JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지 Vol. 23, No. 10: 1311~1319, Oct. 2019

웹서버 로그 데이터의 이상상태 탐지 기법

이화성[1*] · 김기수[2]

Novelty Detection on Web-server Log Dataset

Hwaseong Lee[1*] · Ki Su Kim[2]

[1*]Senior Researcher, Agency of Defense and Development, Daejeon 34186, Korea
[2]Researcher, Agency of Defense and Development, Daejeon 34186, Korea

요 약

현재 웹 환경은 정보 공유와 비즈니스 수행을 위해 보편적으로 사용되고 있는 영역으로 개인 정보 유출이나 시스템 장애 등을 목표로 하는 외부 해킹의 공격 타켓이 되고 있다. 기존의 사이버 공격 탐지 기술은 일반적으로 시그니처 기반 분석으로 공격 패턴의 변경이 발생할 경우 탐지가 어렵다는 한계가 있다. 특히 웹 취약점 기반 공격 중 삽입 공격은 가장 빈번히 발생하는 공격이고 다양한 변형 공격이 언제든 가능하다. 본 논문에서는 웹서버 로그에서 정상상태를 벗어나는 비정상 상태를 탐지하는 이상상태 탐지 기법을 제안한다. 제안된 방법은 웹서버 로그 내 문자열 항목을 머신러닝 기반 임베딩 기법으로 벡터로 치환한 후 다수의 정상 데이터와 상이한 경향성을 보이는 비정상 데이터를 탐지하는 머신러닝 기반 이상상태 탐지 기법이다.

ABSTRACT

Currently, the web environment is a commonly used area for sharing information and conducting business. It is becoming an attack point for external hacking targeting on personal information leakage or system failure. Conventional signature-based detection is used in cyber threat but signature-based detection has a limitation that it is difficult to detect the pattern when it is changed like polymorphism. In particular, injection attack is known to the most critical security risks based on web vulnerabilities and various variants are possible at any time. In this paper, we propose a novelty detection technique to detect abnormal state that deviates from the normal state on web-server log dataset(WSLD). The proposed method is a machine learning-based technique to detect a minor anomalous data that tends to be different from a large number of normal data after replacing strings in web-server log dataset with vectors using machine learning-based embedding algorithm.

키워드 : 웹서버 로그 데이터, 임베딩, 이상상태 탐지, 비정상행위

Keywords : Web-server log dataset, Embedding, Novelty detection, Abnormal Behavior

Received 25 July 2019, **Revised** 1 August 2019, **Accepted** 25 August 2019

*** Corresponding Author** Hwaseong Lee(E-mail:hwaseong@korea.ac.kr, Tel:+82-42-822-4271)
Senior Researcher, Agency for Defense and Development, Daejeon 34186, Korea

Open Access http://doi.org/10.6109/jkiice.2019.23.10.1311 print ISSN: 2234-4772 online ISSN: 2288-4165

Ⅰ. 서 론

웹 어플리케이션은 인터넷의 사용이 일반화되고 클라우드 환경이 도래함에 따라 지속적인 사용 증가가 일어나고 있다. 특히 클라우드 시대를 맞이하여 개인정보와 관련된 민감한 정보와 각종 문서 파일도 클라우드 혹은 웹 환경에서 저장 및 공유되고 있고 웹 환경을 통한 공격도 발생하기 때문에 웹 어플리케이션을 공격으로부터 방어하는 것은 중요한 기술이다[1]. 2017년 발표된 웹 어플리케이션 보안 위협 랭킹에 따르면 인증(Authentication)과 세션 관리 외에도 전통적인 공격방법인 SQL 삽입과 XSS(cross-site scripting) 공격이 각각 1위와 3위를 차지하였다[2]. 이 외에도 새로운 공격이 발생할 경우 이를 탐지할 수 있어야 한다. 따라서 웹서버 취약점을 이용한 SQL 삽입이나 XSS 공격과 그 외에 알려지지 않은 공격에 대해서 강건한 탐지 기법이 필요하다.

상기 설명한 공격을 탐지하기 위해서는 시그너처 기반 탐지와 이상상태 탐지 기법(Novelty Detection)이 있다. 시그너처 기반 탐지는 요청에 대한 공격 패턴을 목록화하여 패턴에서 벗어날 경우 탐지하는 기법으로 변형된 공격에 대해서 미탐이 발생한다. 이와 반대로 이상상태 탐지 기법은 최근에 제안된 이론으로 정상적인 요청으로 정상상태 모델을 생성한 후 정상적인 요청에서 벗어난 형태의 요청이 발생할 경우 이상상태로 탐지한다. 이는 변형된 공격이나 알려지지 않은 공격에 대해서도 탐지 가능성이 있고 사용자의 단순 오용에 대해서도 이상상태로 간주할 가능성이 존재하므로 시그너처 기반 탐지보다 유연한 탐지가 가능하다.

