Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.33, NO.3, Jun. 2023 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2023.33.3.411

Reference 기반 AI 모델의 효과적인 해석에 관한 연구*

이 현 우,[1†] 한 태 현,[2] 박 영 지,[2] 이 태 진[3‡]
[1,2,3]호서대학교 (대학원생, 학생, 교수)

A Study on Effective Interpretation of AI Model based on Reference*

Hyun-woo Lee,[1†] Tae-hyun Han,[2] Yeong-ji Park,[2] Tae-jin Lee[3‡]
[1,2,3]Hoseo University (Graduate student, Student, Professor)

요 약

오늘날 AI(Artificial Intelligence) 기술은 다양한 분야에서 활용 목적에 맞게 분류, 회기 작업을 수행하며 광범위하게 활용되고 있으며, 연구 또한 활발하게 진행 중인 분야이다. 특히 보안 분야에서는 예기치 않는 위협을 탐지해야 하며, 모델 훈련과정에 알려진 위협 정보를 추가하지 않아도 위협을 탐지할 수 있는 비 지도학습 기반의 이상 탐지 기법이 유망한 방법이다. 하지만 AI 판단에 대한 해석 가능성을 제공하는 선행 연구 대부분은 지도학습을 대상으로 설계되었기에 학습 방법이 근본적으로 다른 비 지도학습 모델에 적용하기는 어려우며, Vision 중심의 AI 매커니즘 해석연구들은 이미지로 표현되지 않는 보안 분야에 적용하기에 적합하지 않다. 따라서 본 논문에서는 침해 공격의 원본인 최적화 Reference를 탐색하고 이와 비교함으로써 탐지된 이상에 대한 해석 가능성을 제공하는 기법을 활용한다. 본 논문에서는 산출된 Reference를 기반으로 실존 데이터에서 가장 가까운 데이터를 탐색하는 로직을 추가 제안함으로써 실존 데이터를 기반으로 이상 징후에 대한 더욱 직관적인 해석을 제공하고 보안 분야에서의 효과적인 이상 탐지모델 활용을 도모하고자 한다.

ABSTRACT

Today, AI (Artificial Intelligence) technology is widely used in various fields, performing classification and regression tasks according to the purpose of use, and research is also actively progressing. Especially in the field of security, unexpected threats need to be detected, and unsupervised learning-based anomaly detection techniques that can detect threats without adding known threat information to the model training process are promising methods. However, most of the preceding studies that provide interpretability for AI judgments are designed for supervised learning, so it is difficult to apply them to unsupervised learning models with fundamentally different learning methods. In addition, previously researched vision-centered AI mechanism interpretation studies are not suitable for application to the security field that is not expressed in images. Therefore, In this paper, we use a technique that provides interpretability for detected anomalies by searching for and comparing optimization references, which are the source of intrusion attacks. In this paper, based on reference, we propose additional logic to search for data closest to real data. Based on real data, it aims to provide a more intuitive interpretation of anomalies and to promote effective use of an anomaly detection model in the security field.

Keywords: Artificial Inteligence, Interpretation, Autoencoder, Reference

Received(02. 15. 2023), Modified(1st: 04. 05. 2023, 2nd: 05. 02. 2023), Accepted(05. 02. 2023)

* 이 논문은 2022년도 호서대학교의 재원으로 학술연구비 지원을 받아 수행된 연구임(No.2022-0222-0001, 분석가를 위한 AI 예측결과 해석 및 오류 정정 지원 기술 연구)

† 주저자, 20171247@vision.hoseo.edu

‡ 교신저자, kinjecs0@gmail.com(Corresponding author)

