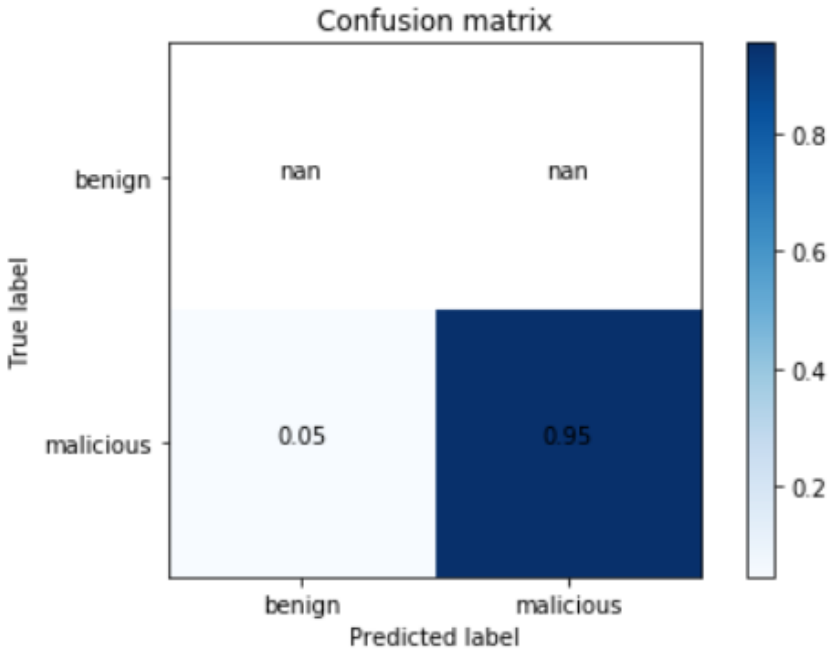
	탐지 / 전체 파일
안랩	3487 / 10678
드림 엔진	10141 / 10678

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

검사를 완료했습니다. [결과 상세 보기]를 클릭하여 감염 파일을 확인하고 악성코드를 치료하십시오.

검사 수: 10678 | 감염 수: 3487 | 치료 수: 0

<그림 13> 안랩 안티바이러스의 악성 PDF 탐지 결과



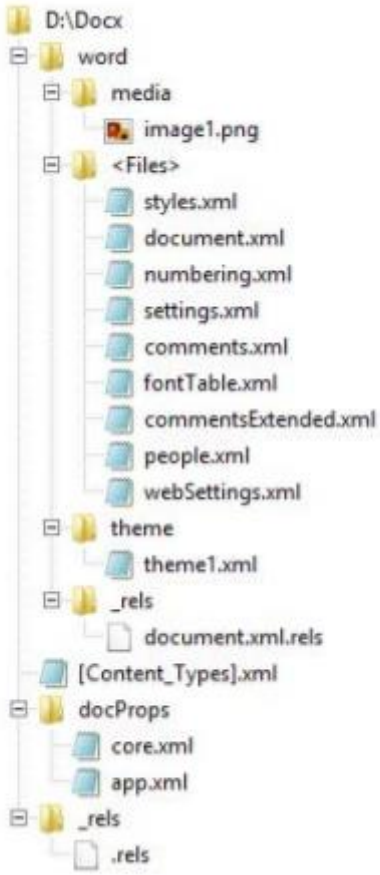
<그림 14> 드림 엔진의 악성 PDF 탐지 혼동 행렬

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.1.3.2. MS Word

1. DOCX

MS Word DOCX의 구조는 <그림 15>과 같다. <참고 논문 ALDOCX> 에 따르면 저자는 DOCX 내부에 있는 파일 경로를 사용하여 악성 파일과 정상 파일을 구분할 수 있다고 주장한다. 그 근거로, 악성코드 제작자가 문서에 악성 매크로나 콘텐츠를 추가하면 새로운 경로나 파일이 추가되어 정상 문서의 구조와 차이가 생긴다고 한다. 따라서 이 점을 특징으로 사용하여 DOCX 파일의 악성 여부를 탐지해보는 실험을 진행했다.



<그림 15> DOCX 내부 구조

출처=NissimN,CohenA,EloviciY. *ALDOCX: Detection of Unknown Malicious Microsoft Office Documents Using Designated Active Learning Methods Based on New Structural Feature Extraction Methodology*. IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 12, NO. 3, MARCH 2017,

특징 추출은 MS Word 파일의 내부 파일 경로를 경로 구분자로 토큰화(tokenization) 하고 DF(Document Frequency)를 사용하여 토큰별 DF 값을 구한 다음, 특징 추출을 진행하여 특징 벡

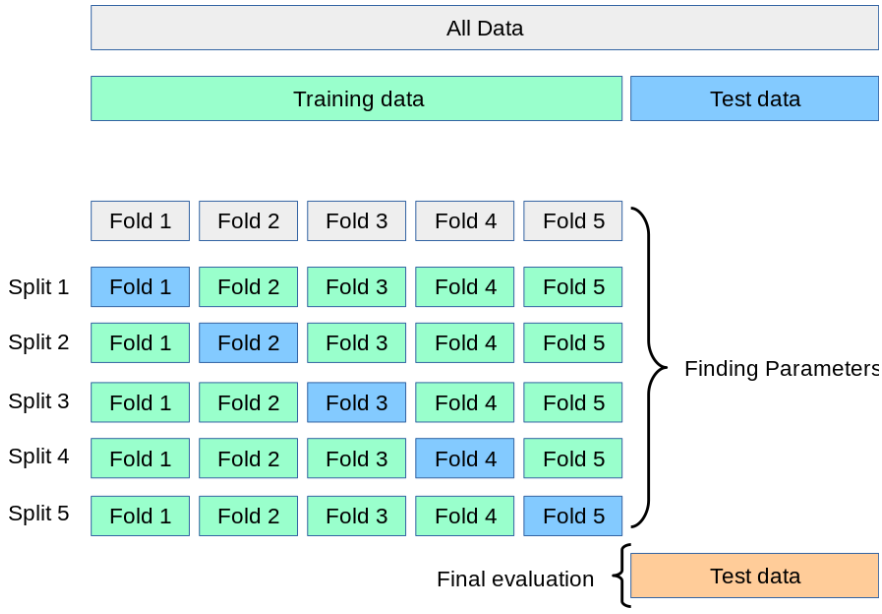
All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

터를 생성하였다. 예를 들어, 파일 내부 경로로 "word₩media₩image1.png" 이 주어진다면 ["word", "word₩media", "word₩media₩image1.png", ...] 로 토큰화하고 각 토큰의 DF 값을 구하여 그 값을 특징 벡터의 요소로 사용하였다. 그리고 논문에서 제시한 특징 외에도 DOCX 내부 각 파일의 엔트로피의 최댓값, 최솟값, 평균값 그리고 파일 크기의 최댓값, 최솟값, 평균값을 추가하여 특징 벡터를 생성하였다. 사용한 DF 값의 수식은 다음과 같다.

$$\mathrm{DF} = \frac{\text{단어가 나타난 문서 수}}{\text{전체 문서 수}}$$

생성한 특징 벡터로 악성 DOCX 파일을 탐지하는 실험을 진행해보았다. 학습에 사용한 데이터는 악성 162 개, 정상 4,882 개이다. 10 폴드 교차 검증(10-Fold Cross Validation)으로 성능을 확인했다.

K 폴드 교차 검증이란 K개의 폴드를 만들어서 진행하는 교차 검증이다. 데이터 셋이 적은 경우 테스트 셋에 대한 성능 평가의 신뢰성이 떨어지기 때문에 데이터를 K개의 폴드로 나누어 학습과 검증을 진행하는 방식이다.



<그림 16> 5 폴드 교차 검증

출처 = https://scikit-learn.org/stable/modules/cross_validation.html

<표 6> MS Word의 모델 별 10 폴드 교차 검증 정확도

	LR	SVM	NB	KNN	RF	XGBoost	LightGBM
기존 방법(정확도)	0.92	0.92	0.61	0.90	0.92	0.98	0.86
제안 방법(정확도)	0.93	0.93	0.33	0.95	0.96	1.00	0.98

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

<표 6>는 논문에서 제시한 방법으로 특징 벡터를 생성하여 검증한 결과와 본 프로젝트에서 연구 중인 방법으로 특징 벡터를 생성하여 검증을 진행한 결과를 비교한 표이다. 논문에서 제시한 방법보다 본 프로젝트에서 사용한 방법이 대부분의 학습 모델에서 향상된 결과를 보였다.

2.2.1.3.3. 딥 러닝(Deep Learning)

딥 러닝은 학습 과정에서 주어진 데이터를 분류하는데 주요한 특징을 추출하는 능력이 탁월한 것으로 알려져 있다. 따라서 파일의 바이트 시퀀스(Byte Sequence)를 학습 데이터로 딥 러닝에 제공하면 학습 과정에서 파일에서 악성 행위를 하는 부분을 특징으로 찾아낼 것이라 기대하여 실험을 진행해보았다.

딥러닝 모델을 구현하기에 앞서 "Malconv" 논문을 참고하였다. 참고한 논문 Malconv는 파일의 바이트 시퀀스를 사용하여 PE 악성 코드를 탐지하는 연구를 진행하였다. <그림 17>는 논문에서 제시한 모델 구조이다. 제시한 모델은 먼저 임베딩(Embedding)을 통해 입력 데이터인 바이트 시퀀스를 저차원으로 표현시키고, 저차원으로 변환된 데이터를 일반적인 CNN 모델과 같이 컨볼루션(Convolution), 풀링(Pooling) 연산 과정으로 데이터에서 악성 파일과 정상 파일의 특징을 추출하여 학습한다.

