Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.35, NO.1, Feb. 2025 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2025.35.1.109

슬라이딩 윈도우 기반 BERT를 활용한 AWS CloudTrail MITRE ATT&CK 공격 탐지

박 현 준,[1†‡] 차 원 제,[2] 최 유 정,[3] 김 태 양,[4] 김 지 윤,[3] 신 예 지[5]
[1]아주대학교 (학생), [2]국립창원대학교 (학생),
[3]덕성여자대학교 (학생), [4]중앙대학교 (학생), [5]수원대학교 (학생)

Anomaly Detection of MITRE ATT&CK Techniques in AWS CloudTrail Using BERT-Based Sliding Window Approach

Hyun-jun Park,[1†‡] Won-je Cha,[2] Yu-jeong Choi,[3]
Tae-yang Kim,[4] Ji-yun Kim,[3] Ye-ji Shin[5]
[1]Ajou University (College student), [2]Changwon National University (College student),
[3]Duksung Women's University (College student),
[4]Chung-ang University (College student), [5]University of Suwon (College student)

요 약

현대 클라우드 환경에서 잠재적인 보안 위협을 식별하는 것은 매우 중요한 과제로 여겨진다. 본 연구에서는 AWS CloudTrail 로그를 분석하여 MITRE ATT&CK 전술을 예측하기 위한 슬라이딩 윈도우 기반 BERT 모델을 제안한다. 슬라이딩 윈도우 기법 시퀀스를 통해 로그 분할하며 이를 바탕으로 모델이 시계열적 및 문맥적 의존성을 포착할 수 있도록 한다. 이러한 접근법은 BERT의 문맥 이해 능력을 활용하여 정밀한 로그 이벤트 분류를 가능하게 한다. 실험 결과, 제안된 프레임워크는 공격 전술을 식별하는 데 0.933의 f1-score를 달성하였다. 본 연구는 AWS CloudTrail 로그 데이터를 기반으로 MITRE ATT&CK 프레임워크의 전술과 기술을 효과적으로 예측하기 위해 슬라이딩 윈도우 기반 BERT 모델을 활용하는 접근법의 중요성을 강조한다.

ABSTRACT

Identifying potential security threats in the modern cloud environment is considered a very important task. In this study, we propose a sliding window-based BERT model for predicting MITRE ATT&CK tactics by analyzing AWS CloudTrail logs. We divide the log sequence through the sliding window technique, based on which the model can capture time-series and contextual dependence. This approach enables precise log event classification by leveraging BERT's ability to understand the context. As a result of the experiment, the proposed framework achieved an f1-score of 0.933 in identifying attack tactics. This study highlights the importance of the approach using the sliding window-based BERT model to effectively predict the tactics and techniques of the MITRE ATT&CK framework based on AWS CloudTrail log data.

Keywords: MITRE ATT&CK, BERT, Sliding Window Algorithm

Received(12. 12. 2024), Modified(02. 03. 2025), Accepted(02. 03. 2025)

† 주저자, bolkv1101@ajou.ac.kr
‡ 교신저자, bolkv1101@ajou.ac.kr(Corresponding author)

I. 서 론

클라우드 인프라가 점점 더 복잡해지면서 잠재적인 보안 위협을 식별하고 대응하는 것은 매우 중요한 과제가 되었다. AWS CloudTrail의 로그는 사용자 활동에 대한 정보를 제공하며 위협 탐지에 중요한 역할을 한다. AWS CloudTrail 로그는 API 호출, 사용자 작업, 리소스 접근 등과 같은 이벤트 정보를 상세히 기록하여 보안 분석의 핵심이 된다.

기존에는 머신러닝 기반의 접근법을 통해 로그를 분석하여 위협을 탐지하는 데 활용되었다. 이러한 방법은 로그에서 유용한 특징을 추출하고 이를 머신러닝 알고리즘을 통해 이상 패턴을 분석한다. 하지만 이러한 기존 접근법은 로그 데이터의 시계열적 특성과 이벤트 간 관계를 충분히 학습하지 못한다. 따라서 지능형 지속 위협(APT) 등의 위협을 탐지하는 데 한계가 있다. 최근 딥러닝 RNN(Recurent Neural Network)을 활용한 연구는 시퀀스 데이터를 모델링하는 데 있어 많은 가능성을 보여주었다.

LSTM(Long Short-Term Memory) 및 GRU(Gated Recurrent Unit)와 같은 RNN 기반 모델은 로그 시퀀스의 정상적인 패턴을 학습한다. 이를 기반으로 이상 로그를 탐지하는 데 사용되었다. 그러나 RNN은 일반적으로 한 방향으로만 데이터를 처리하기 때문에 로그 이벤트가 양방향 문맥에 의존하는 경우 학습이 제한된다. 뿐만 아니라RNN은 국소적 패턴에 대해 초점을 맞추게 되어 다음 이벤트 예측을 목표로 학습된다. 이는 전체 시퀀스의 문맥 정보를 충분히 학습하지 못하게 만든다.