이상상태 탐지 기법은 머신 러닝(Machine Learning)을 이용한 분석이다. 머신 러닝에서 가장 기본적이고 중요한 부분은 라벨이 부여된 모의 데이터셋 생성이다. 그리고 도메인에 따라서 데이터셋에 문자열이 포함되었다면 머신 러닝을 위해서 벡터로 치환(벡터화)하는 임베딩 과정이 필수적이다. 본 논문에서는 먼저 웹서버 로그 데이터 기반의 이상상태 탐지를 위해서 웹서버 환경을 구축하여 정상행위와 비정상행위를 모의하여 라벨링 된 자체 데이터셋 WSLD (Web-server Log Dataset)을 생성한다. 이 후 문자열 항목의 임베딩(Embedding)을 수행하여 벡터로 치환하고 정상행위로 정상상태 모델을 생성한 후 정상행위에서 벗어난 행위가 발생할 경우 이상상태 탐지를 하는 모델을 제안한다.

본 논문에서는 웹서버 로그 데이터의 이상상태 탐지 기법을 제안하기 위하여, 2장에서는 관련 연구를 조사하고 3장에서는 데이터셋 생성 방법을 기술하며 4장에서는 웹서버 로그 기반의 이상상태 탐지 기법을 제안한다. 5장에서 실험을 통해 모델을 분석한다.

Ⅱ. 관련 연구

웹 어플리케이션 영역의 증가로 웹 어플리케이션은 공격자들에게 주요 공격 대상이 되었고, 이와 함께 웹 기반 공격도 증가하고 있다. 웹 공격을 탐지하기 위해 많이 사용되는 규칙 기반 기술은 사전에 규칙으로 정의한 공격은 높은 탐지율을 결과로 보이지만, 알려지지 않은 공격은 탐지할 수 없다는 단점이 있다. 알려지지 않은 공격에 대응이 가능한 비정상 기반 탐지 기술은 정상상태를 정의하는 것이 어려운 문제이나, 최근 머신 러닝 기술 영역의 연구가 활해지면서 이를 적용한 비정상 기반 기술들이 연구 및 개발되고 있다.

Liang et al.[3]은 웹 요청의 비정상을 탐지하기 위해 딥 러닝 모델 기반 비정상 탐지 방법을 제안했다. 웹 요청을 분석하여 얻은 URL의 절대경로와 쿼리 파라미터를 파싱하고, LSTM(Long Short-term Memory) / GRU (Gated Recurrent Unit)를 이용한 RNN(Recurrent Neural Network) 모델의 입력 데이터로 사용하여 절대 경로와 쿼리 파라미터의 정상 패턴을 학습한다. 학습된 RNN 모델의 결과는 Multi layer Perceptron 모델의 입력으로 활용하여 웹 요청의 비정상을 탐지한다. 학습된 모델은 좋은 성능을 보여주지만, GET 요청 방식만 고려한다는 점과 딥 러닝 모델을 학습하는 단계에서 정상 웹 요청과 비정상 요청 데이터를 학습하는 것은 실제 운용 상황에서 적절하지 않을 수 있다.

Mac et al.[4]는 웹 요청의 비정상을 탐지하기 위해 잘 알려진 딥 러닝 모델 중 하나인 Regularized Deep Autoencoder를 사용한 비정상 탐지 방법을 제안했다. 웹 요청의 URL을 요청 방식, 절대 경로, 쿼리 파라미터로 구분하기 위해 토큰화(Tokenizing)하고 문자를 아스키 코드와 일치하는 숫자로 치환하고 정규화한다. 전처리 단계에서 생성한 벡터로 오토인코더(Autoencoder)를 학습한다. 학습된 오터인코더 모델의 결과는 실제 데

이터와 모델이 예측한 데이터간의 오차 오류 값으로 이 값의 분포 내 임계값(Threshold)을 기준으로 정상과 비정상을 식별한다. 학습된 모델은 매우 좋은 성능을 보여준다. 하지만, 웹 URL 경로의 파라미터들 간의 순서는 배제하고 학습하기 때문에 비정상적인 위치 또는 방법으로 접근한 정상행위나, 고도화된 공격의 맥락은 놓칠 수 있다.

Ⅲ. 웹서버 로그 데이터셋(WSLD) 생성

3.1. Access log

로그는 시스템의 문제를 찾거나, 성능 현황을 분석하기 위해서 사용된다. 시스템에는 많은 로그들이 기록되지만 이 중 Apache HTTP server의 Access log는 서버가 처리하는 모든 요청을 기록한다. 웹서버는 클라이언트의 요청(Request)을 받고, 해당 웹 페이지를 응답(Response)한다. 이때 요청 라인 및 상태 라인은 HTTP 구조 내 최상위 라인에 입력되어 전달되고, 웹서버는 접속 이력에 대해서 Access log에 저장한다 (그림 1).