I. 서 론

오늘날 AI(Artificial Intelligence) 기술은 매일같이 방대한 데이터가 발생하는 빅데이터 환경에서 이를 효과적으로 활용하고 처리하기 위해 등장한 기술이다. CAM[1] 등에 의하면 이러한 AI 기술은 활발한 연구에 힘입어 성능이 매우 좋아졌으며 이러한 성능을 기반으로 자연어 처리, 이미지 인식, 라벨 분류, 회기 예측 등 다양한 업무를 지원할 수 있게 되어 다양한 분야 및 기업 등에서 많이 쓰이고 있다고 말한다. 따라서 Shin[2] 이러한 AI를 보안 분야에 도입함으로써 얻게 되는 높은 업무효율 향상을 언급하였다. 하지만 Adadi[3] 와 Das[4] 등은 AI 성능이 좋아짐에 따라 해당 성능을 산출하기 위한 AI 동작 로직은 매우 복잡해지게 되었고, 결국 AI는 좋은 성능을 도출하지만, 산출결과에 대한 원인을 식별하기 어려운 Black Box 성격을 지니게 되었다고 말한다. 이렇듯 고도화된 연산을 수행하여 다양한 업무를 지원하기 위해 DNN을 활용하는 DL 모델은 아무리 좋은 성능을 산출하더라도 의사 결정의 투명성과 해석성이 떨어지게 되고 보안, 의료 분야와 같은 오 탐지 리스크가 큰 분야에 DL 기반 이상 탐지 모델을 채택하기에는 기존 AI 기술의 원인을 식별할 수 없는 결과는 결국 신뢰성 하락으로 이어지게 되어 AI 모델만으로 업무를 대체할 수 없는 치명적인 단점으로 적용되게 된다.

이러한 문제를 해결하기 위해 최근 XAI (eXplainable Artificial Intelligence)를 통해 AI 판단결과에 대한 Interpretation을 제공해 주는 기술이 등장 하였으며, Lee[5]등도 효과적인 AI 활용을 위해 XAI의 중요성이 높아진다고 언급하였다. 선행 연구된 대표적인 XAI 기법으로는 Lundber[6]가 소개한 모델의 판단에 관여한 Feature들의 영향력을 산출하는 SHAP, Zhou[7]가 소개한 이미지 분류모델에 이미지 Importance Heatmap을 추가하여 모델이 판단에 주된 관여를 한 영역을 강조하는 CAM 등이 있다. 이렇듯 기존 XAI 기법은 Label 정보와 산출된 Feature Importance 정보를 기반으로 모델 메커니즘을 해석하여 AI 예측에 대한 Interpretation을 제공하는 연구가 주로 개발되었다. 하지만 선행 연구된 Vision 중심의 AI 매커니즘에 대한 해석연구들은 이미지로 표현되지 않는 보안 분야에 적용하기에 적합하지 않은 방법이다. 또한, Liat[8]과 Han[9] 등에 의하면 기존 XAI 기법들은 대부분 Label 정보가 존재하는 지도학습 기반 분류모델의 해석에 초점을 맞추어 개발되었다고 말하며, Label이 없는 비 지도학습은 지도학습과 학습을 위한 매커니즘이 근본적으로 다르기에 지금까지 진행된 선행 XAI 기법들을 비 지도학습 모델에 직접 적용하기는 부적절하다고 말한다. 하지만 매일 같이 생성되는 수많은 데이터에 label을 부여하기는 어려우며, 특별한 Label 정보가 없이도 예기치 못한 위협을 탐지할 수 있는 비 지도학습 모델 기반의 이상 탐지는 보안 분야에서 매우 중요한 역할을 하는 작업이다.

따라서 보안 분야에서 효과적인 비 지도학습 모델 활용을 위한 XAI 기법의 필요성이 높아지고 있다. 본 논문에서는 Han[9] 등에 의해 소개된 Reference 기반 비 지도학습 모델에 대한 직관적인 해석 가능성을 제공하는 방법을 개량하여 더욱 효과적으로 비 지도학습 기반 모델에 대한 해석 가능성을 제공한다. 개선사항으로는 기존 로직을 통해 산출된 Reference를 더욱 효과적으로 제공하기 위해 Reference와 가장 가까운 실존 데이터를 찾아 Reference를 대체하는 것이다. 앞선 로직을 통해 기존 추정치였던 Reference 수치를 실존 데이터로 대체하여 비교함으로 해석 가능성 향상을 추구한다. 이후 산출된 해당 데이터를 통해 분석가와 전문가에게 Anomaly 판단 데이터에 대한 Interpretation을 더욱 가시적으로 제공하고, 로직 내부적으로 존재하는 Importance 수치를 기반으로 상위 Feature를 판별하여 Anomaly 산출에 주요한 영향을 준 Feature를 제공함으로써 보안 담당자에게 더욱 직관적인 이상 탐지에 대한 해석 가능성을 제공한다.

