The Journal of The Institute of Internet, Broadcasting and Communication (IIBC)
Vol. 19, No. 1, pp.1-8, Feb. 28, 2019. pISSN 2289-0238, eISSN 2289-0246

https://doi.org/10.7236/JIIBC.2019.19.1.1
JIIBC 2019-1-1

귀납 추리를 이용한 침입 흔적 로그 순위 결정

Determination of Intrusion Log Ranking using Inductive Inference

고수정*

Sujeong Ko*

요 약 대량의 로그 자료로부터 가장 적합한 정보를 추출하기 위한 방법 중 귀납 추리를 이용한 방법이 있다. 본 논문에서는 디지털 포렌식 분석에서 침입 흔적 로그의 순위를 결정하기 위하여 귀납 추리를 이용한 방법 중 분류에 있어서 우수한 SVM(Support Vector Machine)을 이용한다. 이를 위하여, 훈련 로그 집합의 로그 데이터를 침입 흔적 로그와 정상 로그로 분류한다. 분류된 각 집합으로부터 연관 단어를 추출하여 연관 단어 사전을 생성하고, 생성된 사전을 기반으로 각 로그를 벡터로 표현한다. 다음으로, 벡터로 표현된 로그를 SVM을 이용하여 학습하고, 학습된 로그 집합을 기반으로 테스트 로그 집합을 정상 로그와 침입 흔적 로그로 분류한다. 최종적으로, 포렌식 분석가에게 침입 흔적 로그를 추천하기 위하여 침입 흔적 로그의 추천 순위를 결정한다.

Abstract Among the methods for extracting the most appropriate information from a large amount of log data, there is a method using inductive inference. In this paper, we use SVM (Support Vector Machine), which is an excellent classification method for inductive inference, in order to determine the ranking of intrusion logs in digital forensic analysis. For this purpose, the logs of the training log set are classified into intrusion logs and normal logs. The associated words are extracted from each classified set to generate a related word dictionary, and each log is expressed as a vector based on the generated dictionary. Next, the logs are learned using the SVM. We classify test logs into normal logs and intrusion logs by using the log set extracted through learning. Finally, the recommendation orders of intrusion logs are determined to recommend intrusion logs to the forensic analyst.

Key Words : SVM(Support Vector Machine), Forensic Analysis, Intrusion Log Ranking, Inductive Reasoning

Ⅰ. 서 론

디지털 포렌식에서 침입자의 흔적을 찾고자할 때 가장 중요한 것은 디지털 저장 장치로부터 디지털 증거를 찾는 것이다[1]. 디지털 포렌식 분석가는 디지털 기기로부터 정보를 획득하고 검색하며 중요한 증거를 보다 정확하게 추출할 수 있도록 분석하는 작업을 한다. 디지털 포렌식 분석가가 수많은 로그 자료로부터 사이버 범죄에 대한 증거자료로 사용할 포렌식 자료를 추출하고자할 때 분석 작업을 수작업으로 할 경우 시간과 노력면에서 비효율성이 나타난다. 따라서 수많은 로그 자료로부터 범죄의 증거자료로 채택하기 위한 자료를 추출하기 위하여 디지털 포렌식 분석 기술에 대한 더욱 많은 연구가 필요하다[2].

로그와 같은 대량의 자료로부터 적합한 정보를 분석하여 추출하기 위하여 기계 학습을 이용할 수 있다. 기계

*정회원, 인덕대학교 컴퓨터소프트웨어학과
접수일자 2019년 1월 15일, 수정완료 2019년 2월 3일
게재확정일자 2019년 2월 8일

Received: 15 January, 2019 / Revised: 3 February, 2019 / Accepted: 8 February, 2019
*Corresponding Author: sjko@induk.ac.kr
Dept. of Computer Software, Induk University, Korea

학습 방법 중 훈련 집합을 기반으로 목표 변수에 대한 일반적인 함수를 도출하는 알고리즘을 '귀납적 추론'이라 한다[3]. 귀납적 추론은 사실의 집합에 대해 사전의 가정들을 기반으로 하여 일반화를 생성하는 이론이다. 귀납적 추론을 이용하여 분석 작업을 수행할 수 있는 방법으로는 의사 결정 트리, 베이지안분류자, 인공 신경망, 가장 가까운 이웃을 기반으로 하는 방법 등이 있다[4].