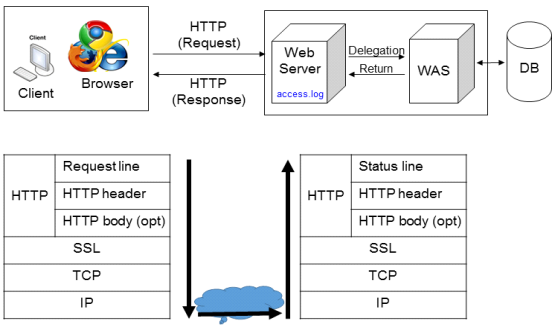

Fig. 1 Web Server and HTTP structure

Table. 1 Fields in Access Log

Type	Desc.
Field List	Remote_host Username Auth_username Timestamp Request-line Response-code Response-size Referer user-Agent
Format	%h %i %u %t "%r" %s %b " {%User-agent}" "%{Referer}"
Example	192.168.19.47 - - [03/Aug/2002:21:56:55 +0900] "GET /doc/images/sub.gif HTTP/1.1" 200 6083 Mozilla/5.0(Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) "http://unix.example.com:8080/login/loginForm.do"

Apache HTTP server의 Access log의 포맷은 혼합 로그 포맷(Combined Log Format)으로 일반적인 포맷에 Referer와 User-Agent 필드가 추가된 포맷이다. 표 1은 Access log에 기록된 정보 및 예시를 나타낸다.

3.2. 행위 모의 및 데이터양

정상상태 모델을 생성하고 모델의 성능을 검증하기 위해서는 정상행위와 비정상행위가 모두 필요하다. 정상/비정상행위는 실험 환경에서 발생하는 행위와 유사하게 모의될 필요가 있다. 모든 행위는 기본적으로 가변성(랜덤 기법) 기반 스케줄링과, 모의 시나리오 등에 의해 자동 생성된다. 데이터양은 모의 시간을 조절하여 계속적으로 증가시킬 수 있지만 실험을 위해서 약 5만개의 정상 및 비정상 접근이력으로 데이터셋을 구성했다.

3.2.1. 정상행위 모의

정상행위 모의란 사용자가 일반적으로 웹서버에 접속하는 방식대로 행위를 모의하는 것을 의미한다. 기본적으로 가장 많이 사용되는 메소드인 GET과 POST에 해당되는 행위를 모의하고, 다양한 모의 데이터를 자동 생성한다.

3.2.2. 비정상행위 모의

비정상행위 모의란 사용자가 악의적인 목적으로 수행하는 사이버 공격 뿐 아니라 단순 오용 등 정상행위에서 벗어난 행위 전반을 발생시키는 것을 의미한다.

- 정적 공격: 숨겨진 혹은 존재하지 않은 리소스에 대한 요청을 시도하는 행위로 임의의 폴더나 파일에 무작위로 접근을 시도한다.
- 동적 공격: 웹서버 취약점을 악용한 공격으로 유효한 요청의 인자를 수정한 공격을 뜻한다. Command injection, SQL injection, XSS 공격 등이 이에 속한다.
- 기타 단순 오용 등: 이 요청은 악의적인 의도를 가지고 있지는 않지만 웹 어플리케이션의 정상적인 행위를 따르지 않는 경우에 해당된다.

3.3. 실험환경 및 라벨링

3.3.1. 실험환경

실험 환경에서 행위를 모의하기 위하여 리눅스 웹서버 2대와 정상행위와 비정상행위를 생성할 호스트 7대로 구성된 실험환경을 구축한다. 호스트 1-2는 정상행위

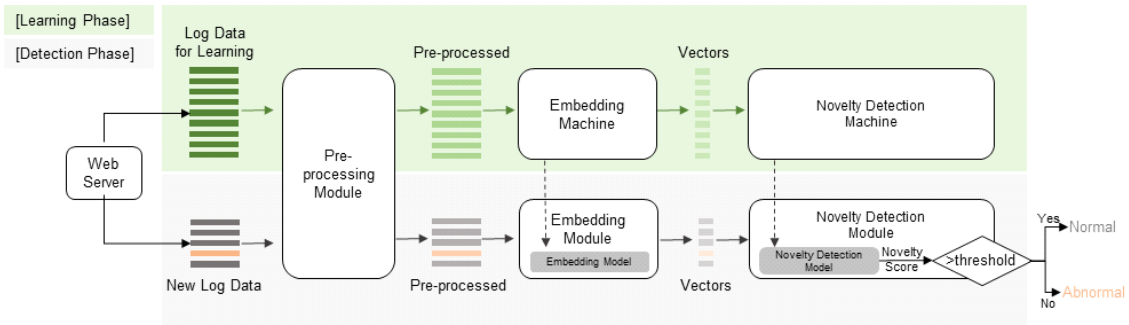

Fig. 2 The architecture of the web novelty detection

와 비정상행위를 모두 모의하고 호스트3-4는 정상행위만 호스트5-7은 비정상행위만 모의한다. 웹서버1은 정상행위만 수집하고 웹서버2는 정상/비정상행위를 모두 수집하도록 네트워크를 구성하였다.