이러한 문제를 해결하기 위해 본 연구에서는 BERT와 슬라이딩 윈도우 기법을 결합하여 AWS CloudTrail 로그에서 MITRE ATT&CK 전술(Tactic) 및 기법(Technique)을 예측하는 접근법을 제안한다. 슬라이딩 윈도우 기법은 로그 데이터의 시계열 구조를 유지하도록 하고 BERT의 self-attention 메커니즘은 윈도우 내의 로그 데이터의 문맥적 관계를 학습하여 높은 정확도의 위협 예측을 가능하게 한다. 이러한 기법을 통해 IAM 관련 공격 데이터에 대해 MITRE ATT&CK의 기법을 탐지하는 실험을 수행한 결과, 평균 f1-score 정확도 0.933을 달성하였다.

II. 관련 연구

2.1 접근

시스템 로그는 문제 해결과 보안 모니터링을 위해 널리 사용되며, 로그 메시지는 일반적으로 정형적인 구조의 텍스트 문자열로 구성된다. 기존 접근법에서는 규칙 기반 시스템과 키워드 매칭 방식이 주로 사용되었다. 이러한 방법은 사전에 정의된 패턴을 기반으로 위협을 탐지하는 데 효과적이지만 복잡한 상관관계를 포함한 위협 탐지에 한계를 보였다. [1]. 머신러닝 기반 접근법은 로그 데이터에서 특징을 추출한 후, 이를 학습하여 이상 패턴을 탐지한다. 예를 들어, Principal Component Analysis (PCA)[2]나 One-Class Support Vector Machines (OC-SVM)[3]과 같은 비지도 학습 방법이 이상 탐지에 활용되었다. 하지만 이러한 방법은 로그 데이터의 시계열적 의존성을 학습하지 못하는 한계가 존재한다.

2.2 딥러닝 기반 모델

딥러닝 모델은 로그 데이터의 시계열적 특성을 학습하는 데 있어 뛰어난 성능을 보였다. DeepLog[4]와 LogAnomaly[5]와 같은 연구는 RNN 기반 모델을 활용하여 정상 로그 시퀀스를 학습하고 이상 패턴을 탐지했다. 이러한 모델은 로그 데이터의 단기 의존성을 학습하는 데 효과적이었다.

[6]은 로그의 preorder 및 postorder 관계를 학습하는 대칭 구조와 이중 LSTM 기반 모델 LogLS를 제안한다. 이는 기존 DeepLog[4]의 단점을 보완하여 긴 로그 시퀀스의 예측 성능을 개선하였으며 HDFS 데이터셋에서 높은 F1 점수를 기록하였다. 하지만 위에서도 언급했듯이 RNN은 이벤트 간의 단방향 의존성만 학습하기 때문에 로그 이벤트 간의 전체 문맥 정보를 반영하지 못한다. 또한 RNN의 순차적 처리 특성으로 인해 긴 로그 시퀀스를 처리하거나 병렬화에 어려움이 존재한다.

2.3 트랜스포머 기반 모델

BERT는 Self-Attention 메커니즘을 통해 양방향 문맥 정보 학습에 뛰어난 성능을 보이며 자연어 처리 분야에서 대표적인 모델로 자리 잡았다[7]. 로

그 분석 분야에서는 LogBERT와 같은 연구가 BERT를 활용하여 로그 데이터의 문맥적 관계를 학습하고 이상로그 탐지에 성공적인 성능을 보였다[8]. LogBERT는 마스킹된 로그 키를 예측하거나, 정상 로그 시퀀스의 패턴을 학습하여 비정상 로그를 탐지한다. 하지만 LogBERT는 긴 시퀀스 처리의 효율성과 시계열적 특성을 명시적으로 반영하지 못하는 한계를 보였다.

우리는 앞서 언급된 기존 연구의 한계점을 보완하기 위해 슬라이딩 윈도우 기반 BERT 모델을 제안한다. 이 접근법은 긴 로그 시퀀스를 작은 윈도우 단위로 나누어 처리함으로써 기존 연구에서 제기된 긴 시퀀스 처리의 비효율성과 문맥적 정보 학습의 제한을 해결할 수 있다.

III. 학습 데이터 구축 및 데이터 전처리

3.1 로그 데이터 처리

AWS CloudTrail 로그는 클라우드 환경에서 발생하는 모든 이벤트를 기록하며, 보안 분석 및 이상 탐지에서 중요한 정보를 제공한다. 본 연구에서는 로그 데이터의 다양한 필드 중 다음과 같은 필드들을 주요 분석 대상으로 선정하여 활용하였다.