본 논문에서는 디지털 포렌식 분석에서 침입 흔적 로그의 순위를 결정하기 위하여 귀납 추리를 이용하는 방법을 제안한다. 귀납 추리를 이용한 방법 중 분류에 있어서 우수한 방법인 SVM(Support Vector Machine)을 이용하여 로그들을 침입 흔적 로그와 정상 로그로 분류하고, 침입 흔적 로그의 가중치를 이용하여 침입 흔적 로그의 순서를 결정한다. 제안된 방법에서는 훈련 로그 집합을 대상으로 Apriori 알고리즘을 사용하여 연관 단어를 추출하여 연관 단어 사전을 생성하고, 생성된 사전을 기반으로 각 로그를 벡터로 표현한다. 다음으로, 벡터로 표현된 로그를 SVM을 이용하여 학습하고, 이를 기반으로 텍스트 로그 집합의 로그를 분류한다. 최종적으로 포렌식 분석가에게 침입 흔적 로그를 추천하기 위하여 분류된 침입 흔적 로그의 가중치를 이용하여 침입 흔적 로그의 추천 순위를 결정한다.

본 논문의 구성은 다음과 같다. 2장에서는 로그 특징 추출과 표현 방법을 기술하며, 3장에서는 로그 집합을 SVM을 이용하여 정상 로그와 침입 흔적 로그로 분류하는 방법을 기술한다. 4장에서는 성능 평가를 기술하고, 마지막으로 5장에서는 결론을 제시한다.

Ⅱ. 로그 특징 추출과 표현

로그 집합에 나타난 모든 필드를 로그의 표현에 사용한다면 로그를 벡터로 표현할 때 소요되는 시간이 과도하게 소비될 뿐 아니라 잡음 정보에 의해 분류의 정확도가 저하된다. 따라서 로그를 대상으로 전처리를 한 후에 벡터로 표현하는 과정이 필요하다.

1. 연관 단어 사전 생성

단어로 이루어진 사전을 만들기 위하여 마이크로소프트 웹서버로부터 로그를 수집하였다. 로그로부터 침입 흔적을 분석하기 위하여 로그 분석에 큰 영향을 미치는 필드만을 사용하기 위해 전체 필드 중 cs-method, cs-uri-query, 그리고 sc-status 등의 필드들을 추출하고, 이외의 나머지 필드는 불용 필드로 간주한다. 사용자가 입력한 정보인 cs-uri-query필드를 대상으로 전처리를 한다. 전처리를 하는 방법은 데이터를 정제하는 단계, 불용어를 처리하는 단계, 숫자를 필터링하는 단계, 어간을 추출하는 단계 등 여러 단계의 처리 과정을 거친다. 전처리를 완료하고 훈련 로그 집합에 속한 로그들을 전문가들이 수작업에 의해서 침입 흔적 로그와 정상 로그의 두 종류로 분류한다.

분류된 훈련 로그를 대상으로, Apriori 알고리즘을 실행하여 단어 사전을 생성한다[5]. Apriori 알고리즘은 연관 단어를 마이닝하기 위해 지지도(support)와 신뢰도(confidence) 값의 임계치를 결정해야 한다[6]. Apriori 알고리즘은 추출한 연관 규칙 중에서 규칙의 지지도와 신뢰도가 지정한 임계점보다 더 큰 값일 경우 침입 흔적이 있는 연관 규칙으로 지정하여 분류한다. 제안한 방법에서는 신뢰도와 지지도의 임계값을 모두 0.1로 지정하여 연관 단어를 추출한다. 임계값을 이와 같이 0.1로 지정한 이유는 해킹 흔적 로그를 추천할 경우에 빈도가 높은 연관 단어를 찾는 결과도 중요하지만 희소한 연관 단어가 해킹의 흔적을 나타내는 경우도 많기 때문에 빈도가 낮은 연관 단어도 포함해야한다.

특징이 비슷한 연관 단어로 구성하기 위하여 6개의 클래스를 정의하고, 마이닝된 연관 단어를 클래스별로 분류하여 연관 단어 사전을 구축한다.