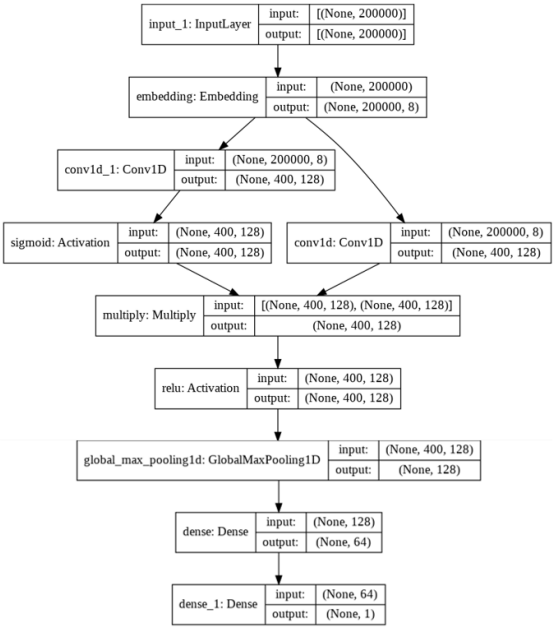
본 프로젝트는 논문 Malconv와 다르게 문서형 악성코드를 대상으로 사용하였다. 학습의 입력 데이터로 사용될 바이트 시퀀스 크기는 200KB로 제한하여 200KB 이상의 바이트는 잘라내거나, 바이트가 200KB보다 작을 경우 패딩 해주는 방식으로 일정한 크기의 특징 벡터를 형성하도록 했다.

입력 벡터 사이즈, 학습률, 옵티마이저(Optimizer) 등 여러 실험을 통해 하이퍼 파라미터(Hyper Parameter)를 수정해보았다. 사용한 하이퍼 파라미터는 <표 7>과 같다.

<표 7> 딥러닝 모델 구조 및 하이퍼파라미터

CNN 기반 특징 추출 과정		하이퍼 파라미터	
합성곱 연산 (1D, 1x128 필터)	512, 512	목적 함수	크로스 엔트로피
배치 정규화	사용 안함	에폭	10
활성 함수	ReLU, Sigmoid	학습률	0.001
풀링 연산	전역 최대 풀링	옵티마이저	Adam
ANN 기반 특징 분류기		배치 크기	64
은닉층 노드 수	64	드랍아웃	사용 안함

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.



<그림 17> 딥러닝 모델 구조

PDF, MS Word DOC 파일을 각각 딥 러닝으로 학습하여 기계 학습에서 했던 실험과 마찬가지로 실험에 대한 검증을 해보았다.

1) PDF

학습에 사용한 PDF 데이터는 악성 PDF 10 만 개, 정상 PDF 12 만 개이다. 사용한 검증 데이터는 <표 5>의 검증 결과에 사용했던 데이터와 같다. 결과는 <표 8>과 같다.

<표 8> 딥 러닝 PDF 검증 결과

	딥 러닝
정확도	0.87
F1-Score	0.84

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2) DOC

학습에 사용한 DOC 데이터는 악성 DOC 6 천 개, 정상 DOC 1 만 6 천 개이다. 사용한 검증 데이터는 바이러스 토탈에서 공유해준 악성 DOC 789 개, 자체 수집한 정상 300 개이다. 검증 결과는 <표 9> 와 같다.

<표 9> 딥 러닝 DOC 검증 결과

	딥 러닝
정확도	0.91
F1-Score	0.93

그리고 타 안티바이러스의 악성 탐지 결과와 탐지 성능을 비교해보았다. 결과는 <표 10>와 같다.

<표 10> 안티바이러스 탐지 결과 비교

	탐지 / 전체 파일
안랩	486 / 789
드림 엔진	702 / 789

<그림 18> 안랩 안티바이러스의 악성 DOC 탐지 결과

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.1.3.4. 문서형 악성코드 탐지 엔진

의심스러운 PDF와 MS Word 파일을 탐지할 수 있는 엔진을 제작했다. 사용자는 엔진 서버에서 제공하는 API를 사용하여 애플리케이션 또는 서비스와 엔진을 연동하여 사용할 수 있다.

지란지교 이상준 소장

<그림 19> 엔진 서버 구조

출처= https://towardsdatascience.com/how-to-do-rapid-prototyping-with-flask-uwsgi-nginx-and-docker-on-openshift-f0ef144033cb

엔진 서버는 파이썬 프레임워크 Flask를 사용하여 개발했고 uWSGI 와 Nginx를 추가하여 다른 애플리케이션과의 연결을 용이하게 하였다 uWSGI와 Nginx를 추가한 장점은 다음과 같다.

1. 가상 호스트(Virtual Host) 관련 문제 해결에 용이하다.
2. 설정 파일(Configuration)을 관리하는 데 편하다.
3. 정적 파일 서빙(Static File Serving)과 유저별 설정 파일 상세 설정이 용이하다.
4. 트래픽이 많아졌을 경우 필요에 따라 업 스케일링이 쉽다.

엔진 기능은 다음과 같다.

1. Start : 엔진을 작동한다.
2. Stop : 엔진을 끈다.
3. Submit : 엔진에 의심스러운 문서 파일을 업로드하여 검사한다.
4. Scan : 바이러스토탈에 의심스러운 파일을 신규 분석 요청한다.

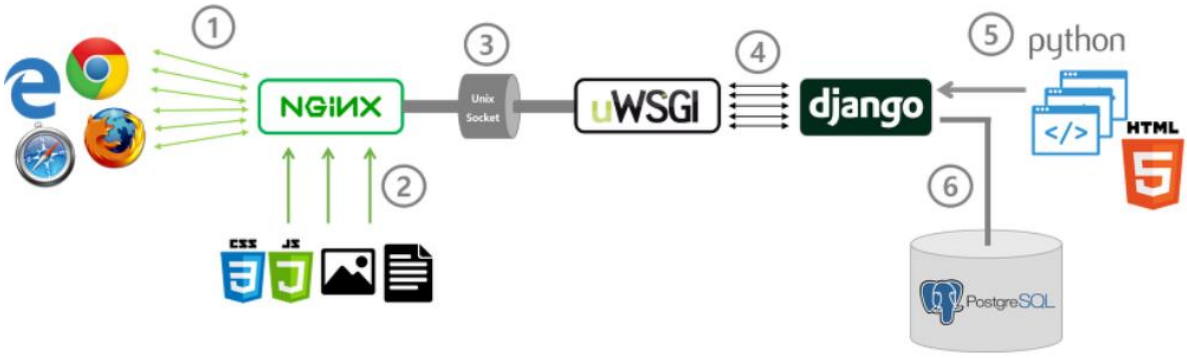
설치 방법과 사용 방법은 [부록]의 설치 매뉴얼과 사용 매뉴얼을 참고한다.

2.2.1.3.5. 웹

시연용 웹 사이트와 문서형 악성코드를 공유하는 웹 사이트를 개발하였다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

두 웹 사이트의 프론트엔드(Front-End)는 부트스트랩(Bootstrap) 프레임워크를 사용하여 개발하였고 사용자의 브라우저 크기에 따라 변하는 반응형 웹으로 제작하였다. 백엔드(Back-End)는 파이썬 프레임워크 Django 를 사용하여 개발하였고, 엔진과 마찬가지로 NGINX, uWSGI 를 서버와 연동하였다.



<그림 20> 서버와 NGINX, uWSGI 연동

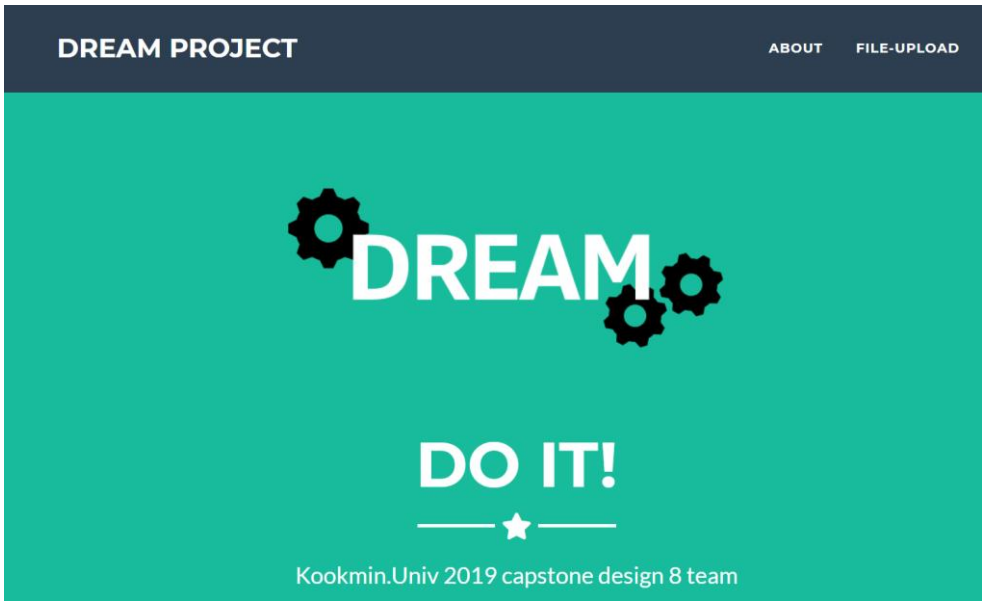
출처 = https://wayhome25.github.io/django/2018/03/03/django-deploy-02-nginx-wsgi/

2.2.1.3.5.1. 시연용 웹

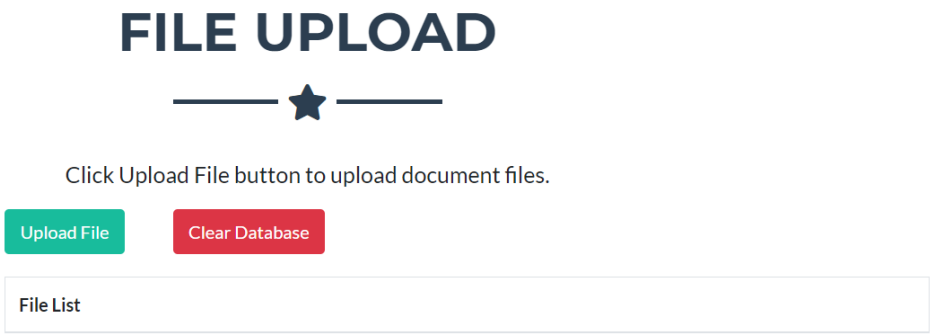
시연용 웹 사이트는 사용자가 웹 사이트에 문서형 악성코드를 올릴 수 있다는 시나리오를 가정하여, 파일 업로드 기능이 있는 웹 사이트로 개발한 후 엔진과 연동 하였다.