본 논문의 구성은 다음과 같다. 2장에서는 보안 업무에서의 AI 활용영역 및 기존 XAI의 적합성과 효과적인 Interpretation을 제공을 위한 Reference 기반 XAI 기법을 다룬다. 이후 3장에서는 본 논문에서 제안하는 모델의 Framework 설명과 더불어 Reference 산출을 위한 세부 로직을 다루고 해석의 유용성을 다루며, 4장에서는 결과 산출을 위한 데이터 세트와 Featuring 방안 설명을 수행하며, 해당 데이터 세트를 통해 실제 Reference 산출 후 결과 유용성을 검증하며, 5장에서는 결론을 맺는다.

II. 관련 연구

2.1 Current situation of Unsupervised Model for Security task

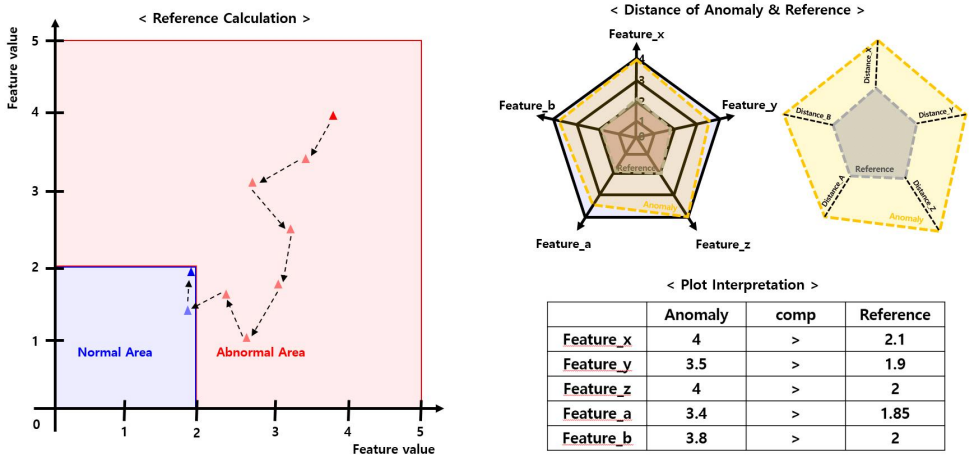
AI는 모델학습과정에서 학습데이터의 Label 정보의 영향 유무에 따라 지도학습과 비 지도학습으로 나뉘게 되며 이에 따라 각 모델이 수행하는 작업도 나뉘게 된다. 이 중 Label 정보가 없는 비 지도학습 기반 모델은 회기 작업을 수행하여 예측한 복원 값과 원본 값의 오차를 산출할 수 있다. 보안 분야에서의 이러한 비 지도학습 모델의 특징을 통해 Anomaly Detection을 수행하고 비 지도학습 모델은 네트워크 침입 탐지, 시스템 로그 이상 탐지, 웹 공격 탐지 등 다양한 환경에서의 보안 관련 이상 탐지 업무에 적용된다.

하지만 오 탐지 리스크가 큰 보안 분야에서 AI를 활용하기 위해서는 악성이나 공격 판단결과와 함께 산출결과에 대한 Interpretation을 함께 제공함으로써 산출결과에 대한 투명성을 제공해야 한다. SHAP, CAM 등 기존 선행연구는 Feature Importance를 기반으로 모델의 판단에 주요 영향을 미치는 정보를 파악함으로써 Vision을 중심으로 AI의 메커니즘을 이해시키는 Interpretation을 제공하고자 하였으나 이와 같은 Interpretation 정보는 이미지로 표현되지 않으며, 실제 수치를 기반으로 정상에서는 나타나지 않는 정보를 기반으로 악성 판단결과에 대한 Interpretation을 식별해야 하는 보안 분야에 적용하기에는 명확한 해석을 제공하지 못한다. 따라서 선행 연구된 XAI 기법들을 보안 분야에 적용하기에는 적절하지 않다. 따라서 효과적으로 보안 분야에 XAI를 활용하기 위해서는 악성 판단에 대한 명확한 원인을 제공하는 Interpretation 기법이 필요한 상황이다.