3.3.2. 데이터셋 라벨링

웹서버 로그의 경우 접속이력이 하나의 라인으로 기록되기 때문에 라인별로 라벨(정상, 비정상)을 부여할 수 있고 웹서버2의 경우 웹서버에 접속한 호스트 IP를 기준으로 정상행위와 비정상행위가 구분 가능하다. 라인별 라벨링으로 정상상태 모델 생성 및 검증을 위한 웹서버 로그 기반 데이터셋 WSLD (Web-server Log Dataset)이 생성된다.

Ⅳ. 웹서버 로그 기반 이상상태 탐지 기법

본 논문에서 제안하는 이상상태 탐지 기법은 그림 2와 같이 학습단계(Learning Phase)와 탐지단계(Novelty Detection Phase)로 나뉜다. 학습단계에서는 정상적인 초기 운용상태일 때를 가정하고 정상행위만으로 지속적인 학습을 통해 정상상태 모델을 생성한다. 정상상태 모델은 기법에 따라 다양한 모델이 생성 가능하기 때문에 정상상태 모델 학습에 사용되지 않은 정상행위와 비정상행위로 모델들 간의 성능 비교로 최적의 모델이 식별 가능하다. 이 때, 모델의 파라미터 뿐 아니라 이상치(Novelty Score)의 임계값이 결정된다.

탐지단계에서는 웹서버가 실제 운용되는 상태로 정상상태 모델을 기반으로 신규 웹서버 로그의 이상치를 산출하고 임계값과 비교하여 이상상태를 탐지한다. 이를 통해 알려지지 않은 공격이나 단순 오용 등 웹서버 이상 여부를 인지할 수 있다. 본 장에서는 3장에서 생성된 데이터셋을 입력값으로 받아 전처리, 임베딩, 이상상태 탐지 방법을 단계적으로 설명한다.

4.1. 데이터 전처리

웹서버 로그를 구성하는 대부분의 항목은 문자열로 구성되어 있고 문자열은 임베딩 통한 벡터화 과정이 필수적이다. 임베딩이 필요한 문자열 항목은 스트링 정보(예: URL, referrer 등)와 범주형 변수(예: http_version, status)로 구성되어 있다(표 2 참고). 범주형 변수는 숫자 형태로 표현되지만 증감을 의미하는 산술적 숫자와는 구별된다. 가령, 응답 코드(Response Code)의 경우 호스트의 요청에 대하여 2**는 성공, 4**은 에러 등 요청의 처리 여부를 표현한다. 따라서 범주형 변수 역시 수치화된 데이터로 변환해줄 필요가 있다.

로그에 내재된 맥락 정보도 정상상태 학습에 필요하기 때문에 로그 정보를 적합한 형태로 전처리(Pre-processing)를 해야 한다. 예를 들어, URL의 절대 경로는 요청한 자원의 경로를 '/'로 구분하여 표현한다. 이 때, n 번째 경로는 n-1번째 경로의 영향을 받기 때문에 그 문맥이 유지되도록 전처리를 진행한다. 임베딩이 필요한 로그 항목은 표 2와 같다. URL의 경우는 경로 및 파라미터 단어 간 공백을 추가하여 문장화하였고 http_referrer와 user_agent는 공백을 제거하여 하나의 단어(Word)로 취급하였다. Timestamp는 unique한 값으로 학습에서 배제하기 위해 전처리 항목에 포함하지 않고 IP와 함께 임베딩 시 문서 ID (Documentation ID)로 사용한다.

Table. 2 A list of fields in web-server log

Field Name	Desc.
method	Request method to a web server (GET, POST etc)
URL	Path to access to HTTP server, including parameters(operator, operand and variables) on GET method
status	Status code that the server sends back to the client, including success, failure, redirection etc
http_version	http version information
referrer	Address of webpage which is linked to the resource being requested. Web-server can see where the request originated (Optional)
user_agent	Software agent that is acting on behalf of a client, such as web browser [ex:Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.1; WOW64; Trident/6.0)]