사용자는 클라이언트에서 파일 업로드 버튼을 통해 사용자가 업로드 하고자 하는 파일을 선택한다. 업로드 가능한 파일의 형식은 PDF, MS Word 와 같은 문서 파일이고, 파일 크기가 100MB 이하인 파일만 가능하다. 파일을 업로드 할 때 상태 바를 출력하여 파일이 업로드되는 경과를 보여준다. 클라이언트에서 서버로 파일의 업로드가 완료되면 해당 파일을 바로 저장하지 않고 파이썬 requests 모듈을 사용하여 엔진에 전달하게 된다. 엔진에서 업로드 할 파일의 악성 확률 결과를 도출하게 되면 엔진에서 결과를 받고 해당 파일의 악성 여부를 판단한 뒤 악성이면 업로드를 제한하고 정상일 경우 정상적으로 업로드하게 한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

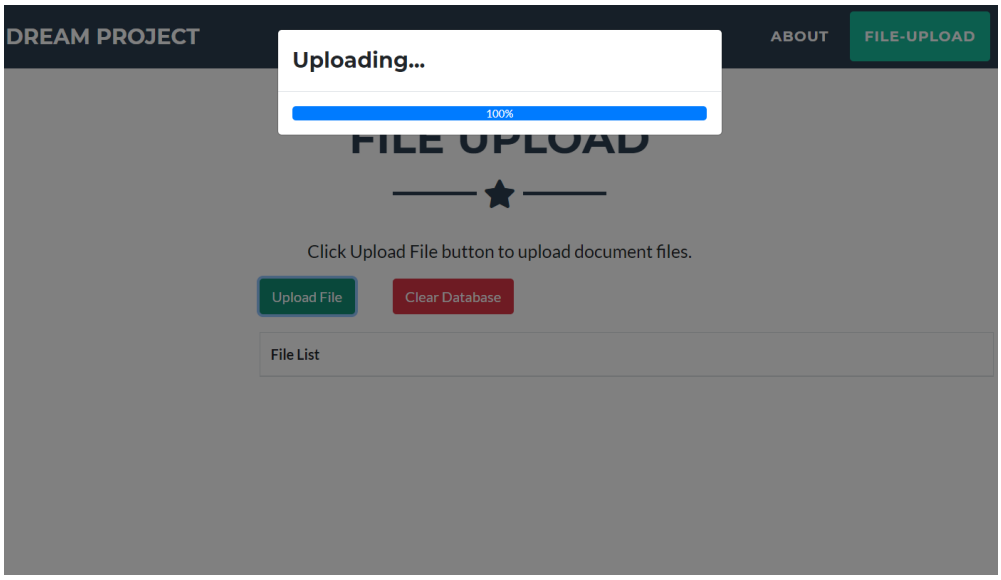


<그림 21> 시연용 웹 초기 화면

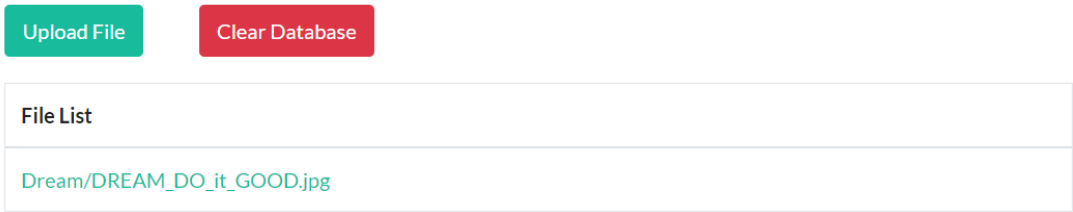


<그림 22> 파일을 업로드 할 수 있는 공간

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.



<그림 23> 파일을 업로드 할 때 생기는 상태 바



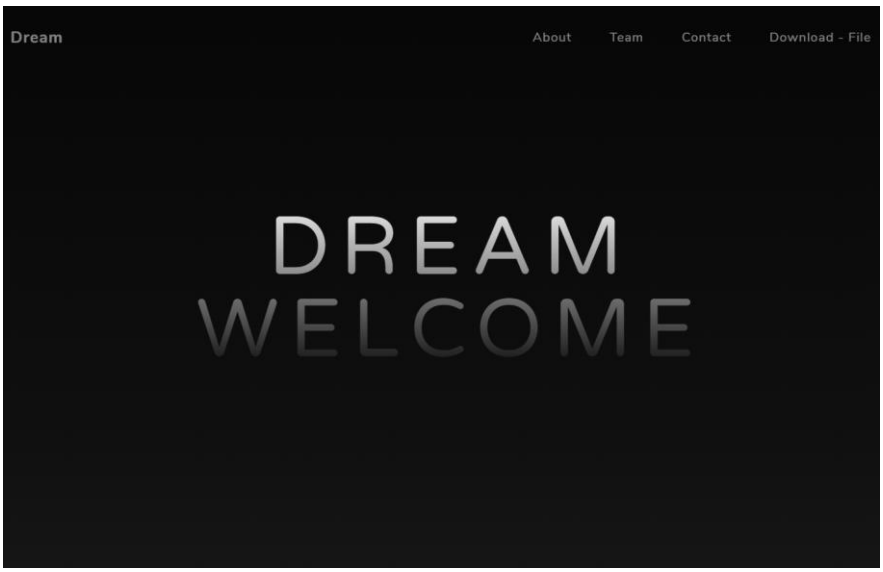
<그림 24> 파일이 정상적으로 업로드 되었을 때 화면

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.1.3.6. 데이터 공유 웹

본 프로젝트는 문서형 악성코드를 문서형 악성코드 개발자 또는 엔진 사용자들에게 공유할 수 있는 웹 사이트를 운영하여 문서형 악성코드 연구 및 개발에 기여하고자 한다.

웹 상단에는 페이지 간의 이동을 위해 네비게이션(Navigation) 바를 배치하였다. 홈 화면에는 드림의 간략한 소개와 이메일, 깃허브 링크를 명시하였다. 다운로드 페이지로 이동하였을 때 네비게이션 바의 DREAM 을 클릭하게 되면 홈 화면으로 돌아오게 된다. 본 프로젝트에서 공유하는 문서형 악성코드를 악의적으로 유포할 수 있기 때문에 계정이 없는 사용자들은 다운로드할 수 없게 한다. 데이터 목록은 전체 파일을 다 보여줄 경우 한 페이지가 길어지기 때문에 Paginator 를 사용하여 한 페이지당 백 개의 파일만을 보여준다. 그리고 검색 기능을 넣어 사용자가 원하는 파일의 MD5 값을 입력하면 해당 파일만 보여주게 된다.



<그림 25> 데이터 공유 사이트 초기 화면

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

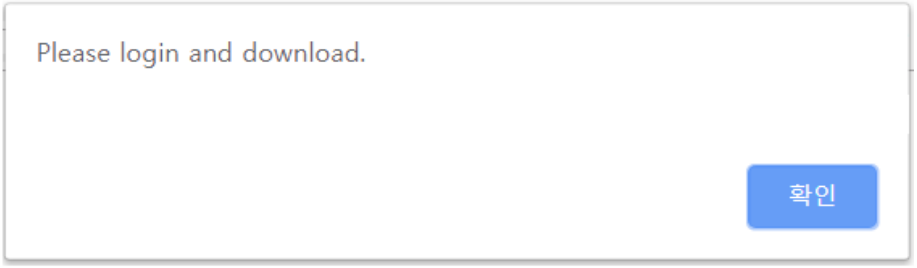
Download

LOG IN

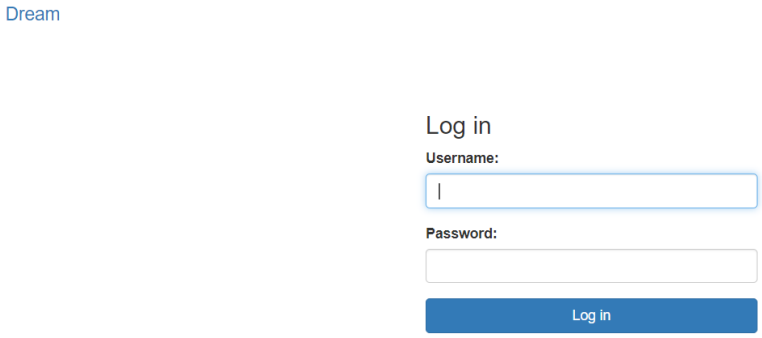
Word:

File	Name	MD5	size	type
Download	000e207f32bf2e43c90d702f2b05be7e.pdf	000e207f32bf2e43c90d702f2b05be7e	178 KB	PDF document, version 1.5
Download	0014546bcf4ad55d4d8971f80a063bc8.pdf	0014546bcf4ad55d4d8971f80a063bc8	3037 KB	PDF document, version 1.5
Download	0016640ed91200002873d3deef721af0.pdf	0016640ed91200002873d3deef721af0	28 KB	PDF document, version 1.7
Download	001a25e187f1f7b5c06890720f86f934.pdf	001a25e187f1f7b5c06890720f86f934	665 KB	PDF document, version 1.4
Download	00253120a070485b3c87b572d3448cf2.pdf	00253120a070485b3c87b572d3448cf2	52 KB	PDF document, version 1.4
Download	0025400b48ddfdcfe51045c4ce494a14.pdf	0025400b48ddfdcfe51045c4ce494a14	7045 KB	PDF document, version 1.6
Download	003246cce8249f376e660b50c5c09648.pdf	003246cce8249f376e660b50c5c09648	37 KB	PDF document, version 1.7
Download	003765bb5d19193804a3533a6136a7ce.pdf	003765bb5d19193804a3533a6136a7ce	207 KB	PDF document, version 1.7
Download	003be6251a3980ba65ae34180d0f4608.pdf	003be6251a3980ba65ae34180d0f4608	153 KB	PDF document, version 1.7
Download	003bf93b1839f774c68e0b3c4643bd39.pdf	003bf93b1839f774c68e0b3c4643bd39	11 KB	PDF document, version 1.7

<그림 26> 다운로드 페이지

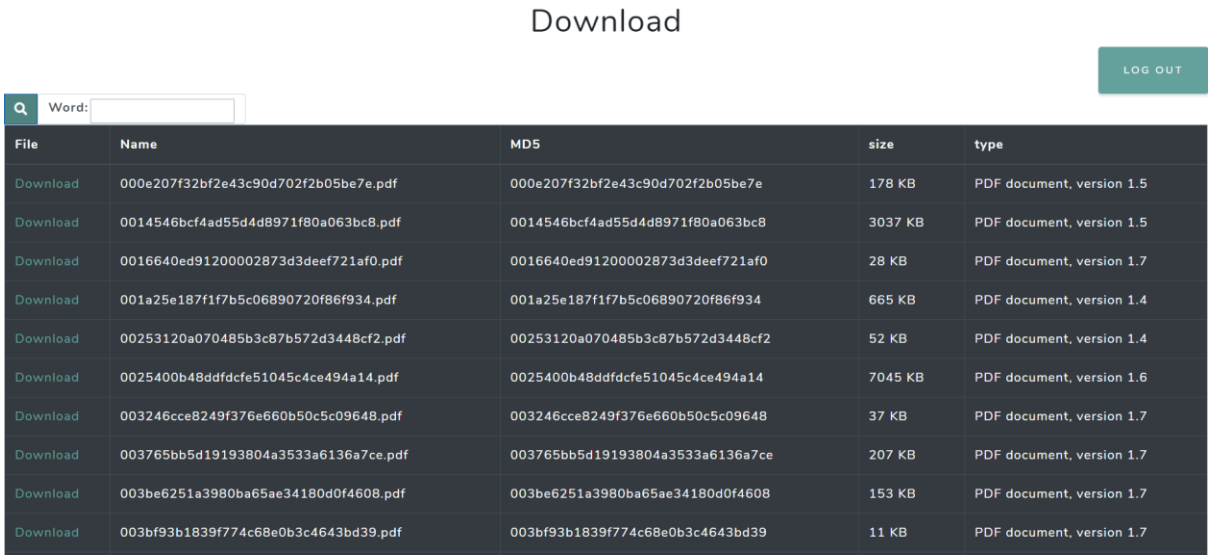


<그림 27> 로그인 없이 파일 다운로드를 시도 할 경우 화면



<그림 28> 로그인 화면

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.



<그림 29> 로그인 후 다운로드가 활성화 된 화면

<그림 30> 검색 결과 화면

2.2.2 시스템 기능 요구사항

Functional Requirements		
팀 명	Do It!	
분석 설계 대상 시스템 명	DREAM	
번호	내용	중요도
FR 1	유저가 "파일 첨부하기"를 누르면, 클라이언트는 파일 목록을 가져온다.	중
FR 2	유저가 "전송" 버튼을 누르면, 클라이언트는 서버로 업로드 할 파일을 전송한다.	중

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

FR 3	클라이언트는 유저에게 로딩 화면을 출력한다.	하
FR 4	서버가 파일을 전송받으면, 해당 파일을 HTTP 로 DREAM 엔진에 전송한다	상
FR 5	엔진은 서버로부터 파일을 전송받으면, 해당 파일의 형식을 검사한다.	상
FR 6	엔진은 서버로부터 파일을 전송받으면, 파일의 악성 여부를 탐지하기 위해 파일에서 특징을 추출하여 특징 벡터를 생성한다.	상
FR 7	엔진에서 특징 벡터를 생성하면, 분류기를 사용하여 특징 벡터로 파일이 악성일 확률을 계산한다.	중
FR 8	엔진에서 파일이 악성일 확률값을 계산하면, 계산된 값을 서버에 전송한다.	중
FR 9	서버는 엔진으로부터 결과값을 전송받으면, 결과 값을 통해 업로드 할 파일의 악성 여부를 판단한다.	중
FR 10	서버는 업로드한 파일의 악성 판단 결과를 클라이언트로 전송한다.	하
FR 11	클라이언트는 서버로부터 받은 악성 여부 결과가 악성이면, 업로드된 파일 목록에 악성으로 분류된 파일을 추가하지 않는다.	하
FR 12	클라이언트는 서버로부터 받은 악성 여부 결과가 정상이면, 업로드된 파일 목록에 파일을 추가한다.	상
FR 13	클라이언트는 서버에서 판단한 파일들의 악성 여부 결과를 전송받아 출력한다.	중

- 시연용 웹

1. (FR 1) 유저가 "파일 첨부하기"를 누르면, 클라이언트는 파일 목록을 가져온다. 완료
   ➔ 사용자가 선택한 경로에 위치한 폴더의 파일 목록을 보여준다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2. (FR 2) 유저가 "전송" 버튼을 누르면, 클라이언트는 서버로 업로드 할 파일을 전송한다. 완료
   ➔ AJAX를 사용하여 유저가 전송한 파일들을 서버로 전송한다.
3. (FR 3) 클라이언트는 유저에게 로딩 화면을 출력한다. 완료
   ➔ JQuery를 사용하여 파일이 실시간으로 전송되는 상태를 보여주는 상태 창을 화면에 출력한다.
4. (FR 4) 서버가 파일을 전송받으면, 해당 파일을 HTTP로 DREAM 엔진에 전송한다. 완료
   ➔ 파이썬의 requests 라이브러리를 사용하여 파일을 HTTP 형식으로 DREAM 엔진에 전송한다.
5. (FR 9) 서버는 엔진으로부터 결과값을 전송받으면, 결과 값을 통해 업로드 할 파일의 악성 여부를 판단한다. 완료
   ➔ DREAM 엔진으로부터 받은 확률값을 사용자가 설정한 임계 값에 따라 악성 정상을 판단한다.
6. (FR 10) 서버는 업로드한 파일의 악성 판단 결과를 클라이언트로 전송한다. 변경
   ➔ 업로드 할 파일의 악성 여부 판단 결과를 클라이언트에 전송할 필요가 없어 전송하지 않는 것으로 수정하였다.
7. (FR 11) 클라이언트는 서버로부터 받은 악성 여부 결과가 악성이면, 업로드된 파일 목록에 악성으로 분류된 파일을 추가하지 않는다. 변경
   ➔ 서버에서 업로드 할 파일을 악성으로 판단하면 해당 파일을 저장하지 않는다.
8. (FR 12) 클라이언트는 서버로부터 받은 악성 여부 결과가 정상이면, 업로드된 파일 목록에 파일을 추가한다. 변경
   ➔ 서버에서 업로드 할 파일을 정상으로 판단하면 해당 파일을 저장한다.
9. (FR 13) 클라이언트는 서버에서 판단한 파일들의 악성 여부 결과를 전송받아 출력한다. 변경
   ➔ 서버는 서버에서 정상적으로 업로드 한 파일들의 결과를 클라이언트에 전송하여 클라이언트는 업로드 된 파일들의 목록을 출력한다.