학습에 사용된 필드들은 오픈소스 Offensive Tool에서 제공하는 공격 시나리오에 따라서 결정하였다. 이러한 필드들은 이벤트의 맥락 정보를 학습하고, MITRE ATT&CK 전술 및 기법과 매핑하는 데 필수적인 정보를 제공한다. 본 연구에서는 AWS 환경에서 발생할 수 있는 권한 상승(Privilege Escalation) 시나리오에 대해 효과적으로 탐지하기 위한 필드를 설계하였다.

Offensive Tool에서 제공하는 공격들은 기본적으로 현재 공격자가 가지고 있는 권한을 이용해 다른 탈취 가능한 권한을 열거하고, 탈취하여 악성 행위를 하는 공격의 비율이 약 60% 이상으로 다루어졌다. 이는 클라우드 환경에서 IAM 권한 관련 위협이 주요 보안 문제임을 나타낸다. IAM은 사용자, 역할, 권한 등을 관리하여 클라우드 자원에 대한 접근을 통제한다. 잘못 구성된 IAM 정책, 과도한 권한, 자격 증명의 탈취는 클라우드 보안 사고의 주요한 원인으로 지목된다.

Table 1. CloudTrail field used in learning

Field Name	Purpose of use
userIdentity userIdentity.type sourceIpAddress	It is used to distinguish between internal users and attackers by recording user information.
accessKeyId	It is used to detect abnormal key usage by identifying AWS user credentials.
eventName	The API call information performed by the user is recorded to analyze behavior patterns and detect abnormal behavior.
eventTime	It is used to identify attack timing and activity patterns through the time of event occurrence.
resource	It records the resources accessed by the user to detect abnormal access attempts to the resources.
requestParameter	It analyzes the parameters delivered during API calls to detect rising authority or data leakage.
responseElements	Log API call results to detect successful request and suspicious behavior
errorCode	It analyzes repetitive and suspicious error code patterns to detect under-authorization or abnormal access attempts.
eventType	It detects attack scenarios by classifying event types such as management events or data access events.
managementEvent	Early detection of privilege escalation by logging events related to changes in IAM settings.

3.2 학습 데이터 구축

보안 로그 데이터의 확보는 본 연구의 핵심 단계로, Opensource Offensive Tool을 활용하여 현실적인 공격 시나리오를 포함한 데이터를 생성하였다. 학습 데이터는 3개의 Stratus Red Team, Pacu, CloudGoat 오픈소스 보안 도구를 활용해 AWS 환경에서 발생할 수 있는 공격을 시뮬레이션함으로써 확보하였다. Stratus Red Team을 통해 클라우드 환경에서 MITRE ATT&CK 전술과 기법 기반의 공격을 시뮬레이션함으로써 로그 데이터를 생성할 수 있었다. 먼저 Stratus Red Team[9]은 MITRE ATT&CK 프레임워크에 기반하여 클라우드 환경에서의 전술과 기법을 재현하는 도구로 사용되었다. 이 도구는 클라우드 자원에 대한 비인가 접근, 권한 상승, 서비스 거부(DoS)와 같은 다양한 공격을 로그로 생성하여 데이터 수집에 기여하였다. 또한, Pacu[10]와 CloudGoat[11]는 AWS 환경을 대상으로 한 권한 상승, 데이터 유출 등 현실적인 공격 시나리오를 재현하여 로그 데이터를 확보하였다. 이 과정에서 생성된 로그 데이터는 각 공격 활동의 상세한 이벤트 기록을 포함하고 있다. TrailDiscover[12]는 AWS CloudTrail 로그와 MITRE ATT&CK 프레임워크 간의 매핑을 가능하게 하는 중요한 데이터셋으로, 이를 활용해 다양한 연구에서 로그 데이터에 대한 라벨링 및 분석 작업이 이루어지고 있다. HLogformer[13]는 TrailDiscover를 기반으로 AWS 로그의 eventName를 MITRE ATT&CK 프레임워크에 대해 매핑을 수행하였다. 본 연구에서도 TrailDiscover 데이터셋을 활용하여 AWS CloudTrail 로그와 MITRE ATT&CK 전술 및 기법을 매핑하였으며, 이를 통해 로그 데이터의 라벨링을 진행하였다. IAM은 클라우드 보안의 핵심 요소이기 때문에, 현재 확보된 공격 데이터는 IAM 관련 공격 시나리오를 중심으로 이루어졌다. 현재 포함된 공격들에서는 비인가된 IAM 권한 사용, 권한 상승, IAM 정책 조작 등의 기법을 포함하고 있다. 향후 연구에서는 IAM 관련 공격 외에도 네트워크, 데이터베이스 등 다양한 서비스에 대한 공격 시나리오를 학습 데이터로 추가할 수 있다. Table 2는 본 연구에 사용된 Cloud 상의 MITRE ATT&CK 전술 및 기법과 해당하는 데이터를 보여준다.