2. 로그의 벡터 표현

본 논문에서는 로그 표현 형태로서 정보 검색 분야에서 사용되는 단일 단어 벡터 모델[5]을 응용한 연관 단어 벡터 모델 방법을 채택한다. 연관 단어 기반의 벡터 모델을 이용하는 방법은 필드의 특징을 연관 단어로 표현하기 때문에 단어 간의 중의성 문제로 인해 발생하는 사용자의 혼란을 줄일 수 있다. 식 (1)은 p개의 벡터로 구성된 로그 log_j의 특징을 정의한다.

$$log_j = \{F_1, F_2, AW_1, AW_2, ..., AW_k, ..., AW_p\} \quad (1)$$

log_j = {sc-method: 분류코드, sc-status : 분류코드, $Class_1$ 연관단어가중치, $Class_2$ 연관단어가중치, ..., $Class_k$ 연관단어가중치, ... $Class_p$ 연관단어가중치}

식 (1)의 log_j는 로그의 중요 필드와 6개의 클래스로 구분한 연관 단어들의 가중치로 구성된다. 제안한 방법에서는 6개의 클래스로 연관 단어를 분류하므로 p는6이다.

표 1은 식 (1)에 의해 표현된 로그의 특징 벡터로 표현된 로그들의 예를 나타낸다.

표 1. 특징 벡터에 의해 표현된 로그의 예
Table 1. Examples of logs expressed by feature vectors

	cs-method (F1)	sc-status (F2)	AW_1	AW_2	AW_3	AW_4	AW_5	AW_6
log_1	1	1	1	0.6	0.3	0.2	0	0
log_2	1	1	1	0.4	0.4	0	0	0
log_3	1	1	1	0.3	0.1	0	0	0
log_4	2	1	1	0.6	0.3	0.2	0	0
log_5	1	2	1	0.3	0	0	0.5	0
log_6	1	2	1	0.5	0.4	0	0	0
log_7	1	2	1	0.7	0.1	0.6	0	0
log_8	1	3	1	0.6	0.3	0.2	0	0
log_9	1	3	1	0.5	0.4	0	0	0

Ⅲ. 귀납 추리를 이용한 로그 분류

로그를 벡터로 표현한 후 이를 대상으로 SVM을 이용하여 침입 흔적 로그와 정상 로그로 분류하고, 침입 흔적 로그로 분류된 로그에 대해 침입 흔적에 대한 가중치가 얼마인가를 계산하여 침입 흔적에 대한 순위를 결정한다.

1. SVM

최근에 패턴분류에 있어서 많은 분야에서 응용되고 있는 SVM 모델은 1995년 Vapnik에 의해 개발된 통계적 학습 이론이다[7]. 본 논문에서 제안한 방법은 SVM을 이용하여 침입 흔적 로그 집합과 정상 로그 집합을 대상으로 학습하여 그 결과를 기반으로 테스트 로그를 이원 분류하는 방법을 사용한다. 고차원 매핑을 통해 비선형 문제를 선형화하여 해결하면서 커널 함수를 통해 계산량 문제를 해결하는 방법을 커널법(kernel method)이라고 하며 SVM을 비롯하여 선형성을 가정하는 방법론에서 최근 활발히 사용되고 있다. SVM을 비롯한 여러 응용에서 주로 사용되는 커널은 표 2와 같다[8].

표 2. 커널의 종류 및 커널별 함수
Table 2. Kernel types and kernel-specific functions

커널	함수
선형커널 (Linear kernel)	$\kappa(x,y)=(x \cdot y)$
다항식커널 (Polynomial kernel)	$\kappa(x,y)=(x \cdot y+d)^{\alpha}$
가우시안 RBF (Gaussian Radial Basis Function kernel)	$\kappa(x,y)=\exp\left\{-\frac{\parallel x-y \parallel^2}{2\gamma^2}\right\}$

학습하기 전 N개의 입출력 쌍으로 이루어진 훈련 로그 집합 $\{X=\{(log_i, c_j)\}i = 1,..,N\}$이 주어지면, 하이퍼 파라미터 C와 커널 함수$\kappa(\log_i,\log_j)$를 정의한다. 이때 log_i는 두 개의 클래스 중에서 하나로 분류되며, $c_j \in \{-1,|1\}$는 해당 클래스를 표시하는 라벨을 표시한다. 여기서 '-1'의 값은 정상 로그를 의미하며 'not_hacking_log'로 표기하고, 1은 침입 흔적 로그로 'hacking_log'로 표현한다.

2. 로그 분류 및 순위 결정

SVM을 이용하여 침입 흔적이 있는 로그인가 아닌가를 분류하기 위해서 훈련 로그 집합과 테스트 로그 집합에 Apriori 알고리즘을 적용하여 식 (1)과 같이 로그를 표현하였다. 표 3은 훈련 로그 집합에 대하여 특징으로 표현한 후 침입 흔적 로그(hacking_log)인가 정상 로그(not_hacking_log)인가를 표현한 예이다.