- 엔진
  1. (FR 5) 엔진은 서버로부터 파일을 전송받으면, 해당 파일의 형식을 검사한다. 완료
     ➔ 파이썬의 python-magic 라이브러리를 사용하여 파일의 유형을 확인한다.
  2. (FR 6) 엔진은 서버로부터 파일을 전송받으면, 파일의 악성 여부를 탐지하기 위해 파일에서 특징을 추출하여 특징 벡터를 생성한다. 완료
     ➔ 파일이 PDF 또는 MS Word 파일이면 파일 유형에 맞게 특징 벡터를 생성한다.
  3. (FR 7) 엔진에서 특징 벡터를 생성하면, 분류기를 사용하여 특징 벡터로 파일이 악성일 확률을 계산한다. 완료
     ➔ 생성한 특징 벡터를 여러 분류기를 사용하여 파일이 악성일 확률을 계산한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

4. (FR 8) 엔진에서 파일이 악성일 확률값을 계산하면, 계산된 값을 서버에 전송한다. 완료
   ➔ 계산한 결과를 JSON 형식으로 맞추어 서버에 전송한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.3 시스템 비기능(품질) 요구사항

Nonfunctional Requirements				
팀 명		Do It!		
분석 설계 대상 시스템 명		DREAM		
번호	내용	중요도	품질속성 상세	품질속성 내용
NFR 1	서버는 충분히 빠른 시간 내에 파일을 엔진에 전송해야 한다.	중	Time Behavior	엔진에 파일이 전송되는데 소요되는 시간이 최대 1 초를 넘기면 안 된다.
NFR 2	엔진은 충분히 빠른 시간 내에 특징을 추출해야 한다.	상	Time Behavior	서버로부터 전송받은 파일에서 특징을 추출하는 소요 시간이 최대 2 초를 넘기면 안 된다.
NFR 3	엔진은 사용하기 쉬운 형태로 구성되어야 한다.	상	Understandability	주요한 함수에 대해서는 Pseudo Code 를 첨부하여야 한다.
NFR 4	플랫폼에 의존적이지 않아야 한다.	상	Adaptability	파일을 첨부할 수 있는 서비스에는 해당 엔진이 사용 가능해야 한다.
NFR 5	엔진은 치명적인 오류가 발생했을 때 관리자에게 오류를 보고해야 한다.	상	Analyzability	관리자에게 오류를 보고함으로써 분석 및 패치를 할 수 있도록 한다.
NFR 6	엔진은 해당 모듈의 사용 설명문서가 내장되어야 한다.	중	Understandability	별도의 설명서를 추가하여 빠르게 이해할 수 있도록 해야 한다.
NFR 7	엔진은 중요한 데이터의 손상 없이 삭제되고 재설치 되어야 한다.	상	Replace-ability	엔진이 설치, 삭제 시 기존 데이터에

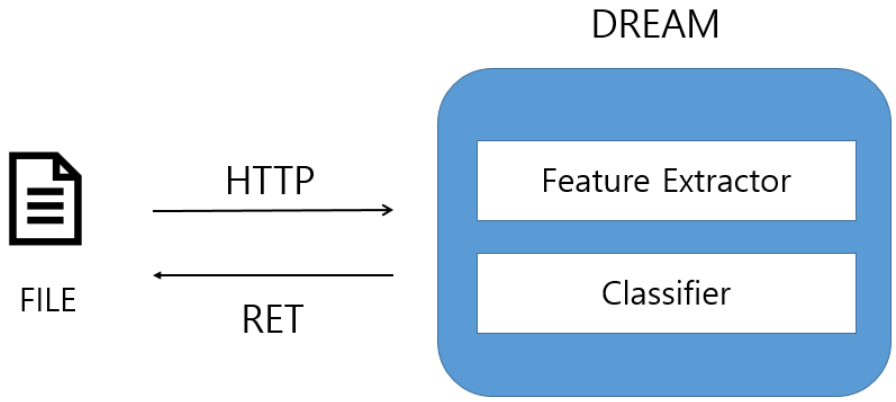
All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

				영향을 끼치지 않아야 한다.
NFR 8	엔진은 사용자의 별도 조작 없이 갑작스럽게 종료되면 안 된다.	상	Stability	엔진은 사용자가 별도의 조작을 통해 종료되어야 한다.
NFR 9	엔진은 파일이 악성일 확률값을 충분히 빠르게 결과를 도출해야 한다.	상	Time Behavior	생성한 벡터로부터 예측값 연산이 최대 1 초를 넘으면 안 된다.

1. (NFR 1) 서버는 충분히 빠른 시간 내에 파일을 엔진에 전송해야 한다. 달성
   ➔ 파일을 서버에서 엔진으로 전송하는데 파일당 평균 0.7초가 소요된다. 하지만 파일 용량이 10MB 이상일 경우 전송 소요 시간이 1초 이상으로 될 수 있다.
2. (NFR 2) 엔진은 충분히 빠른 시간 내에 특징을 추출해야 한다. 달성
   ➔ 파일에서부터 특징을 추출하여 특징 벡터를 생성하는데 파일당 평균 1초가 소요된다. 하지만 파일 용량이 1MB 이상일 경우 소요 시간이 다소 증가할 수 있다.
3. (NFR 3) 엔진은 사용하기 쉬운 형태로 구성되어야 한다. 달성
   ➔ 사용자들이 엔진을 사용하기 쉽게 API를 제공한다.
4. (NFR 4) 플랫폼에 의존적이지 않아야 한다. 달성
   ➔ 엔진 API 를 사용하거나 HTTP 사용해서 엔진과 연동할 수 있어 플랫폼에 의존 없이 엔진과 연동할 수 있다.
5. (NFR 5) 엔진은 치명적인 오류가 발생했을 때 관리자에게 오류를 보고해야 한다. 달성
   ➔ 에러 발생 시 에러 메시지를 출력한다.
6. (NFR 6) 엔진은 해당 모듈의 사용 설명문서가 내장되어야 한다. 달성
   ➔ 깃허브에 사용 문서를 정리하여 기록하였다.
7. (NFR 7) 엔진은 중요한 데이터의 손상 없이 삭제되고 재설치 되어야 한다. 달성
   ➔ 엔진을 PyPI 에서 배포하여 쉽게 엔진을 설치 및 삭제할 수 있다.
8. (NFR 8) 엔진은 사용자의 별도 조작 없이 갑작스럽게 종료되면 안 된다. 달성
   ➔ 엔진 종료 API를 사용하여 엔진을 종료할 수 있다.
9. (NFR 9) 엔진은 파일이 악성일 확률값을 충분히 빠르게 결과를 도출해야 한다. 달성
   ➔ 생성된 특징 벡터로 파일의 악성 여부를 판단할 때 소요되는 시간이 평균 0.1초이다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.4 시스템 구조 및 설계도



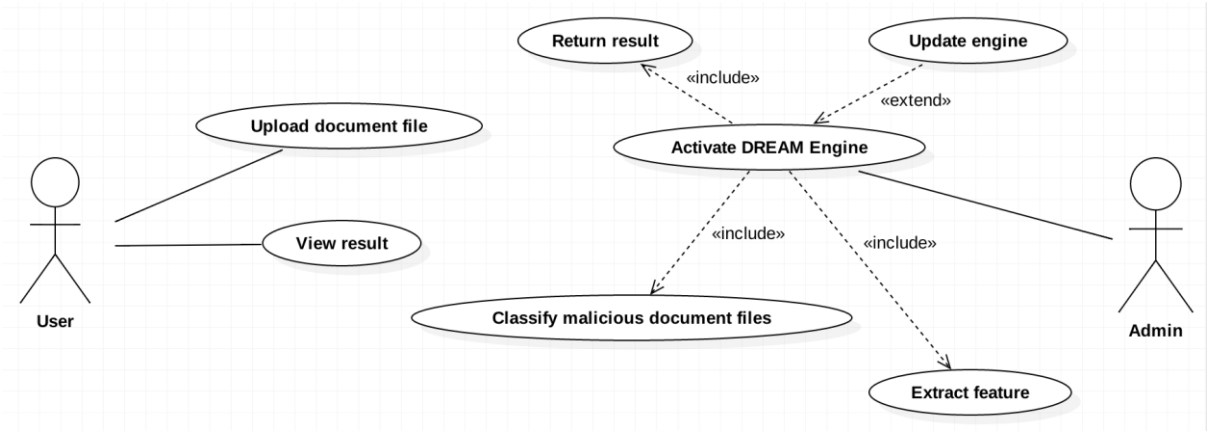
<그림 17> 시스템 구조도

어떠한 플랫폼이나 환경에서 문서 파일을 엔진에 업로드 하면 엔진은 문서 파일의 특징을 추출하여 특징 벡터를 생성한다. 생성한 특징 벡터로부터 학습된 기계 학습 분류기가 문서 파일이 악성일 확률을 도출해내고, 도출한 확률값을 다시 문서 파일을 업로드 했던 환경에 반환한다.