앞장에서 설명한 것과 같이 웹서버 로그는 혼합 로그 포맷으로 총 9개의 영역으로 구성되어 있다. 각각은 요청한 호스트 정보, 요청 날짜와 시간, 요청한 행위 등 호스트가 웹서버에 송신한 요청의 정보를 표현한다. 이 중 정상과 비정상행위 식별에 영향을 미칠 것으로 예상되는 주요 항목은 URL 항목이다. 왜냐하면, 웹서버 취약점을 이용한 사이버 공격 시 GET 방식의 공격은 URL에 공격 흔적이 남고 존재하지 않은 자원 요청이나 파라미터의 도메인 범위 밖의 변수를 포함한 요청 등 사용자의 실수로 인한 접근(Access)도 URL에 정보가 남기 때문이다. 그 외에 문자열 정보 역시 머신 러닝 학습 시 유용한 정보로 사용될 수 있으므로 문자열 전체 항목을 추출하여 실험할 필요도 있다. 따라서 최적의 이상상태 탐지 모델을 생성하기 위하여 본 논문에서는 다음과 같이 두 개의 방법으로 임베딩 및 이상상태 탐지를 진행한다.

- 전처리 대상 항목 Ⅰ: URL
- 전처리 대상 항목 Ⅱ: method, URL, status, http_version, referer, user-agent

4.2. 임베딩 기법

모델 생성을 위해 문자열을 수치화된 데이터로 표현하는 과정(벡터화)을 임베딩이라고 한다[5]. 벡터화하는 방법으로는 기존에는 원-핫 인코딩(One-hot Encoding)을 사용하지만, 데이터 간 연관 관계를 표현할 수 없다는 단점이 있고 이를 보완한 임베딩 기법이 Doc2Vec 모델과 오토인코더(Autoencoder)이다.

Doc2Vec은 Word2Vec의 방법론을 응용하여 하나 이상의 문장으로 구성된 문서 단위의 객체 내 내재된 맥락 정보를 학습하여 수치화된 벡터를 생성하는 비지도 학습 모델이다[6]. Word2Vec의 경우 임의의 벡터 공간에 각 단어에 대응되는 벡터를 생성하여 단어 간 의미론적 연관관계를 효과적으로 표현할 수 있지만, 하나 이상의 단어로 구성된 문장에 내재된 맥락 정보를 충분히 표현하지 못한다는 단점이 있다. Doc2Vec은 Word2Vec으로 생성된 단어 벡터와 문서 단위의 벡터를 연관시키는 학습으로 N개의 단어로 구성된 문서에서 1개의 문서 벡터와 N-1개의 단어들로 남은 하나의 단어를 예측하는 방식으로 학습한다.

오토인코더는 기본적으로 신경망을 이용해서 입력으로부터 계산되는 출력이 입력값과 비슷해지도록 학습한는 기법이다. 즉, 입력값을 중간층으로 인코딩한 후 다시 입력값과 같은 차원으로 디코딩하고 그 오차를 줄이는 방향으로 역전파를 한다. 이 때, 중간층의 가중치 벡터는 입력값의 중요한 성질들이 나타나므로 이를 벡터화에 이용한다. Seq2Seq은 문장을 벡터화하는 방법으로 RNN(Recurrent Neural Network) 기반으로 가변길이의 복수 Sequence를 벡터화하는 모델이다.

본 논문에서는 5장 실험을 통해 웹서버 로그 도메인에 적합한 임베딩 기법을 확인하고 파라미터(에폭, 단어 길이 등)를 결정하여 임베딩 모델을 생성한다.

4.3. 정상상태 모델 생성

정상상태 모델 생성은 정상행위로 구성된 데이터셋을 전처리 및 임베딩으로 벡터화한 후 이상치 탐지 기법의 지속적인 학습을 통해 최적의 파라미터를 생성하는 과정을 의미한다. 가장 잘 알려진 이상치 탐지 기법 중에 적용성이 뛰어난 기법이 Isolation Forest이다.

Isolation Forest는 비정상을 탐지하는 모델 중 하나로 일반적으로 알려진 비정상 탐지 모델과는 다른 접근법을 제안한다. 일반적인 비정상 탐지 모델은 정상 상태의 프로파일을 생성하고 이 프로파일을 기준으로 정상과 비정상을 구별한다. 이와 달리 Isolation Forest는 이상치(Outlier) 데이터의 특성을 고려한 모델로 이상치 데이터는 정상치 데이터에 비해 그 양이 적으며, 정상 데이터와는 다른 특성을 갖는다는 전제를 가지는 모델이다[7]. 학습을 위해서 전체 데이터 셋 중 일부를 샘플링하

여 하나 이상의 트리 모델을 생성하는 앙상블 기법으로 각 트리는 랜덤하게 데이터를 구분하며 데이터 셋의 모든 데이터가 단말 노드에 의해 구별된 유일한 데이터로 고립될 때까지 랜덤으로 분기를 반복한다. 하나의 트리가 완성되었을 때 이상치 데이터는 트리를 탐색하는 경로가 정상 데이터에 비해 경로 길이가 짧으며, 정상과 비정상 사이의 탐색 경로 길이를 임계값으로 정상과 비정상을 구분한다.