2.2 Reference based XAI

Han[9] 등에 의해 제안된 개념으로 Reference는 기존 XAI 적용이 어려웠던 비지도학습에서의 Interpretation 제공을 위해 산출하는 값이다. 기존 비 지도학습은 데이터에 대한 라벨이 존재하지 않기에 지도학습 메커니즘에 적용하기 위해 선행 연구된 XAI 기법을 적용하는데 어려움이 존재한다. 이러한 특성 때문에 기존의 비 지도학습기법은 모델은 보안 분야에서 효과적으로 활용되지 못하였다. Reference는 이러한 문제를 해소하기 위해 라벨이 없는 비 지도학습 환경에서 anomaly 데이터에 대한 anomaly 산출 원인을 제공할 수 있는 적합한 Reference를 탐색하고 anoamly 데이터와 Reference의 차이를 대조를 통해 두 데이터의 차이를 직관적으로 제시함으로써 비 지도학습 모델의 판단결과에 대한 Interpretation을 제공한다. Han[9]의 로직을 통해 산출되는 Reference의 주요 의미는 anomaly로 산출된 데이터에 대해 해당 데이터와 가장 가까우면서도 학습된 모델이 판단했을 때 정상인 값을 의미한다. 즉, 산출된 Reference는 Anomaly와 Normal 판단의 경계인 Decision Boundary 주변에 형성되고 이는 즉, 정상 중에서도 Anomaly와 가장 가까운 값이 된다. Reference를 통한 Interpretation 제공 예시로는 Fig. 1과 같다. 4가지 Feature를 지니는 데이터에서 Normal의 범주가 0~2, Anormaly의 범주가 2~5인 경우라면, Reference는 Anomaly 데이터를 기반으로 해당 데이터와 가장 가까우면서도 Normal 범주에 속하는 값인 2 주변 값으로 산출된다. 따라서 Anomaly로 판단된 데이터는 Normal 범주의 최 외각인 Reference보다 높은 수치를 지니기에 Anomaly로 판단된 원인을 Reference와의 비교를 통해 Interpretation로써 제공할 수 있다. 이렇게 Reference를 산출하여 원본 anomaly Data와의 비교 대상으로 지정함으로써 라벨이 존재하지 않는 비 지도학습에서의 데이터 비교분석이 가능해지고 Reference와의 차이를 직관적으로 제공함으로써 비 지도학습에서의 효과적인 Interpretation을 제공한다. 또한, Reference와 원본 anomaly 데이터 사이의 거리가 멀다는 것은 anomaly data가 모델이 판단하기에 해당 데이터는 정상 범주에서 멀리 떨어져 있다는 것을 의미함으로 해당 데이터는 anomaly라는 직관적인 해석이 가능하다.

Montavon[10]에 의하면 최근 여러 연구에서는 최종 결정에 있는 가장 영향력 있는 Feature만으로 Interpretation을 제공함으로써 더욱 직관적인 해석 가능성을 제공할 수 있다고 한다. 선행 연구된 Han[9]의 내부 코드에는 내부적으로 복구 정도에 따른 Feature 별 Importance 수치를 산출할 수 있으며, 복구가 잘 안 되는 Feature를 anomaly 판단에 많이 기여한다고 판단하여 중요 Feature를 가시적으로 식별할 수 있다. 이를 통해 악성 판단에



< Plot Interpretation >

	Anomaly	comp	Reference
Feature_x	4	>	2.1
Feature_y	3.5	>	1.9
Feature_z	4	>	2
Feature_a	3.4	>	1.85
Feature_b	3.8	>	2

Fig. 1. Reference Based Interpretation

기여한 핵심적인 Feature만으로 더욱 직관적인 해석 지원이 가능하다. 이렇듯 Reference 탐색을 통해 기존 방식으로는 불가능했던 비 지도학습 모델에서의 비교 대상을 생성하고 이 둘 사이의 오차를 제공함으로써 보안 분야에 적합하고 명확한 Interpretation 제공 및 보안 분야에서의 효과적인 AI 활용을 도모하고자 한다.