표 3. 특징으로 표현한 로그를 분류한 훈련 집합의 예
Table 3. An example of a training set that classifies logs by their characteristics

cs-method	sc-status	AW1	AW2	AW3	AW4	AW5	AW6	분류
1	4	0.9	0.3	0.1	0	0	0	hacking_log
1	4	0.6	0.3	0.2	0.4	0	0	hacking_log
1	4	0.9	0.5	0.4	0	0	0	hacking_log
1	4	0.7	0.3	0	0	0.5	0	hacking_log
1	4	0.6	0.3	0.2	0.4	0	0	hacking_log
1	1	1	-0	0	0	0	0	not_hacking_log
1	1	-0	0	0	0	0	0	not_hacking_log
1	1	1	-0	0	0	0	0	not_hacking_log
2	1	1	-0	0	0	0	0	not_hacking_log

표 3과 같이 식 (1)의 특징으로 표현하여 분류한 학습 로그 집합을 기반으로 표 2의 3가지 커널법을 각각 적용하여 테스트 로그 집합의 로그를 침입 흔적 로그와 정상

로그로 분류할 수 있다.

표 2에 제시된 3가지 커널법에서는 학습 과정이 이루어지기 전에 사용자가 직접 파라미터를 결정하며, 그 파라미터 값에 따라 분류의 성능은 달라진다. 파라미터 중 오류 패널티 변수 C값은 3가지 커널법 모두에 적용되는 값으로 학습 과정에서 마진폭과 분류 오류 간의 타협점을 찾아 주는 값이다. 또한, RBF커널법에서는 C외에 γ (감마) 파라미터 값의 변화에 의해서도 그 성능은 달라진다. 다항식커널법에서는 C값과 degree에 의해 성능이 달라진다.

본 논문에서 제안한 방법에서는 C값과 degree, 감마 파라미터의 결정을 위하여 침입 흔적 로그 5000개를 대상으로 선형커널함수에서는 C값을 변경하고, RBF커널함수에서는 C값과 감마값을 변경하며, 마지막으로 다항식커널함수에서는 C값과 degree를 변경해가며 커널함수별 오분류율을 측정하였다. 표 4는 C값을 1부터 1씩 증가해가며 15까지 선형커널법, RBF커널법, 다항식커널법의 정확도를 카파계수[9]에 의해 평가한 결과이다.

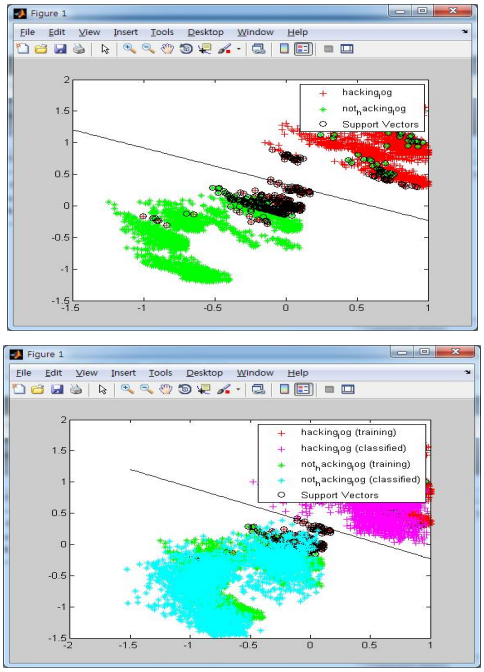
표 4. C값의 변화에 따른 분류의 정확도
Table 4. Accuracy according to change of C value

C	선형커널	다항식커널 (degree=1)	가우시안 RBF (감마=0.1)
1	0.672	0.721	0.73
2	0.673	0.726	0.731
3	0.681	0.729	0.738
4	0.685	0.731	0.747
5	0.689	0.734	0.749
6	0.69	0.736	0.751
7	0.691	0.739	0.759
8	0.693	0.743	0.761
9	0.696	0.746	0.771
10	0.697	0.749	0.78
11	0.699	0.75	0.78
12	0.703	0.75	0.78
13	0.705	0.75	0.78
14	0.709	0.75	0.78
15	0.710	0.75	0.78

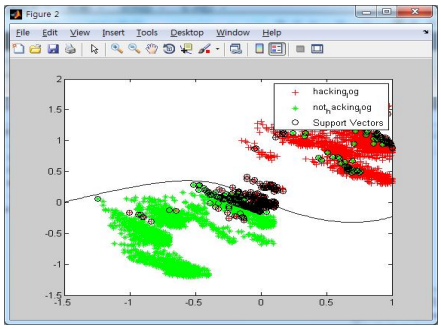
표 4에서 선형커널법은 C값이 15일 경우 가장 높은 성능을 나타냈고, RBF커널법에서는 C의 값이 10일 경우 가장 높은 성능을 나타내었다. 또한, 다항식커널법은 C의 값이 11일 경우 가장 높은 성능을 보였다.