4.4. 이상상태 탐지

이상상태 탐지는 학습 단계에서 생성된 임베딩 모델과 정상상태 모델을 운용 환경에 적용하여 이상 여부를 판단하는 기능이다. 운용 환경에서 신규 웹서버 로그가 발생하면 로그 라인별로 정상상태 모델의 입력값으로 받아 각 라인마다 이상치를 산출하고 이상치가 특정 임계값보다 높으면 정상으로 식별하고 낮으면 비정상으로 탐지한다.

Ⅴ. 실험 및 평가

본 장에서는 제안 기법의 성능을 정량적으로 평가하기 위해서 정상행위와 비정상행위(예: 서버 대상 사이버 공격 및 단순 오용 등)를 복합 발생시키며 모델 성능을 분석한다. 모델 성능 지표는 표 3과 같다. 기본적으로 탐지율(TPR)이 높아야 하며 실제 운용 시 사용자의 편의성을 위해서 오탐률(FPR)이 낮아야 한다. AUROC의 경우는 1에 가까울수록 성능이 좋은 모델이다. 실험은 다음 두 단계로 진행한다.

(i) 임베딩 기법별 이상상태 탐지 실험

URL 항목을 추출하여 Doc2Vec과 오토인코더로 각각 임베딩한 후 Isolation Forest로 탐지 성능을 비교하여 웹서버 로그의 경향성을 보존하는 임베딩 기법을 확인한다.

(ii) 추출항목별 이상상태 탐지 실험

이상탐지 모델의 성능을 극대화하기 위해서 URL 항목 뿐 아니라 문자열 전체항목도 추출하여 (i) 실험으로 확인된 임베딩 기법으로 벡터화한 후 모델 성능을 비교한다.

Table. 3 Measurement

Type	Desc.
ACC (Accuracy)	Ratio of true positive(abnormal data) and true negative(normal data) to total predictions made
TPR (True positive rate)	Ratio of true positives to actual positives. It is also called to recall
FPR (False positive rate)	Ratio of false positives to actual negatives
Precision	Ratio of true positives to true positives and false positives
AUROC	Area under a ROC(receiver operating characteristic) curve

5.1. 임베딩 기법별 이상상태 탐지 실험

WSLD 데이터셋을 공개 라이브러리 gensim[8]을 통해서 임베딩한다. Doc2Vec의 경우 Doc2Vec의 기본 원리와 맞도록 정상 학습 시 학습용 정상 데이터의 단어만 사용한다. 임베딩 기법은 결과를 정량적으로 평가하는 방법은 없다. 대신, 임베딩 결과를 t-SNE로 시각화하여 군집 여부나 경향성 확인은 가능하다. t-SNE(t-Stochastic Neighbor Embedding)은 벡터 시각화를 위하여 자주 사용되는 알고리즘으로 고차원 벡터로 표현되는 데이터 간의 이웃 구조(Neighbor Structure)를 보존하여 2 차원 공간에서도 유사하도록 원 공간에서의 점들 간 유사도를 보존하면서 차원을 축소함으로써 고차원의 데이터를 2차원으로 표현하는 방법이다[9].

그림 3과 그림 4는 URL 항목을 Doc2Vec과 오토인코더로 임베딩 한 결과를 t-SNE로 시각화 한 것이다. 그림에서 알 수 있듯이 일부 불분명한 군집은 존재하나 전반적인 경향성을 확인할 수 있지만 정확한 임베딩의 기법의 비교를 위해서는 이상탐지 결과를 통해서 확인해야 한다[10].

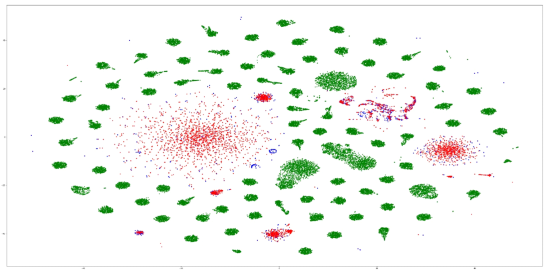

Fig. 3 t-SNE visualization for URL-then- Doc2Vec (green: normal data for learning, blue: normal data for validation, red: abnormal data for validation)

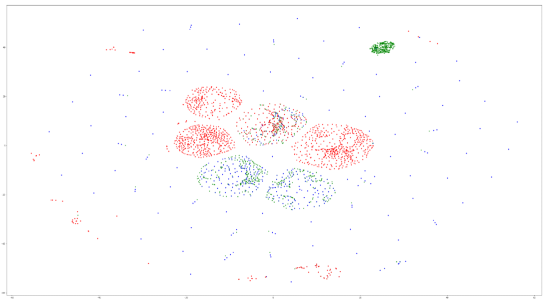

Fig. 4 t-SNE visualization for URL-then-Autoencoder (green: normal data for learning, blue: normal data for validation, red: abnormal data for validation)

Isolation Forest를 이용할 경우 앙상블 트리의 개수(Estimate)와 이상치(Outliner)의 기준을 설정해야 한다. 본 논문에서는 앙상블 트리 개수는 100, 이상치 기준은 0.1로 설정한다. 왜냐하면, 실험을 통해 트리가 100 이상이면 탐지 성능의 차이가 거의 발생하지 않았고 Isolation Forest는 설계상 이상치가 전체의 약 10% 존재한다는 가정하기 때문이다 [7].