Table 2. MITRE ATT&CK Tactics and Techniques Used in Learning

Tactic	Technique	Support
TA0003-Persistence	T1136-Create Account	20
	T1098-Account Manipulation	398
TA0004-Privilege Escalation	T1484-Domain Policy Modification	10
	T1556-Modify Authentication Process	20
TA0005-Defense Evasion	T1070 - Indicator Removal on Host	471
	T1552-Unsecured Credentials	1813
TA0006-Credential Access	T1555-Credentials from Password Stores	20
	T1111-Multi-Factor Authentication Interception	10
TA0007-Discovery	T1087 - Account Discovery	2277
	T1016-System Network Configuration Discovery	141
	T1069-Permission Groups Discovery	116
TA0008-Execution	T1059-Command and Scripting Interpreter	54
TA0009-Collection	T1530-Data from Cloud Storage	42
	T1560-Archive Collected Data	44
TA0010 - Exfiltration	T1020 - Automated Exfiltration	306
	T1048 - Exfiltration Over Alternative Protocol	10
	T1537 - Transfer Data to Cloud Account	10
TA0040 - Impact	T1489 - Service Stop	43
No Attack	No Attack	10,603
Total		16,398

IV. 제안 기법

본 연구에서는 슬라이딩 윈도우 기반 BERT 모델을 활용하여 AWS CloudTrail 로그 데이터를 분석하고 MITRE ATT&CK 프레임워크의 전술 및 기법을 예측하는 접근 방식을 제안한다. 슬라이딩 윈도우 기법은 로그 데이터를 고정된 크기로 나누어 문맥 정보를 학 수 있도록 하며, 병렬 처리 및 BERT 모델을 결합하여 대규모 로그 데이터를 효과적으로 처리하고 학습할 수 있도록 설계되었다.

4.1 프레임워크 구조

전처리 된 AWS CloudTrail 로그는 JSON 형식으로 저장된 이벤트 기록으로, 각 이벤트는 로그의 주요 속성(예: eventTime, eventName, sourceIPAddress)을 포함한다. 전체 로그에 대해 슬라이딩 윈도우 기법을 적용하여 고정 크기 W=5의 로그 시퀀스를 생성한다. 이를 통해 로그 데이터 내 문맥적 및 시계열적 정보를 유지하며 학습 효율성을 높인다.

4.2 슬라이딩 윈도우 기법

로그 데이터를 효과적으로 분석하기 위해 Fig 1과 같이 슬라이딩 윈도우 기법을 활용하였다. 슬라이딩 윈도우는 긴 로그 시퀀스를 고정된 크기의 작은 단위로 분할하여 처리할 수 있도록 하는 방법이다. 이 기법은 로그 데이터의 시계열적 의존성을 유지하며, 윈도우 내 문맥적 관계를 학습할 수 있는 환경을 제공한다. 본 연구에서는 Window size는 5로 설정하였으며, stride는 1로 설정하여 윈도우 간 데이터가 겹치도록 설계하였다. 이러한 설계는 각 윈도우가 독립적이면서도 시계열적 연속성을 보장하도록 돕는다. 로그 시퀀스가 주어질 때, window size W와 stride size s를 설정하여, 윈도우를 생성한다. 이는와 Fig 1과 같은 형태로 로그 데이터를 분할한다. 이를 통해 전체 로그 시퀀스를 학습하기 위한 연산 부담은 줄이고, 국소적 문맥 정보를 효과적으로 학습할 수 있다. 또한, 윈도우 간의 겹침을 통해 시퀀스 내 정보 손실을 최소화하고 데이터의 연속성을 보장한다.

4.3 표현 (Input representation)

슬라이딩 윈도우로 생성된 로그 시퀀스는 BERT 모델의 입력으로 사용된다. 각 로그 이벤트는 텍스트 데이터로 표현되며, 이를 BERT 모델이 이해할 수 있도록 벡터화 과정을 거친다. 윈도우 내에서, 각 로그들은 주요 속성(eventName, sourceIPAddress, resources 등)을 포함한다.

4.4 BERT 모델

BERT 모델은 Transformer 아키텍처를 기반으로 하며, Self-Attention 메커니즘을 사용하여 입력 데이터의 문맥적 정보를 학습한다. 본 연구에서 BERT 모델은 Token Classification을 활용하여 Window 안의 각각의 로그 데이터를 토큰 단위로 하여 분석하고, 각 로그에 대응하는 MITRE ATT&CK 전술 및 기법을 예측한다. 입력 데이터는 여러 Transformer 레이어를 통과하며, 각 레이어