III. 제안 모델

3.1 Proposed Framework

본 논문에서 제안하는 Framework는 Fig.2와 같다. 우선 EDR 시스템에서 추출된 원본 log 데이터를 Featuring하여 Autoencoder 모델 학습을 위한 데이터 세트로 데이터 전처리를 수행한다. 산출된 데이터로 Anomaly Detection을 수행하는 Autoencoder 모델을 학습시킨 후 Anomaly Detection을 수행함으로써 공격 Process를 탐지한다. 이후 탐지된 Anomaly 즉, 악성 Process에 대한 Anomaly 탐지 원인을 Interpretation 제공하기 위해 Han[9] 등이 소개한 기법으로 Reference를 산출한다. 이후 보다 효과적인 해석 가능성 제공을 위해 KNN(k=1) 알고리즘을 기반으로 Train data set에서 Reference와 가장 가까운 데이터 즉, Nearest Real Data를 탐색한다. 앞선 산출결

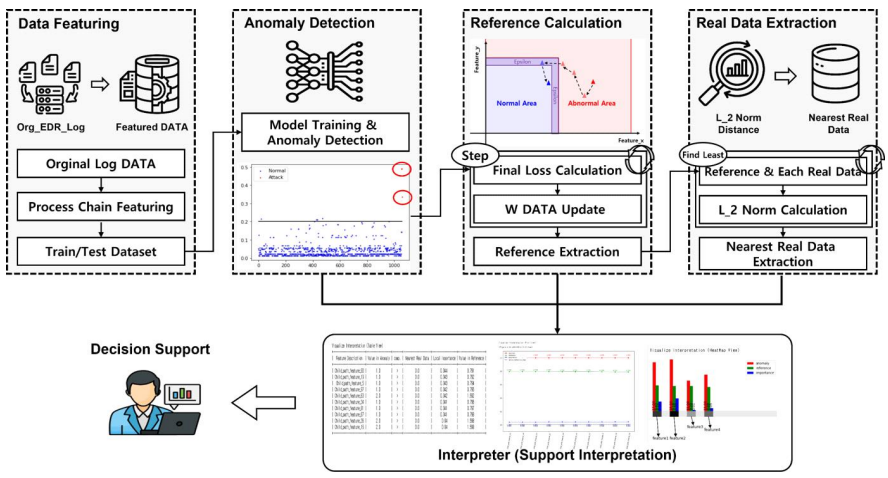


Fig. 2. Proposed Framework

과인 원본 Anomaly Data, Reference value, Feature Importance, Nearest Real Data 정보를 종합하여 제공함으로써 Anomaly 산출 원인에 대한 가시적인 해석 가능성을 제공한다.

3.2 Reference 산출 Algorithm

Reference는 Interpretation을 제공하기 위해 선택된 Anomaly Data에서 시작하여 2가지 loss 값을 설정하고 이들을 종합한 최종 loss 값을 줄여나가는 방향으로 Reference 값이 갱신되며, 갱신 중인 Reference 값을 데이터 w라고 칭한다. 즉, 데이터 w는 업데이트를 수행할수록 AI 모델이 정상으로 판단하면서도 Anomaly Data와 가장 가까운 방향으로 이동하게 되고 내부적으로 지정한 탐색 Step만큼 갱신이 이루어지거나 최종 loss가 줄어들지 않는 시점에서 Early Stop이 발동되어 갱신 중인 현재 데이터 w 값이 최종 Reference로써 산출된다.

Reference 산출을 위해 사용되는 첫 번째 loss인 loss1 수식은 수식(1)과 같다.

$$loss1 = Relu(MSE(modle(w), w) - (thres - eps)) \quad (1)$$

loss1 으로 인해 갱신되는 데이터 w의 업데이트 추이는 Fig.4와 같다. 수식(1)에서 대한 상세한 해석을 진행해보면, 우선 w는 선택한 Anomaly Data에서 시작하여 설계된 최종 loss 값을 줄여가는 방향으로 업데이트가 수행되는 대상이다. thres는 앞서 학습된 Autoencoder 모델로부터 산출된

Algorithm 1 : Calculate Reference value

Input : Trained Model, Anomaly Data, thres, eps, lbd
Output : Reference

```
w <- Anomaly Data;
Before_loss <- Inf;
for i = 1 to Step do
    loss1 = Relu(MSE(w, model(w) - (thres-eps));
    loss2 = L_2 Norm(w, Anomaly Data);
    Loss = loss1 + lbd * loss2;
    w <- SGD (w, Loss);
    if Before_Loss > Loss then Before_Loss <- Loss;
    else exit;
end
Reference <- w
returen Reference
```