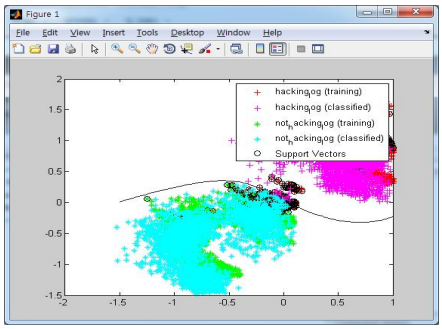
그림 1은 선형커널함수, 다항식커널함수, 가우시안 RBF커널 함수를 사용하여 훈련 로그 집합과 테스트 로그 집합을 대상으로 로그를 분류한 그림을 나타낸다. 그림 1에서 선형커널은 C값을 15로 고정시키고 분류하였다. 다항식커널에서는 C값을 10으로 고정시키고 degree를 1,2,3으로 변화시켜가면서 분류한 결과, 1의 degree일 경우 가장 오분류가 낮음을 볼 수 있었다. 가우시안 RBF커널에서는 감마값을 0.01로 설정하였을 경우에 선형커널과 다항식커널과 비슷한 성능을 보였다. 반면, 0.5로 설정하였을 경우 미탐지율이 높았으나 상대적으로 오탐지율이 증가하였다. 그래서 감마값을 0.1로 지정하였다. 전반적으로, RBF커널을 사용한 분류는 다항식커널, 선형커널보다는 높은 성능을 보였다.

훈련 로그 집합에서 침입 흔적 로그로 분류한 로그들의 중앙을 나타내는 벡터를 구하고, 이 벡터와 RBF커널을 사용하여 분류한 침입 흔적 로그 벡터와의 코사인 유사도[10] 값을 계산한다. 계산한 결과를 침입 흔적의 가중치로 간주하고, 가중치가 높은 로그를 추천의 우선 순위가 높은 로그로 결정한다.

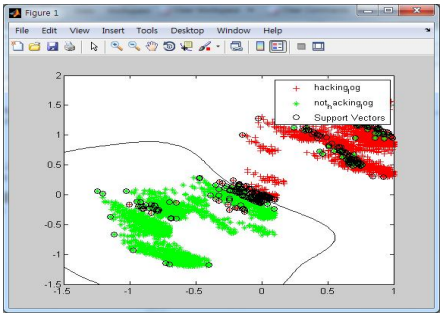


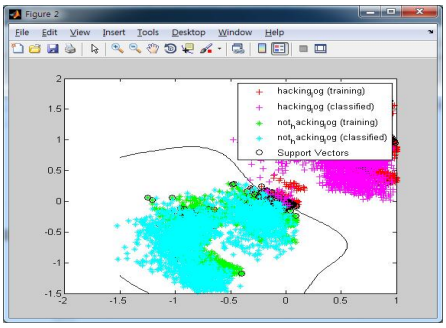
a. Polynomial kernel function-Training log set
b. Polynomial kernel function-Test log set





c. Linear kernel function-Training log set
d. Linear kernel function-Test log set





e. Gaussian RBF kernel function-Training log set
f. Gaussian RBF kernel function-Test log set

그림 1. 선형, 다항식, 가우시안 RBF커널 함수를 사용한 침입 흔적 로그 집합 분류
Fig. 1. Classification of intrusion log set using linear, polynomial, gaussian RBF kernel functions

Ⅳ. 성능 평가

본 논문에서는 제안한 귀납 추리를 이용한 침입 흔적 로그 순위 결정 방법(Inductive_R)의 성능을 평가하기 위하여 정규 표현을 사용하는 개념을 기반으로 로그 파일을 분석하는 방법(Regular_E)[11], 악의적인 활동을 분석하기 위하여 로그 파일과 퍼지 규칙을 사용하는 방법(Fussy)[12], 문자열 시각화를 이용하는 방법(Visualization_S)[13] 등의 방법과 로그 수를 변화시키면서 비교하였다. 2018년 3월 1일부터 2018년 3월 15일까지 총 15일간 수집한 웹로그를 대상으로 전처리를 한 후, RSM(Rank Scoring Metric)[14]과 카파계수(Cohen's Kappa)[9]의 척도를 사용하여 성능을 평가하였다.