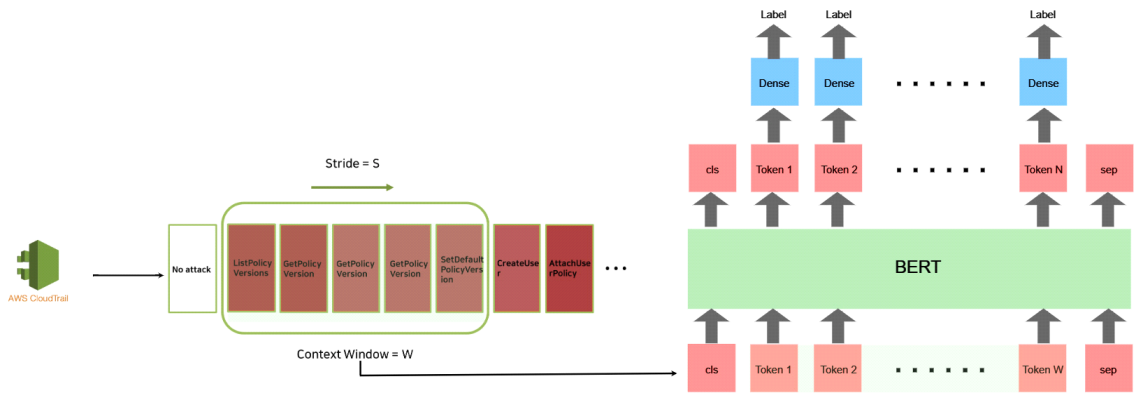

Fig. 1. BERT model structure based Sliding Window

는 다중 헤드 어텐션(Multi-Head Attention)과 위치별 피드포워드(Position-wise Feed Forward) 레이어로 구성된다. BERT 모델은 입력 벡터로 각 윈도우 시퀀스를 사용하여 로그 데이터 간의 문맥적 관계를 포괄적으로 학습한다. BERT 출력은 MITRE ATT&CK 전술 및 기법에 대한 분류 결과를 생성한다.

V. 성능 검증

본 연구에서는 슬라이딩 윈도우 기반 BERT 모델을 활용하여 AWS CloudTrail 로그 데이터를 분석하고 MITRE ATT&CK 전술 및 기법을 예측하였다. 이를 검증하기 위해 모델의 성능을 정밀도(Precision), 재현율(Recall), F1 점수(F1 Score)의 세 가지 척도를 기준으로 평가하였다. 본 연구에서 사용한 성능 지표에서는 TP(True Positive)의 경우 실제로 악성 로그를 이상으로 탐지하여 MITRE ATT&CK 지표에 맞게 매핑한 경우를 의미한다.

5.1 실험 설정

실험에서 사용된 데이터는 AWS CloudTrail 로그를 기반으로 슬라이딩 윈도우 기법을 적용하여 전처리되었다. 슬라이딩 윈도우는 윈도우 크기 W=5와 stride S=1로 설정하여 시계열적 맥락 정보를 유지하며 데이터를 학습할 수 있도록 하였다. 데이터셋은 전체 로그 데이터를 Stratified Sampling을 통해 Train:Validation:Test = 7:1.5:1.5 비율로 분리하였다. 이를 통해 각 데이터셋에서 클래스 불균형 문제를 최소화하며, 테스트셋에서도 모든 클래스에 대해 고르게 평가될 수 있도록 설계되었다. 모델 학습 과정에서 사용된 주요 설정은 다음과 같다. 모델은 Pre-trained bert-base-uncased를 기반으로 한 BertForTokenClassification 모델을 사용하였다. 손실함수는 클래스 불균형 문제를 해결하기 위해 가중치를 적용한 nn.CrossEntropyLoss를 사용하였다.

클래스별 가중치는 데이터 분포를 기반으로 산정되었으며, 이를 통해 특정 클래스에 대한 과소평가 문제를 완화하였다. 최적화 알고리즘은 AdamW을 사용하였으며, 하이퍼 파라미터는 learning_rate(학습률)은 5e-5, batch_size는 1, 학습 epoch 수는 6을 적용했다. 또한 과적합을 막기 위하여 Early Stopping을 사용하였다. Validation 손실이 더 이상 개선되지 않을 경우 학습을 중단하기 위해 patience=3, min_delta=0.01로 설정하였다. 다중 GPU환경에서 PyTorch DistributedDataParallel 기법을 활용하여 학습 시간을 단축하고 메모리 활용도를 최적화하였다.

5.2 실험 결과

본 연구의 제안 모델은 Weighted Average Precision 0.941, Recall 0.932, F1 Score 0.933으로 우수한 성능을 보였다. 특히, 대부분의 MITRE ATT&CK 기술들에 대해 높은 Recall을 유지하며, 예측 결과의 신뢰도를 입증하였다. Table 3은 주요성능 지표인 Precision, Recall, F1 Score와 Support를 나타낸다.