- 변경 사항
  AWS 로 갱신된 분류기를 다운받을 수 있는 클라우드 서버를 운영하여 엔진 사용자들이 엔진의 갱신 기능을 통해 최신의 분류기를 다운받아올 수 있도록 하려고 했으나, PyPI 에서 엔진 패키지와 함께 분류기를 배포하는 형식으로 변경했다. 그 이유는 AWS 는 대용량의 파일을 저장하고 배포하고자 할 때 적합하다. 분류기는 용량이 1MB 이하이기 때문에 PyPI 에서 패키지와 함께 분류기를 배포하는 것이 가능하다. 오히려 AWS 를 사용한다면 AWS 를 유지하는데 비용적인 부분이 발생한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.4.1. 유즈 케이스(Use Case)



<그림 16> 유즈 케이스(Use Case)

Name	Upload document file
ID	UC1
Description	User가 문서파일을 업로드한다.
Actors	User
Organizational Benefits	User가 업로드한 문서파일을 웹서버에서 엔진으로 전송하기 때문에 다양한 서버에서 엔진을 사용할 수 있다.
Frequency of Use	문서 파일을 검사하려는 사용자에게 복수적으로 일어난다.
Triggers	[파일추가]를 클릭하여 파일을 업로드한다.
Preconditions	업로드하려는 파일은 문서 파일이어야 한다.
Postconditions	로딩중인 화면을 출력해야한다.
Main Course	1. User가 [파일추가] 버튼을 클릭한다. 2. 원하는 파일을 선택한다. 3. [추가하기]버튼을 클릭한다.
Alternative Courses	없음
Exceptions	없음

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

Name	Update engine
ID	UC2
Description	Administrator가 DREAM Engine을 시작 할 때 자동으로 업데이트한다.
Actors	Administrator
Organizational Benefits	자동으로 Engine을 업데이트 함으로써 최신데이터로 학습된 Engine을 통해 새로운 악성문서를 탐지할 수 있다.
Frequency of Use	Engine을 구동시킬 때마다 단일적으로 일어난다.
Triggers	Administrator가 DREAM Engine을 시작한다.
Preconditions	Engine을 사용하는 환경은 인터넷에 연결되어 있어야 한다.
Postconditions	Engine은 최신 상태의 Engine 이어야 한다.
Main Course	1. Administrator는 DREAM Engine을 시작한다. 2. Engine은 클라우드 서버에 접근한다. 3. 클라우드 서버에서 분류기를 최신 분류기로 업데이트한다.
Alternative Courses	AC1: Administrator가 업데이트를 원하지 않을 경우 1. Engine을 실행 시, -nu 옵션을 주어 실행한다. 2. Engine은 분류기를 업데이트 하지않고 실행한다. AC2: 이미 최신 상태일 경우 1. 업데이트를 진행하지 않고 실행한다.
Exceptions	EX1: 서버 혹은 서비스가 인터넷에 연결되어 있지 않은 경우 1. 경고 메시지를 출력한다. 2. Engine은 업데이트 없이 실행할 것인지 묻는 메시지를 출력한다. 3. 사용자의 대답에 따라 Engine을 종료하거나, 업데이트 없이 실행한다.

Name	Extract feature
ID	UC3
Description	Engine은 웹서버로부터 전송받은 파일에서 feature를 추출하여 vector를 형성한다.
Actors	Engine
Organizational Benefits	문서의 구조적 정보를 특징으로 사용하기 때문에 적은 시간내에 feature를 추출함으로써 사용자에게 빠른 정보를 제공 할 수 있다.
Frequency of Use	Engine이 문서 파일을 전송 받았을 때, Engine에서 복수적으로 일어난다.
Triggers	웹서버로부터 문서파일을 정상적으로 전송받는다.
Preconditions	Engine은 웹서버로부터 문서파일을 정상적으로 받았어야 한다.
Postconditions	Feature Vector가 생성되어야 한다.
Main Course	1. 웹서버로부터 문서파일을 전송 받는다. 2. 전송받은 파일의 타입을 검사한다. 3. 문서 파일에 맞는 feature를 추출하여 Vector를 생성한다.
Alternative Courses	없음
Exceptions	EX1: PDF, DOCX 외의 다른 문서파일을 전송 받았을 경우 1. 파일 타입 에러 메시지를 서버에게 전송한다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

Name	Return result
ID	UC5
Description	전송 받은 파일의 악성일 확률의 값을 웹서버로 전송한다.
Actors	Engine
Organizational Benefits	확률값을 제공함으로써, 다양한 웹서버에서 별다른 수정없이 Engine을 사용할 수 있다.
Frequency of Use	전송받은 문서파일에 대한 악성 확률값을 구한 Engine에서 복수적으로 일어난다.
Triggers	파일에 대한 악성 확률값이 생성된다.
Preconditions	파일의 악성 확률값이 연산되었어야 한다.
Postconditions	웹서버는 파일의 악성 확률값을 정상적으로 받아야 한다.
Main Course	1. 전송받은 파일의 악성 확률값을 웹서버로 전송한다.
Alternative Courses	없음
Exceptions	없음

Name	classify malicious document file
ID	UC4
Description	Engine은 생성한 Feature Vector를 사용하여, 전송 받은 문서파일이 악성일 확률값을 구한다.
Actors	Engine
Organizational Benefits	Administrator는 악성일 확률에 대한 Threshold를 설정하여 사용함으로써 Engine을 사용자가 원하는 방향으로 사용할 수 있다.
Frequency of Use	문서 파일에서 Feature Vector가 생성된 Engine에서 복수적으로 일어난다.
Triggers	Feature Vector가 정상적으로 생성된다.
Preconditions	Feature Vector가 생성되어 있어야 한다.
Postconditions	악성일 확률값이 연산되어야 한다.
Main Course	1. Feature Vector를 입력받는다. 2. 파일이 악성일 확률값을 구한다.
Alternative Courses	없음
Exceptions	없음

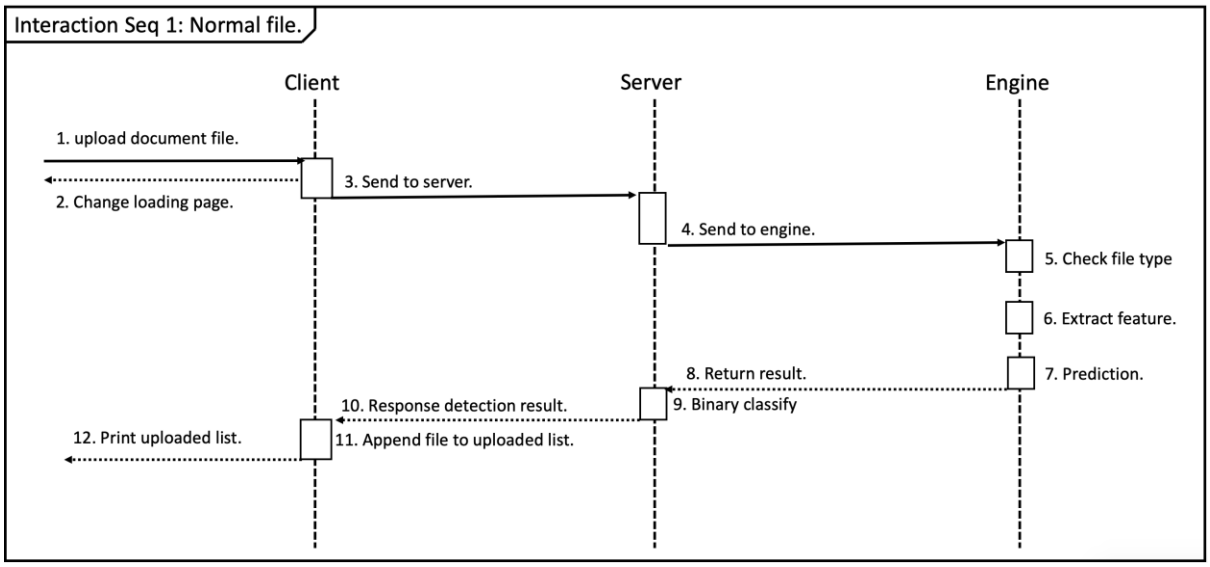
All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

Name	View result
ID	UC6
Description	업로드한 결과를 확인한다.
Actors	User
Organizational Benefits	없음
Frequency of Use	파일을 업로드 하였을 때 Client에서 복수적으로 발생한다.
Triggers	웹서버로부터 업로드한 파일의 악성 유무 결과를 전달받는다.
Preconditions	웹서버로부터 업로드한 파일의 악성 유무 결과를 전송받았어야 한다.
Postconditions	User는 업로드한 파일의 결과를 확인한다.
Main Course	1. 업로드의 결과를 확인한다.
Alternative Courses	없음
Exceptions	EX1: 업로드한 PDF 혹은 Microsoft Words 파일이 아닐 경우 1. 웹서버로부터 받은 파일 타입 에러 메시지를 출력한다.

2.2.4.2. 시퀀스 다이어그램(Sequence Diagram)

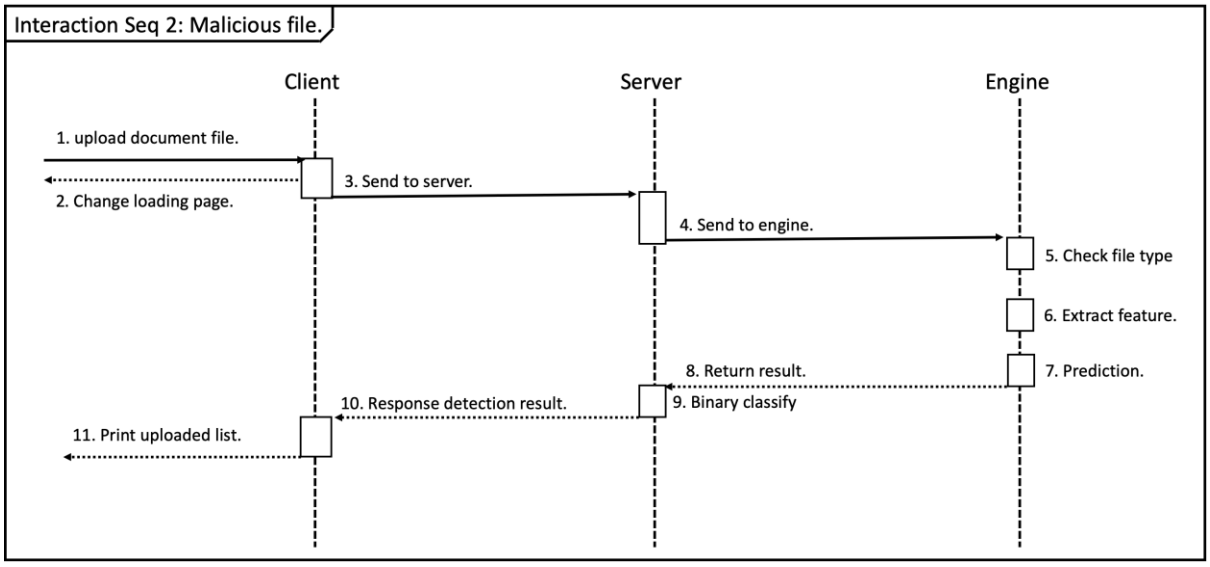
엔진 사용자가 운영하는 서버와 엔진을 연동한 후 발생할 수 있는 상황을 시퀀스 다이어그램으로 나타내었다.