Fig. 3. Algorithm of Calculation Reference Value

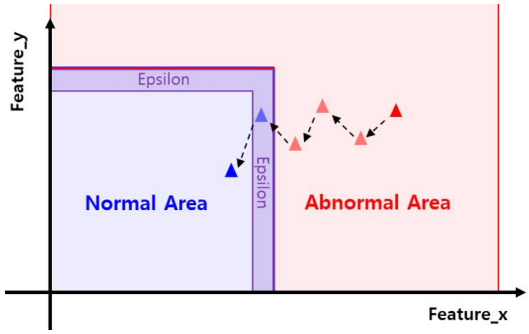


Fig. 4. loss1 based Data(w) update

Threshold Value이다. thres에 담기는 세부적인 수치는 학습 시 지정된 threshold rate에 해당하는 상위 mse 값으로 threshold rate = 0.01이면 mse가 높은 상위 1%를 Anomaly로 판단하겠다는 의미이다. 즉, model은 복원한 데이터와 원본 데이터 간의 mse가 thres 값보다 크면 Anomaly, 작으면 Normal 으로 판단함을 의미한다. 이러한 기존 Autoencoder의 Anomaly Detection 개념을 Relu와 접목하여 loss1은 갱신 중인 w가 Anomaly에서 정상 범주로 이동하게 하는 역할을 한다. 수식(1)을 해석하면, eps 라는 파라미터가 존재한다. eps(=0.1)는 작은 소수 값을 지니는 Reference 산출을 위한 하이퍼 파라미터로 thres에서 해당 수치를 빼줌으로써 Normal과 Anomaly 사이의 Decision Boundary에 약간의 간격을 만들어 갱신되는 Reference가 Normal Area에 더욱 가까워지게 유도하는 역할을 한다.

ReLu 함수 내부에서는 현재 갱신 중인 w의 mse값에서 산출된 thres 값을 빼는 연산이 존재한다. 이는 즉, 데이터 w의 mse가 수식 앞에 존재하기에 현재 갱신 중인 w가 model이 판단하기에 Anomaly라면 w의 mse 수치가 thres 보다 높을 것이기에 ReLu 내의 값이 양수가 될 것이며, 반대로 w 데이터가 Normal이면, ReLu 내부의 값이 음수가 될 것이다. ReLu는 0 이하의 값은 0으로 반환하고 0을 초과하는 값은 해당 수치가 그대로 반환되는 로직을 지니기에 ReLu 내부의 값이 음수라면 loss_1의 값이 0으로 양수라면 0이 아닌 값이 반환된다. 즉, Optimizer는 현재 loss를 줄이는 방향으로 업데이트를 수행하기에 데이터 w는 ReLu 내부의 값이 음수가 되어 0이 반환되도록 w를 갱신하게 되고 w는 loss1에 의해 모델이 판단했을 때 Normal 으로 판단되도록 이동하게 된다. 이렇듯 loss1은 데이터 w가

Normal Area 방향으로 갱신시키는 역할을 한다. 두 번째 loss인 loss2의 수식은 수식(2)와 같다.

$$loss2 = L\_2\,Norm(w - Anomaly) \quad (2)$$

loss2는 갱신 중인 데이터 w와 원본 데이터인 Anomaly Data와의 L_2 Norm 값이다. 즉, 해당 수치는 두 데이터 간의 유클리디안 거리 값을 의미하며 해당 Optimizer가 loss2 값을 줄이는 방향으로 갱신시킨다는 것은 Fig.5처럼 갱신 중인 데이터 w가 Original Data 방향으로 이동하게 되는 것을 의미한다.

선행 연구된 Han[9]의 Reference 산출 로직에서는 Reference 산출을 위해 앞서 2가지 loss 산출 로직을 설계하였다. Reference는 현재 학습된 모델이 판단하기에 Normal이면서 Anomaly와 가장 가까운 값이다. 즉, Reference는 우선 모델이 판단했을 때 Normal 이여야 하며, 데이터가 Normal Area 범주에 속하면서도 선택한 Anomaly Data와 최대한 가까운 값 이여야 한다. 따라서 선행연구에서는 w를 Normal Area로 이동하게 하는 loss_1과 데이터 w가 원본 Anomaly Data로 이동시키는 loss_2를 합친 최종 loss인 Final_Loss를 활용한다. 하지만 앞서 산출한 loss1과 loss2는 서로 반비례하는 성향이 나타나기에 두 loss를 합쳐 활용하기 위해서는 두 loss의 영향력을 조절하는 매개변수가 필요하다. Reference는 정상 범주에서 속하면서도 Anomaly와 가까운 값이기에 Anomaly에 가까워지려는 loss2의 영향력보다 정상 범주로 이동시키는 loss1의 영향력이 더 강해야 한다. 따라서 Final_loss 값은 수식(3)과 같이 loss2에 lbd라는 작은 소수 값을 곱해줌으로써 Final_Loss에서 loss1의 영향력을 높이고 반