기존 연구들은 AWS CloudTrail과 같은 클라우드 로그가 아닌 HDFS와 같은 시스템 내부 로그를 활용하여 이상 탐지를 수행하였다. 이러한 연구들은 MITRE ATT&CK을 기준으로 악성 탐지를 라벨링하지 않았으며, 시스템 내부 로그에 비해 CloudTrail 로그는 필드의 수가 매우 많고 구조적으로 복잡하기 때문에 직접적인 성능 비교가 어려운 한계가 있다. 이에 본 연구에서는 두 가지 baseline을 설정하였다. 첫째, Feature extraction 없이 CloudTrail의 모든 로그 필드를 입력으로 사용하고 슬라이딩 윈도우를 적용하지 않은 상태에서 BERT를 활용한 경우, 둘째, Feature extraction을 포함하되 기본 BERT를 활용한 경우를 비교 대상으로 설정하였다.

실험 결과, Feature extraction 없이 CloudTrail의 모든 로그 필드를 입력으로 사용한 baseline 모델은 Table 4와 같이 낮은 성능을 보였다. 이는 로그 필드의 수가 과도하게 많아지면서 불필요한 정보가 포함되고, 중요한 정보에 대해 적절히 Attention하지 못한 결과로 해석된다. 반면, Feature extraction을 적용하여 중요한 필드만을

Table 3. MITRE ATT&CK Technique Key Performance Indicators

Average Test Loss	Accuracy	Precision	Recall	F1 Score(weighted)
0.122	0.932	0.941	0.932	0.933

Table 4. Performace Comparison of Models Without Feature Extraction and Sliding Window

Average Test Loss	Accuracy	Precision	Recall	F1 Score(weighted)
1.595	0.600	0.650	0.601	0.624

Table 5. Performace Comparison of Models With Feature Extraction but Without Sliding Window

Average Test Loss	Accuracy	Precision	Recall	F1 Score(weighted)
0.231	0.833	0.850	0.832	0.841

선별한 baseline 모델은 Table 5와 같이 성능이 개선되었으나, 시계열적 맥락 정보를 반영하지 못해 로그 간의 상관관계 학습이 제한된 것으로 판단된다.

Table 6는 MITRE ATT&CK Technique별 성능 지표를 나타낸다. 대체로 Precision, Recall, F1-Score에서 좋은 성능을 보이지만, T1555, T106, T1530, T1020에서는 상대적으로 낮은 성능을 보인다. 실험에서는 예측이 틀린 경우, 해당 윈도우를 뽑아내어 분석하였다. T1555(Credentials from Password Stores)의 경우 비밀번호 저장소에서 자격 증명을 탈취하는 기술로, 유사한 이벤트가 다른 Credential Access 관련 기술들과 혼동되는 경향을 보였다. 이 기술과 다른 Credential 관련 기술들이 eventType 또는 eventName 필드에서 유사한 패턴을 가지며, 학습 데이터 수도 적었기 때문에 T1555에 대한 클래스의 로그 특징을 효과적으로 학습하지 못했다고 보여졌다. T1016 (System Network Configuration Discovery)의 경우 Precision(0.59)이 비교적 낮은 이유는 네트워크

Table 6. Test Performance Indicators by MITRE ATT&CK Technique

Tatic	Technique	Precision	Recall	F1 Score	Support
TA0003-Persistence	T1136-Create Account	1.00	1.00	1.00	12
	T1098-Account Manipulation	0.88	1.00	0.93	265
TA0004-Privilege Escalation	T1484-Domain Policy Modification	1.00	1.00	1.00	8
	T1556-Modify Authentication Process	1.00	1.00	1.00	17
TA0005-Defense Evasion	T1070-Indicator Removal on Host	0.97	0.93	0.95	332
	T1552-Unsecured Credentials	0.97	1.00	0.98	1381
TA006-Credential Access	T1555-Credentials from Password Stores	0.27	1.00	0.43	3
	T1111-Multi-Factor Authentication Interception	0.78	1.00	0.87	6
TA0007-Discovery	T1087-Account Discovery	0.97	0.87	0.86	1616
	T1016-System Network Configuration Discovery	0.59	0.78	0.67	109
	T1069-Permission Groups Discovery	0.98	1.00	0.99	87
TA0008-Execution	T1059-Command and Scripting Interpreter	0.87	1.00	0.93	46
TA0009-Collection	T1530-Data from Cloud Storage	0.67	1.00	0.80	3
	T1560-Archive Collected Data	0.81	1.00	0.89	34
TA0010-Exfilration	T1020-Automated Exfiltration	0.61	0.97	0.75	255
	T1048-Exfiltration Over Alternative Protocol	1.00	1.00	1.00	5
	T1537-Transfer Data to Cloud Account	1.00	0.50	0.67	2
TA0040-Impact	T1489-Service Stop	1.00	1.00	1.00	26
No Attack	No Attack	0.95	0.94	0.94	7550