- 경우 1: 엔진에 업로드 된 파일이 정상 파일 일 때



All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

- 경우 2: 엔진에 업로드 된 파일이 악성 파일일 때



경우 1, 2 공통

1. 사용자가 웹 클라이언트에 문서 파일을 업로드한다.
2. 웹 클라이언트는 사용자에게 파일을 업로드하는 로딩 화면을 출력한다.
3. 웹 클라이언트는 업로드 할 문서 파일을 웹 서버로 전송한다.
4. 웹 서버는 업로드 할 파일이 악성인지 검사하고자 엔진으로 파일을 전송한다.
5. 엔진은 전송받은 파일의 형식을 검사한다.
6. 엔진은 파일 형식에 맞게 특징을 추출하여 특징 벡터를 생성한다.
7. 엔진은 생성한 특징 벡터로부터 파일의 악성 확률을 계산한다.
8. 엔진은 계산한 결과를 웹 서버로 전송한다.
9. 웹 서버는 받은 결과로 파일의 악성 여부를 판단한다.
10. 판단 결과를 웹 클라이언트에 전송한다.

경우 1

11. 업로드 목록에 업로드 할 파일을 추가한다.
12. 업로드된 목록을 출력한다.

경우 2

11. 업로드된 목록을 출력한다.

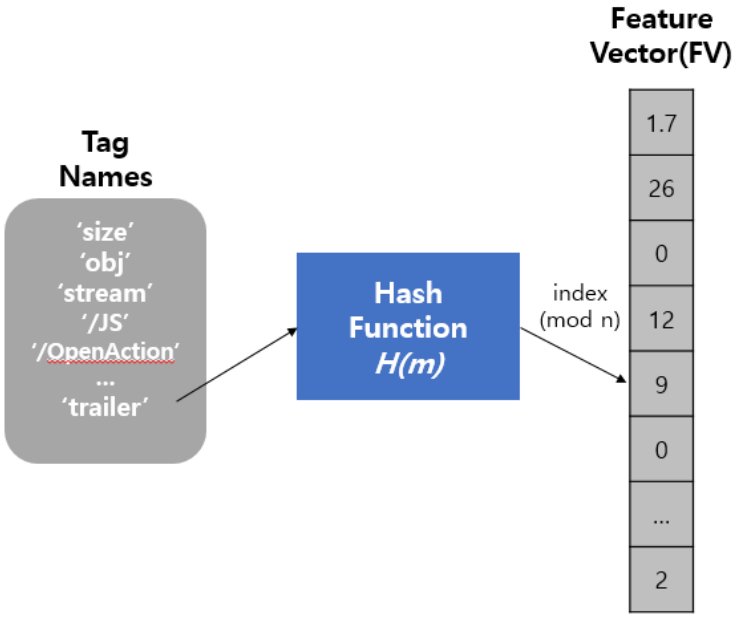
All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2.2.5 활용/개발된 기술

- 해싱 트릭

해싱 트릭이란 고차원의 특징을 저차원으로 투영시킬 수 있는 방법으로, <그림 31>과 같이 해시 함수를 사용하여서 한 특징의 해시값을 구하고 그 값을 특징 벡터 크기로 나눈 값을 인덱스(Index)로 하여 그 인덱스에 값을 누계하는 방법이다.

$$\text{FEATURE_VECTOR}[h(tag\ \ name)\ \ mod\ \ SIZE_OF_FV]\ \ +=\ \ \text{Num_of_tag}$$



<그림 31> 해싱 트릭

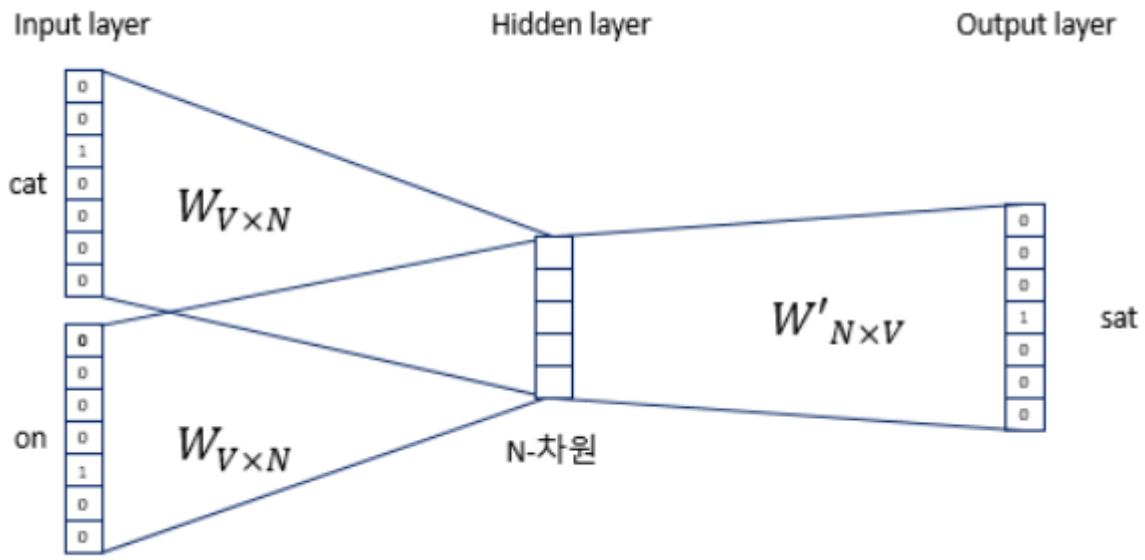
출처 = Kilian Weinberger KILIAN, Anirban Dasgupta ANIRBAN, John Langford et.al. *Feature Hashing for Large Scale Multitask Learning.* Proc. ICML 2009.

- 워드 임베딩(Word Embedding)

워드 임베딩(Word Embedding)은 단어를 벡터로 표현하는 대표적인 방법으로 주로 희소 표현에서 밀집 표현으로 변환하는 것을 의미한다. 일반적인 벡터는 원-핫 인코딩을 통해서 나온 원-핫 벡터들은 표현하고자 하는 단어의 인덱스의 값만 1이고, 나머지 인덱스에는 전부 0으로 표현되는 벡터 표현 방법이 가장 일반적인 벡터라고 할 수 있다. 이러한 벡터를 희소 벡터라고 하며 희소 벡터의 문제점은 단어의 개수가 늘어나면 벡터의 차원이 한없이 커진다는 점이다. 희소 표현과 반대되는 표현이 있으니, 이를 밀집 표현(Dense Representation)이라고 한다. 밀집 표현은 벡터의 차원을 단어 집합의 크기로 상정하지 않는다. 사용자가 설정한 값으로 모든 단어의 벡터 표현의 차원을 맞춘다. 또한, 이 과정에서 더 이상 0과 1만 가진 값이

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

아니라 실수값을 가지게 된다. 밀집 표현을 사용하고, 사용자가 밀집 표현의 차원을 X로 설정한다면, 모든 단어의 벡터 표현의 차원은 X로 바뀌면서 모든 값이 실수가 된다. 이 벡터의 차원은 X일 때 이 벡터의 차원이 조밀해졌다고 하여 밀집 벡터(Dense Vector)라고 한다. 단어를 밀집 벡터(Dense Vector)의 형태로 표현하는 방법을 워드 임베딩(Word Embedding)이라고 한다. 그리고 이 밀집 벡터를 워드 임베딩 과정을 통해 나온 결과라고 하여 임베딩 벡터(Embedding Vector)라고도 한다. <그림 32>는 임베딩 벡터의 예시이다.



<그림 32> 임베딩 벡터 예시(CBOW)

출처 = https://wikidocs.net/22660

- PDFiD

PDFiD는 PDF의 키워드나 오브젝트, 태그 정보 등을 스캔할 수 있는 오픈소스 도구이다. PDF를 분석할 때 먼저 PDFiD 를 사용하여 의심스러운 PDF를 분류한 다음 다른 PDF 파서 도구를 사용하여 PDF를 분석할 수 있다.

2.2.6 현실적 제한 요소 및 그 해결 방안

2.2.6.1. 현실적 제한 요소

1. 학습에 필요한 데이터
   - 학습을 위해 양질의 데이터가 필요했으나 현실적 요소로 데이터를 쉽게 구하지 못하였다.
   - 악성 HWP 파일 탐지와 관련된 연구도 진행해보려 했으나 HWP 파일은 국내에서만 사용하기 때문에 악성 HWP 파일 수집 자체가 어려웠다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

2. 관련 연구의 부족
   - 문서형 악성코드 탐지와 관련된 최근 연구가 적어 참고할 수 있는 자료가 적었다. 특히 악성 HWP 탐지와 관련된 논문은 전무하다.

2.2.6.2. 해결 방안

데이터 부족 문제를 해결하고자 바이러스 토탈에 Academic API를 요청하여 데이터를 공유받고자 했다. 바이러스 토탈의 승인으로 악성 PDF (1만 개)와 악성 MS Word (Doc 5천 개, Docx 5백 개)를 공유받아 실험에 활용할 수 있었다.