1. RSM을 이용한 성능 분석

RSM은 로그들이 침입 흔적 로그로 분류할 확률이 목록의 하단으로 갈수록 지수값으로 감소한다는 전제에서 계산한다. 각 로그는 침입 흔적 가중치에 따라 내림차순으로 j에 의해 순서대로 정렬되어있다고 가정한다. 식 (2)는 순위가 부여된 침입 흔적 로그에 대해 조사관 U_a가 순위스코어 측정값의 기대이용도(Expected utility)를 계산한다.

$$R_a = \Sigma_j \frac{\max(w_{a,j} - d, 0)}{2^{(j-1)/(\alpha-1)}} \qquad (2)$$

식 (2)에서 w_{aj}는 조사관 U_a가 침입 흔적 로그로 판별한 log_j의 침입 흔적 가중치를 나타낸다. 또한, α는 반감기(halflife)이며, d는 침입 흔적 로그의 가중치 대한 중간값이다. 반감기는 침입 흔적 로그로 평가될 확률이 50%인 목록에 있는 로그의 수이다. 그림 2는 반감기를 2에서 10까지 증가시키면서 ROC 곡선[15]을 그린 결과이다. 실험 결과 반감기 8일 경우 가장 높은 성능을 보이므로 반감기를 8로 지정하고 실험을 실시하였다.

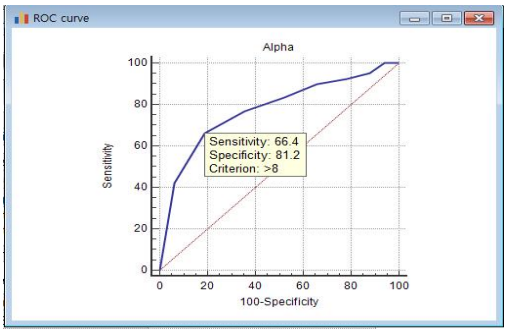


그림 2. 반감기를 2에서 10까지 증가시키면서 변화된 ROC 곡선

Fig. 2. Change in ROC curve with increasing half-life from 2 to 10

식 (3)은 순위 스코어 척도를 사용하여 분류한 로그의 분류에 대한 판단의 정확도를 나타내는 식이다.

$$R = 100X\frac{\sum_u R_u}{\sum_u R_u^{\max}} \qquad (3)$$

식 (3)에서 R_u^{Max}는 침입 흔적 로그가 순위가 있는 목록에서 상위에 배정된 경우에 측정된 RSM에 대한 기대 이용도의 최대값이다.

그림 3은 로그의 수를 2000부터 20000개로 증가시켜 가며 식 (3)을 기반으로 계산한 Inductive_R, Regular_E, Fussy, Visualization_S의 순위 스코어의 비교 결과를 나타낸다.

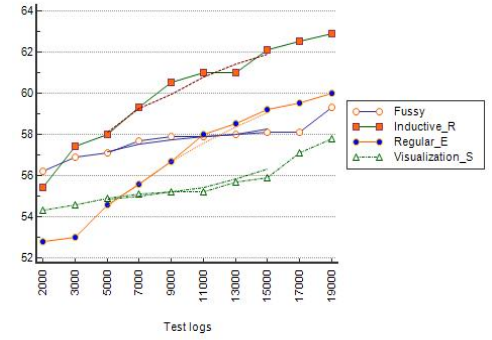


그림 3. 로그수의 변화에 따른 순위 스코어

Fig. 3. Rank score according to change of log number

그림 3에서 Inductive_R, Fussy, Regular_E, Visualization_S 방법의 곡선 형태를 분석하면, 침입 흔적 로그를 추천한 Inductive_R의 방법이 테스트 로그의 수가 증가함에 따라 가장 높은 기대이용도를 나타내었다. 또한, Regular_E와 같이 정규화 표현을 이용한 방법은 로그에 대한 전처리의 정확도가 높아짐에 따라 추천의 정확도 또한 높음을 볼 수 있다. 반면, Fussy를 이용한 방법은 침입 흔적 로그와 정상 로그를 분류하는 데 있어서의 경계값을 결정하는 문제점이 있어서 다소 기대이용도가 낮음을 볼 수 있다. Visualization_S의 방법은 특징의 모호성 등에 대한 처리 문제가 있어 낮은 결과를 나타낸다.