구성 관련 이벤트가 유사한 다른 기법인 T1087(Account Discovery)로 오분류되었기 때문이다. 특히, eventName 필드의 유사성이 주요 원인으로 작용한다. T1530 (Data from Cloud Storage)의 Precision(0.67)이 낮은 것은 해당 기법이 데이터에서 드물게 나타나며, 이는 데이터 수집 및 보관 관련 활동으로 다른 데이터 수집 기법인 T1560(Archive Collected Data)과 일부 유사한 로그 구조를 가져 혼동되었다. T1020 (Automated Exfiltration)는 Precision(0.61)과 Recall(0.97) 간의 큰 차이를 보였다. 이 기술이 자주 탐지되지만, 유사한 다른 Exfiltration 기술 T1048(Exfiltration Over Alternative Protocol)과의 구분이 어려운 경우가 존재했기 때문이다. 따라서, Attack 클래스의 경우 No Attack으로 예측된 경우가 거의 없었다는 점에서 모델이 공격과 비공격을 구분하는 데에는 우수한 성능을 보였음을 확인할 수 있다. 이는 공격 이벤트 탐지의 신뢰성을 높이는 긍정적인 결과로 해석될 수 있다. 그러나, 개별 Attack 클래스들 간의 세부적인 구분에서는 성능 저하가 관찰되었다. 이는 클래스별 데이터의 불균형이 주요 원인으로 작용했을 가능성이 높다. 따라서, 클래스별 데이터를 늘리고 소수 클래스에 대한 데이터 증강(Data Augmentation) 기법을 활용한다면 이러한 성능 저하를 보완할 수 있을 것이다. 또한, 추가적인 데이터 확보를 통해 특정 공격 기술의 특징을 더 잘 학습할 수 있도록 데이터셋을 확장하는 것도 하나의 해결책이 될 수 있다.

현재 실험에서는 stride=1, w=5 슬라이딩 윈도우 방식을 사용하여 데이터를 처리하였기 때문에, 하나의 로그가 5개의 윈도우에 중복되어 나타났다. 이로 인해 데이터의 support 양이 증가하였다. 실제 환경에서는 실제 환경에서는 슬라이딩 윈도우 방식으로 수집된 데이터에서 5개의 윈도우 중 3개 이상이 공격으로 탐지되었을 경우 이를 공격으로 판단하는 Hard Voting 방식을 이용할 수 있다.

Table 7. Learning Results by Window Size

Window Size	Precision	Recall	F1 Score
1	0.850	0.832	0.841
3	0.910	0.921	0.915
5	0.941	0.932	0.933
7	0.900	0.910	0.905

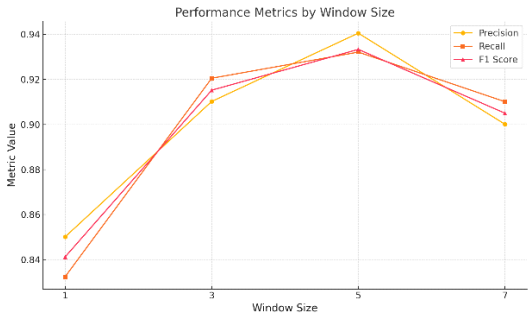

Fig. 2. Learning Results by Window Size

슬라이딩 윈도우 크기 W에 따른 모델의 성능을 분석하기 위해 W = 1,3,5,7의 크기를 각각 실험하였다. 윈도우 크기가 증가함에 따라 로그 데이터 내 문맥 정보가 더 잘 반영되었으며, 적절한 크기에서 모델의 성능이 최적화되었다. 윈도우 크기가 7에서 성능이 낮아진 모습을 볼 수 있다. 이는 학습 데이터의 공격 로그 시퀀스 길이에 의하여 윈도우 크기가 5일 때 보다 낮은 모습을 보인다. 또한, 슬라이 anomaly와 연관되지 않은 이벤트까지 학습 대상이 되어 오탐률이 증가할 수 있다.

VI. 결 론

본 연구에서는 AWS CloudTrail 로그 데이터를 분석하여 MITRE ATT&CK 프레임워크 기반 위협 전술 및 기법을 효과적으로 예측하기 위해 슬라이딩 윈도우 기반 BERT 모델을 제안하였다. 슬라이딩 윈도우 기반 접근법으로 긴 로그 시퀀스를 작은 윈도우로 분할하여 처리함으로써 문맥 정보를 세밀하게 학습하고 이벤트 간의 연관성을 효과적으로 반영하도록 설계되었다. 본 연구 결과, Weighted Average Precision 0.941, Recall 0.932, F1 Score 0.933을 기록하며, 주요 MITRE ATT&CK 기술에 대해 높은 탐지 성능을 입증하였다. 결론적으로, 본 연구는 로그 데이터 분석에서 문맥 정보를 활용하는 것이 얼마나 중요한지를 보여주었으며, 슬라이딩 윈도우 기반 BERT 모델은 효율적이고 신뢰할 수 있는 위협 탐지 시스템으로 자리 잡을 가능성을 입증하였다. 특히, 본 연구에서 제안한 모델 구조는 단순히 클라우드 환경에 국한되지 않고, 네트워크 보안, 엔드포인트 위협 탐지, IoT 환경 등 다양한 도메인에서의 위협 탐지에도 확장 가능성이 높다. 그러나 본 연구에서는 모든 MITRE ATT&CK 전술 및 기