2.2.7 결과물 목록

- 문서형 악성코드 탐지 엔진 (설치 매뉴얼 O, 사용자 매뉴얼 O)
- 문서형 악성코드 공유 사이트
- 엔진 사용자 오픈소스 커뮤니티

2.3 기대효과 및 활용방안

2.3.1 기대효과

본 프로젝트는 문서형 파일을 서비스에 업로드 하기 전 서비스에서 업로드 할 파일이 악성인지 정상인지 판별하여 업로드를 제한함으로써 문서형 악성코드의 유포를 방지한다. 현재 문서형 악성코드는 사람들의 관심을 끌 만한 제목으로 유포되고 있으며 사용자들은 해당 파일들에 경계심을 갖지 않고 다운로드하여 실행할 수 있다. 이로 인한 피해로 발생할 수 있는 사회적 문제를 예방할 수 있을 것이라 기대한다.

2.3.2 활용방안

- 웹/메일 서버

웹 또는 메일 서버 운영자는 서버에 본 프로젝트의 엔진을 연동하여 업로드되는 파일을 검사함으로써 문서형 악성코드가 무단으로 업로드되어 유포되는 것을 방지할 수 있다.

- PC

본 프로젝트의 엔진을 PC에서 실행하여 사용자의 컴퓨터에 문서형 파일을 검사할 수 있다.

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

3 자기평가

최종 결과물 - 기계학습 기반의 문서형 악성코드 탐지 엔진

본 프로젝트는 문서형 악성코드로 발생하는 피해가 증가하지만, 대다수의 기존 안티바이러스들이 탐지하지 못하는 문제를 보완하고자 한다. 최근 악성코드는 규칙(Rule) 기반의 안티바이러스를 우회하기 위해 규칙에 탐지되지 않는 변종 혹은 신종 악성코드를 양산하고 있다. 따라서 기존에 많이 사용된 규칙 기반이 아닌 기계학습 기반의 탐지 엔진을 개발하고자 하였다. 이를 통해 "알려진 문서형 악성코드 및 신종/변종 문서형 악성코드에 대해 올바른 분석을 하는가"를 목표로 한다.

수많은 문서, 예를 들면 이력서와 같은 문서가 들어오는 기업체 같은 경우 문서형 악성코드에 대한 대비가 필요하다. 드림 엔진은 어떤 웹 서버, 메일 서버, 개인 PC 등 다양한 곳에 사용할 수 있는 높은 확장성을 가진 엔진이다. 이에 따라 드림 엔진을 탑재한다면 문서형 악성코드를 탐지할 수 있다. 또한 드림 엔진은 악성을 여부를 확률값으로 주기 때문에 사용자가 원하는 분류기를 선택해서 악성 판단 임계 값을 설정할 수 있다.

최종적으로 드림 엔진을 오픈소스 소프트웨어로 개발하고 데이터 공유 웹 사이트를 운영하여 사용자들과 함께 발전 및 확장 시킬 수 있다. 이를 통해 문서형 악성 코드로 발생하는 사회적 문제들을 해결하는데 기여할 수 있다.

- 보안업체와 연계하여 본 프로젝트가 추가로 진행될 예정

4 참고 문헌

번호	종류	제목	출처	발행년도	저자	기타
1	웹 페이지	바이러스토탈 주간 통계	https://www.virustotal.com/en/statistics/	2019		
2	기사	이력서 위장 '시그마' 랜섬웨어... 전 세계 유포	https://www.sedaily.com/NewsView/1RX4PK7LDU	2018		
3	기사	MS 워드 매크로 악용 '갠드크랩' 랜섬웨어 기승	http://it.chosun.com/site/data/html_dir/2018/11/16/2018111601952.html	2018		
4	기사	정부 사칭 이메일 공격 계속 발견	http://www.boan.com/news/article.htm	2018		

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

			l?id=20181130150004			
5	논문	문서 구조 및 스트림 오브젝트 분석을 통한 문서형 악성코드 탐지	http://www.dbpia.co.kr/Journal/ArticleDetail/NODE07565787	2018	강아름, 정영섭, 외 4인	
6	논문	Malicious PDF Detection using Metadata and Structural Features	https://dl.acm.org/citation.cfm?id=2420987	2012	Charles Smutz, Angelos Stavrou	
7	기사	국내 유명 변호사 사칭한 악성코드, 상당수 안티바이러스 탐지 못해	https://www.boannews.com/media/view.asp?idx=74302	2018		
8	기사	요즘 해커들 사이에서 가장 인기 높은 건, MS 오피스	https://www.boannews.com/media/view.asp?idx=76967&page=1&mkind=1&kind=1	2019		
9	논문	ALDOCX: Detection of Unknown Malicious Microsoft Office Documents Using Designated Active Learning Methods Based on New Structural Feature Extraction Methodology	https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7762928	2017	Nir Nissim, Aviad Cohen, and Yuval Elovici	
10	논문	A Research of Anomaly Detection Method in MS Office Document	http://kiss.kstudy.com/thesis/thesis-view.asp?key=3498648	2017	조성혜, 이상진	
11	논문	Malware Detection by Eating a Whole EXE	https://arxiv.org/abs/1710.09435	2018	Edward Raff, Jon Barker, Jared Sylvester, Robert Brandon et.al.	
12	논문	Feature Hashing for Large Scale Multitask Learning	https://arxiv.org/abs/0902.2206	2009	Kilian Weinberger KILIAN, Anirban Dasgupta ANIRBAN, John Langford et.al.	
13	웹 페이지	워드 임베딩	https://wikidocs.net/22660	2019		

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

5 부록

5.1 설치 매뉴얼

본 프로젝트에서 개발한 엔진은 파이썬 패키지 배포 사이트인 PyPI 를 통해서 배포하고 있다.

- DREAMAV 설치 : https://pypi.org/project/dreamav/

5.2 사용자 매뉴얼

엔진 사용 방법과 엔진을 다른 서비스와 연동시키는 방법을 깃허브에 공개한다.

- DREAMAV 사용 매뉴얼 : https://github.com/kookmin-sw/2019-cap1-2019_8/tree/master/dreamav

1. 엔진 사용 방법
   - dreamav start : 엔진 가동
   - dreamav stop : 엔진 끄기
   - dreamav submit /path/to/file : 파일의 악성 여부 검사
   - dreamav scan path/to/file : 바이러스토탈에 파일을 신규 스캔 요청

API

- [POST] /dreamav_upload : 파일을 엔진에 전송한다.

2. 서버와 엔진 연동하기.

   엔진은 JSON 형식으로 여러 모델에 대한 파일의 악성일 확률의 결과를 제공한다.

PDF

```
{
    "result":{
        "LightGBM": PROB,
        "XGBoost": PROB,
        "RF": PROB,
    }
}
```


All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

MS Word(DOC)

```
{
    "result":{
        "DL": PROB
    }
}
```

사용자는 파일의 악성 여부를 판단할 모델을 선택하고 악성일 확률의 임계 값을 설정하여 문서형 악성코드를 판단한다. 아래는 Django 로 개발한 웹 서버에서 엔진을 연동한 예시이다.

```
req = requests.post("http://localhost:8080/dream_upload",
files={"file": self.request.FILES["file"]})

if req.status_code == 200:
    result = req.json()

# set threshold(recommend=0.5)
th = 0.5
if float(result["result"]["LightGBM"]) < th:
    ...
```

3. 엔진을 사용하는데 필요한 라이브러리
   - numpy==1.16.2
   - lightgbm==2.2.4
   - uWSGI==2.0.18
   - Flask==1.0.2
   - python-magic==0.4.15
   - Click==7.0
   - requests==2.21.0
   - tensorflow==1.13.1
   - keras==2.2.4
   - scikit-learn==0.20.0

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.

5.3 테스트 케이스

대분류	소분류	기능	테스트 방법	기대 결과	테스트 결과
엔진	연동	엔진과 웹 서비스를 연동한다.	1. 웹 서비스에 업로드되는 파일을 엔진으로 보낸다. 2. 웹 서비스에서 보낸 파일을 엔진이 받으면 파일을 받았다는 메시지를 웹 서비스에 보낸다.	엔진과 웹 서비스를 연동하여 웹 서비스에 업로드되는 파일을 엔진이 검사하여 파일의 악성 여부를 검사하게 할 수 있다.	성공
엔진	검사	웹 서비스에 업로드된 파일을 엔진으로 검사한다.	1. 웹 서비스에 문서 파일을 업로드한다. 2. 웹 서비스는 업로드 할 파일의 악성 여부를 판단하기 위해 파일을 엔진에 보낸다. 3. 엔진은 파일을 받아 파일이 악성일 확률을 계산하여 웹 서비스에 반환한다. 4. 웹 서비스는 결과값을 받아 악성 여부를 판단하여 업로드 할 파일의 제한을 결정한다.	웹 서비스에 업로드 할 문서 파일이 악성인지 검사하여 제한함으로써 문서형 악성코드가 웹 서비스에 업로드되는 것을 방지할 수 있다.	성공
웹	공유	데이터 공유 웹 사이트에서 데이터를 다운받는다.	1. 데이터 공유 웹사이트에서 데이터를 다운받는다. 2. 계정이 없을 경우 데이터를 다운받을 수 없다. 3. 계정이 있을 경우 데이터를 다운받을 수 있다.	문서형 악성코드를 문서형 악성코드 개발자 또는 엔진 사용자들에게 공유하여 문서형 악성코드 연구 및 개발에 기여하고자 한다.	성공

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.