2. 카파계수(Cohen's Kappa)를 이용한 성능 평가

카파계수는 Cohen[9]에 의해 제안된 두 명의 관찰자의 일치성을 나타내는 지표로, 두 관찰자 사이의 측정 범주값의 일치도를 측정하는 척도이다. 표 5는 카파계수를 계산하기 위한 오차행렬(confusion matrix)이다. 오차행렬에서는 클래스를 침입 흔적 로그 클래스와 정상 로그 클래스로 구분한 후 실제 침입 흔적 로그의 수를 a, 실제 정상 로그의 수를 b, 침입 흔적 로그로 예측한 수를 c, 정상 로그로 예측한 수를 d로 정의한다.

표 5. 카파계수 평가를 위한 오차행렬

Table 5. Confusion matrix for evaluating kappa coefficients

		Predicted class		
		Clog1 (Hacking log)	*Clog2* (Normal log)	Total
Correct class	*Clog1* (Hacking log)	a	b	a+b=$Clog1_{corr}$
	Clog2 (Normal log)	c	d	c+d=$Clog2_{corr}$
	Total	a+c=$Clog1_{pred}$	b+d=$Clog2_{pred}$	N

표 5의 오차행렬을 이용하여 카파계수를 식 (4)와 같이 계산할 수 있다.

$$Cohen's\ \kappa = \frac{N\sum_{i=1}^{m} ClogM_{ii} - \sum_{i=1}^{m} Clogi_{corr}Clogi_{pred}}{N^2 - \sum_{i=1}^{m} Clogi_{corr}Clogi_{pred}} \qquad (4)$$

식 (4)에서 $ClogM_{ii}$는 Table 6의 오차행렬에서 대각선 요소를 나타낸다.

그림 4는 식 (4)의 카파계수를 이용하여 Inductive_R, Regular_E, Fussy, 그리고 Visualization_S의 성능을 계산한 결과를 나타낸다. 그림 4에서 Inductive_R, Fussy, Regular_E, Visualization_S 방법은 순위 스코어의 방법과 같이 SVM을 이용하여 학습한 후 분류한 Inductive_R의 방법이 가장 높은 성능을 나타냈다. Fussy, Regular_E, Visualization_S의 방법은 침입 흔적 로그와 정상 로그로 분류하는 데 있어서의 전처리 과정에서 다양한 학습을 하지 않으므로 테스트 로그에 대한 분류에서도 다소 성능이 낮음을 볼 수 있다.

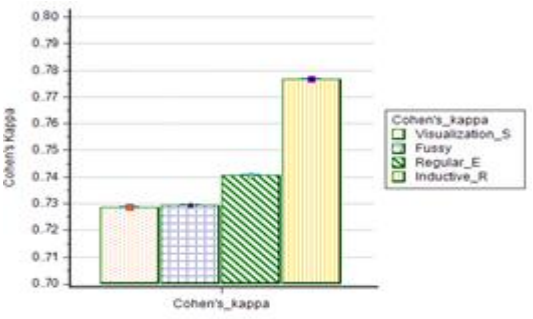


그림 4. 카파계수를 이용한 성능평가 결과
Fig. 4. Performance evaluation results using kappa coefficients

V. 결 론

귀납 추리를 이용한 침입 흔적 로그의 순위를 결정하여 포렌식 분석가에게 추천하는 방법은 다음과 같은 면에서 장점을 갖는다. 첫째, 분류 분야에 있어서 우수한 성능을 보이는 SVM을 이용하여 로그를 분류함으로써 분류의 정확도를 높였다. 둘째, 순위를 결정하여 추천함으로써 포렌식 분석가가 분석에 필요한 시간을 단축시킬 수 있다. 셋째, 로그의 특징을 추출하기 위한 전처리로 Apriori 알고리즘을 사용하여 연관 단어 사전을 생성하고, 이를 기반으로 로그를 벡터로 표현함으로써 로그를 정확하게 분류할 수 있다. 마지막으로, 학습된 결과를 기반으로 로그를 분류하고, 로그의 가중치를 이용하여 로그의 순위를 귀납적으로 추론함으로써 추천의 정확도를 높일 수 있다.

향후, 침입 흔적 로그의 가중치를 지정할 경우 코사인 유사도 뿐 아니라 보다 다양한 유사도 방법을 사용할 경우 성능을 비교하는 연구가 필요하다.