법에 대한 케이스의 확보가 이루어지지 않아, 모든 전술과 기법을 포괄하지 못했고, 제한된 라벨 값으로만 실험을 진행한 한계가 있다. 향후 연구에서는 데이터셋을 확장하여 더 다양한 전술과 기법을 다룰 수 있도록 하고, 데이터 불균형 문제를 해결하며, 실시간 위협 탐지 및 대응 능력을 강화하는 방향으로 발전할 수 있을 것이다.

References

〔1〕 He, Shilin, et al, "Experience report: System log analysis for anomaly detection.", 2016 IEEE 27th international symposium on software reliability engineering (ISSRE), (p. 207-218), 2016.

〔2〕 Xu, Wei, et al, "Detecting large-scale system problems by mining console logs.", Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles, (p.117-132), 2009.

〔3〕 Smola, Alex J., and Bernhard Scholkopf, "A tutorial on support vector regression.", Statistics and computing 14, (p.199-222), 2004

〔4〕 Du, Min, et al, "DeepLog: Anomaly detection and diagnosis from system logs through deep learning.", Proceedings of the 2017 ACM SIGSAC conference on computer and communications security, p.1285-1298, 2017.

〔5〕 Meng, W., Liu, Y., Zhu, Y., Zhang, S., Pei, D., Liu, Y., Chen, Y., Zhang, R., Tao, S., Sun, P., & Zhou, R. (2019). "LogAnomaly: Unsupervised Detection of Sequential and Quantitative Anomalies in Unstructured Logs." Proceedings of IJCAI, (p.4739-4745), 2019.

〔6〕 Chen, Yiyong, Nurbol Luktarhan, and Dan Lv, "LogLS: research on system log anomaly detection method based on dual LSTM", Symmetry 14.3, p.454, 2022

〔7〕 Kenton, Jacob Devlin Ming-Wei Chang, and Lee Kristina Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding.", Proceedings of naacL-HLT. Vol. 1, 2019.

〔8〕 Guo, Haixuan, Shuhan Yuan, and Xintao Wu. "Logbert: Log anomaly detection via bert." 2021 international joint conference on neural networks (IJCNN). (p.1-8), 2021.

〔9〕 DataDog, "stratus-red-team," GitHub, 2024. [Online]. Available: https://github.com/DataDog/stratus-red-team. [Accessed: Nov. 28, 2024].

〔10〕 RhinoSecurityLabs, "pacu," GitHub, 2024. [Online]. Available: https://github.com/RhinoSecurityLabs/pacu. [Accessed: Nov. 28, 2024].

〔11〕 RhinoSecurityLabs, "cloudgoat," GitHub, 2024. [Online]. Available: https://github.com/RhinoSecurityLabs/cloudgoat. [Accessed: Nov. 28, 2024].

〔12〕 A. Alvarez, "TrailDiscover," GitHub, 2024. [Online]. Available: https://github.com/adanalvarez/TrailDiscover. [Accessed: Nov. 28, 2024].

〔13〕 Hou, Zhichao, et al. "HLogformer: A Hierarchical Transformer for Representing Log Data." arXiv preprint arXiv:2408. 16803. 2024.

〈저자소개〉

박 현 준 (Hyun-jun Park) 학생회원
2022년 3월~현재: 아주대학교 국방디지털융합학과 재학
〈관심분야〉 AI, NLP, Cloud Security

차 원 제 (Won-je Cha) 학생회원
2022년 3월~현재: 창원대학교 정보통신공학과 재학
〈관심분야〉 AI, Cloud Security, 정보보안

최 유 정 (Yu-jeong Choi) 학생회원
2021년 3월~현재: 덕성여자대학교 사이버보안전공 재학
〈관심분야〉 Cloud Security, System Security, 정보보안

김 태 양 (Tae-yang Kim) 학생회원
2019년 3월~현재: 중앙대학교 산업보안학과 재학
〈관심분야〉 AI, 개인정보, 소프트웨어 공학, 정보보안

김 지 윤 (Ji-yun Kim) 학생회원
2020년 3월~현재: 덕성여자대학교 사이버보안전공 재학
〈관심분야〉 AI, 소프트웨어 공학, 정보보안

신 예 지 (Ye-ji Shin) 학생회원
2021년 3월~현재: 수원대학교 정보보호학과 재학
〈관심분야〉 AI, 소프트웨어 공학, 정보보안