그림 5와 그림 6은 각각 URL 항목에 대해서 Doc2Vec과 오토인코더 후 Isolation Forest를 진행한 ROC 커브이고 표 4는 실험 결과 비교표이다. 실험 결과를 통해 알 수 있듯이 URL 항목에 대한 Doc2Vec 및 Isolation Forest의 AUROC는 0.459으로 매우 낮은 반면 오토인코더 후 Isolation Forest는 0.963으로 높은 편이다. 이를 통해서 Doc2Vec보다 오토인코더가 웹서버 로그 데이터셋의 유효성 및 정확성을 높일 수 있는 임베딩 기법임을 확인할 수 있다. Doc2Vec은 설계 원리상 학습단계에서 해당 도메인의 단어로 사전을 생성하고 검증(Validation) 과정에서 사전에 없는 단어가 식별되면 Unknown으로 치환한다. 사이버공격의 특성상 알려지지 않은 공격에 해당되는 비정상행위의 단어를 미리 학습시키는 것은 현실적으로 불가능하기 때문에 Doc2Vec의 탐지결과가 낮은 것으로 판단된다. 따라서, Doc2Vec의 설계 원리상 한계점으로 인해서 Doc2Vec와 오토인코더의 실험결과가 현저히 차이가 나므로 추출항목별 이상상태 탐지에서는 Doc2Vec은 배제하고 오토인코더만 실험을 진행한다.

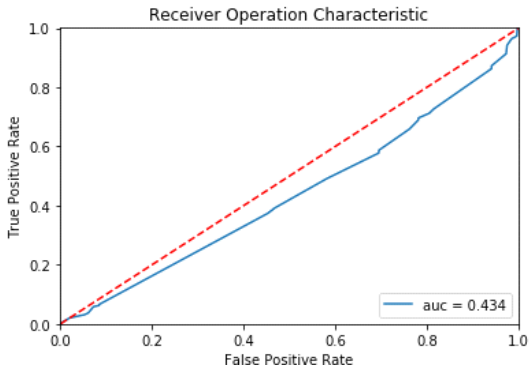

Fig. 5 ROC cuve of 'URL-Doc2Vec-IF' experiment

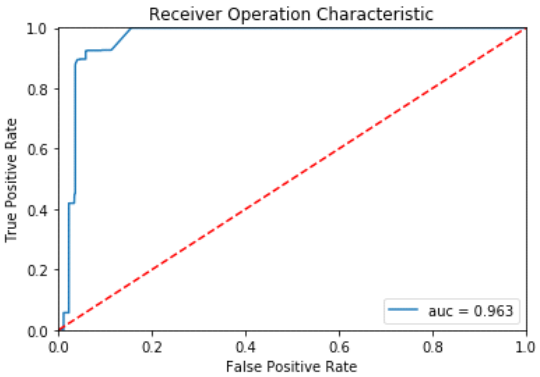

Fig. 6 ROC cuve of 'URL-Autoencoder -IF' experiment

5.2 추출항목별 이상탐지 탐지 실험

본 절에서는 상기 실험을 통해 확인한 바와 같이 웹서버 로그 도메인에 적합한 임베딩 기법인 오토인코더으로 문자열 전체 항목을 벡터화 한 후 Isolation Forest 탐지를 수행하였다. 이상상태 탐지를 위한 임계값은 동일하게 0.1로 설정한다. 실험 결과는 그림 7과 표 4와 같다. 실험을 통해 알 수 있듯이 문자열 전체 항목에 대한 오탐률은 0.102로 앞서 진행한 실험과 유사한 결과를 유지하면서 정탐률이 0.994로 향상되었고 AUROC 역시 0.983으로 향상되었다.