References

[1] D. Hong, S. Jeon, C. Kim, H. Kim, “Analysis of Digital Forensics Technology Trends Based on Big Data,“ Journal of The Korea Knowledge Information Technology Society(JKKITS), pp. 51-63, Vol. 9, No. 1, 2014.
UCI : G704-SER000001483.2014.9.1.018

[2] N. Kumar S. Deepak S. Tomar, and B. Nath Ray, “An Approach to Understand the End User Behavior through Log Analysis,” International Journal of Computer Application, Vol. 5, No. 11, 2010.
DOI: https://doi.org/10.5120/953-1330

[3] Antonio J. Tallón-Ballesteros and José C. Riquelme, “Data Mining Methods Applied to a Digital Forensics Task for Supervised Machine Learning,” Vol. 555, pp. 413-428, Studies in Computational Intelligence, 2014.
DOI: https://doi.org/10.1007/978-3-319-05885-6_17

[4] Feelders A., Verkooijen W., “On the Statistical Comparison of Inductive Learning Methods,” In: Fisher D., Lenz HJ. (eds) Learning from Data, Lecture Notes in Statistics, Vol. 112, Springer, 1996.
DOI: https://doi.org/10.1007/978-1-4612-2404-4_26

[5] Tamas Abraham, Olivier de Vel, “Investigative Profiling with Computer Forensic Log Data and Association Rules,” In Proceeding of IEEE International Conference on Data Mining(ICDM), 2002.
DOI: https://doi.org/10.1109/icdm.2002.1183880

[6] Sujeong Ko, “A Text Mining-based Intrusion Log Recommendation in Digital Forensics”, KIPS Transactions on Computer and Communication Systems, Vol. 2, No. 6, pp. 279-290, 2013.
DOI: https://doi.org/10.3745/KTCCS.2013.2.6.279

[7] S. Kim, J. Lee, “A Study on Face Recognition

using Support Vector Machine," The Journal of the Institute of Internet, Broadcasting and Communication, Vol. 16, No. 6, pp.183-190, 2016.
DOI: https://doi.org/10.7236/JIIBC.2016.16.6.183

[8] Z. Zeng, S. Zhu, "A kernel-based sampling to train SVM with imbalanced data set," IEEE Conference Anthology, pp. 1-5, 2013.
DOI: https://doi.org/10.1109/ANTHOLOGY.2013.6784693

[9] Cohen, J., "A Coefficient of Agreement for Nominal Scales," Educational and Psychological Measurement, Vol. 20, pp. 37-46, 1960.
DOI: https://doi.org//10.1177/001316446002000104

[10] R. Jalam, O. Teytaud, "Kernel-based text categorisation," International Joint Conference on Neural Networks, Proceedings (Cat. No.01CH37222), pp. 1891-1896, Vol. 3, 2001.
DOI: https://doi.org/10.1109/IJCNN.2001.938452

[11] Chang, Joong-Hyuk, "Finding high utility old itemsets in web-click streams," Journal of the Korea Academia-Industrial cooperation Society, Vol. 17, No. 4, 2016.
DOI: 10.5762/KAIS.2016.17.4.521

[12] D. Meena1, H. Gupta, "Digital Crime Investigation using Various Logs and Fuzzy Rules: A Review," International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 4, 2013.

[13] Heidi Lam, Daniel M. Russell, and Diane Tang, "Visual Exploratory Analysis of Web Session Logs," Symposium on Visual Analytics Science and Technology (VAST), IEEE, pp. 147-154, 2007.
DOI: https://doi.org/10.1109/VAST.2007.4389008

[14] Herlocker, J., Konstan J., Terveen L., and Riedl J., "Evaluating Collaborative Filtering Recommender Systems," ACM Transactions on Information Systems (TOIS) TOIS Homepage archive, Vol. 22, Issue 1, 2004.
DOI: https://doi.org/10.1145/963770.963772

[15] David Faraggi, Benjamin Reiser, "Estimation of the area under the ROC curve," STATISTICS IN MEDICINE, Vol. 21, pp. 3093-3106, 2002.
DOI: https://doi.org/10.1002/sim.1228

저자 소개

고 수 정(정회원)

- B.S, M.D, Ph.D in Dept. of Computer Science, Inha University, 1990-2002.
- Post Doc., University of Illinois at Urbana Champaign. 2003
- Research Scientist, Colorado State University, 2004
- Professor in Dept. of Computer Software, Induk University, 2005~Present
- 관심분야 : Information Security, Data mining, Big data, IoT