Table. 4 Comparison of novelty detection on the WSLD

		ACC	Precision	AUROC
URL only	Doc2Vec	0.513	0.558	0.459
	Autoencoder	0.913	0.902	0.963
All strings	Autoencoder	0.946	0.907	0.983

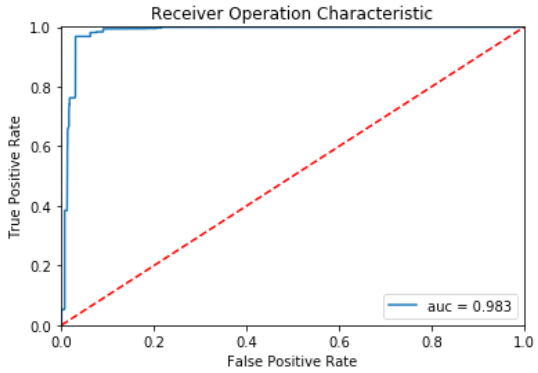

Fig. 7 ROC curve of 'Strings-Autoencoder -IF' experiment

실험을 통해서 Doc2Vec 보다 오토인코더가 웹서버 로그 데이터셋의 특성을 보존하면서 임베딩 되는 것을 확인하였다. 더불어, URL 단일항목보다는 문자열 전체 항목을 추출하여 임베딩 및 이상상태 탐지를 진행했을 때, 동일 오탐률을 유지하면서 정탐률을 높여 결과적으로 정확도 및 AUROC 그래프 성능을 향상시킴을 알 수 있다. 또한 Isolation Forest은 태생적으로 One-class SVM(Support vector machine)처럼 한 알고리즘 내 상호 영향을 끼치는 파라미터 쌍(nu, gamma)이 존재하여 적용 환경에 따라 파라미터 튜닝이 필수적인 기법이 아니다. 따라서 제안 모델은 비교적 높은 정확도를 제공하면서 운용 환경에 따른 별도의 튜닝 과정이 없어 적용이 용이하다는 장점이 있다. 더불어, 제안 기법과 II장의 관련 연구의 두 모델은 데이터셋 및 기법이 상이하여 절대적인 성능 비교는 어려우나, 설계상 제안 기법은 관련 연구에서 제시된 모델의 단점을 보완하였다. 먼저는 데이터셋의 경우 GET 뿐 아니라 POST 방식 데이터도 생성하였고, 정상데이터만으로 정상상태 모델을 생성하였으며 임베딩 시 문맥을 보존하기 위해서 문자열 순서까지 고려하여 벡터화를 진행하였다.

VI. 결 론

본 논문에서는 웹서버 로그 모의 환경을 구축하여 정상행위와 비정상행위로 구성된 자체 데이터셋 WSLD을 생성하였고, 실험을 통해서 웹서버 로그 도메인에서는 Doc2Vec보다 오터인코더가 더 적합한 임베딩 기법임을 확인하였다. 또한, URL 단일항목보다 문자열 전체 항목을 임베딩 후 Isolation Forest를 진행할 경우 오탐률은 동일한 수준으로 유지하면서 정탐률이 높이는 이상상태 탐지 모델이 생성됨을 확인하였다. 제안된 이상상태 탐지 모델은 알려지지 않은 공격이나 사용자의 단순 오용을 탐지할 수 있는 모델로 사용 가능하다.

REFERENCES

[1] Symantec Corporation. 2016. Internet security threat report.
[2] OWASP Top Ten Project, 2017 [Internet]. Available: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project.
[3] J. Liang, W. Zhao, and W. Ye, "Anomaly-Based Web Attack Detection: A Deep Learning Approach," *the VI International Conference on Network, Communication and Computing*. ACM, pp. 80-85, 2017.
[4] H. Mac, D. Truong, L. Nguyen, H. A. Tran, and D. Tran, "Detecting Attacks on Web Applications using Autoencoder," *the 9th Internationa Symposium on Information and Communication Technology*, Viet Nam, pp. 416-421, 2018.
[5] T. Mikolov, I. Sutskever, K. Chen, G. Corrando, and J. Dean, "Distirubuted representations of words and phrases and their compositionality." *Advances in neural information processing systems*, pp. 3111-3119, 2013.
[6] Q. Le, "Distributed Representations of Sentences and Documents," *International conference on machine learning*, vol. 32, pp. 1188-1196, Jun. 2014.
[7] F. T. Liu, K. M. Ting, and Z. Hua, "Isolation Forest," *the 8th IEEE International Conference on Data Mining*, pp. 413-422, 2008.
[8] Gensim, Last updated on July, 2019. [Internet]. https://radimrehurek.com/gensim/models/doc2vec.html.
[9] L. V. D. Maaten, and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579- 2695, 2008.
[10] H. Lee, K. S. Kim, and H. Kim, "Embedding Model Based on Web-server Log Dataset," *the Korea Institute of Military Science and Technology*, pp.1183-1184, 2019.

이화성 (Hwaseong Lee)

정보보호학과 공학박사
※관심분야 : 이상상태 탐지, 웹로그 기반 분석

김기수 (Ki Su Kim)

컴퓨터공학과 공학석사
※관심분야 : 이상상태 탐지, 웹로그 기반 분석