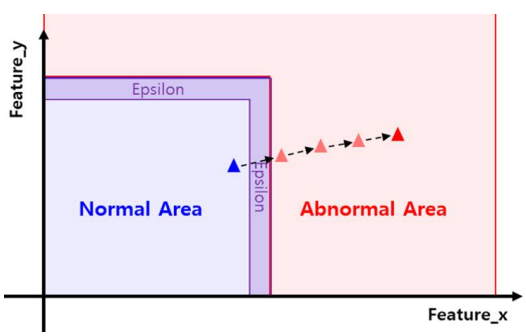


Fig. 5. loss2 based Data(w) update

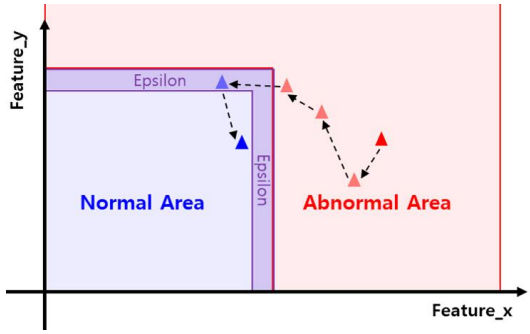


Fig. 6. Final Loss based Data(w) update

대로 loss2의 영향력을 줄임으로써 앞서 언급된 Reference가 산출되도록 유도한다.

$$Final\_Loss = loss1 + lbd * loss2 \quad (3)$$

이를 통해 Final_Loss가 줄어드는 방향으로 업데이트가 이루어지면 Fig.6처럼 갱신 중인 w 값은 원본 Anomaly Data에서 시작하여 Normal이면서 Anomaly와 가장 가까운 Decision Boundary 주변의 값으로 Reference가 산출되게 된다.

이렇듯 Reference는 비 지도학습 model에 의하여 산출된 Anomaly Data에 대해 해당 데이터와 가장 가까우면서도 Normal Data 수치를 산출함으로써, 원본 Anomaly Data의 비교 대상이 되어주고 두 데이터 사이의 거리를 통해 Anomaly 판단 원인에 대한 Interpretation을 직관적으로 제시한다.

3.3 Reference 기반 Nearest Real Data를 통한 더욱 명확한 Interpretation 제공방법론

앞서 우리는 Reference 산출 로직을 분석하였으며, 이를 통해 업데이트되는 데이터 추이를 해석해 보았다. 산출결과 Final_Loss 값이 줄어드는 방향으로 데이터 (w) 값의 갱신이 일어나면 데이터가 점점 anomaly에서 정상 범주로 이동하는 것을 확인할 수 있었다. 하지만 원본 코드에서 산출되는 Reference는 Optimizer에 의해 최적화되며 갱신되어 산출되는 추정치이기 때문에 실존하기 어려운 value가 산출될 수도 있음을 식별하였다.

따라서 본 논문에서는 더욱 직관적인 해석을 위해 산출된 Reference를 실존 데이터로 대체하는 기법을 제시한다. Reference를 대체하는 실존 데이터 탐색방법은 산출된 Reference와 Train 데이터 세

트에 존재하는 데이터에 대해 L_2 Norm 값을 계산한 후 이중 가장 오차가 가장 작은 데이터를 선정한다. 즉, Reference와 가장 가까우면서도 실제로 존재하는 데이터를 Reference value로 대체하는 로직을 제안함으로써 실존하는 데이터를 기반으로 Anomaly Data와 비교분석을 수행할 수 있다. 해당 결과를 Interpretation으로 제공함으로써 이전의 추정치인 Reference를 기반으로 비교 분석하는 기법보다 실존 데이터를 기준으로 비교하기에 더욱 직관적인 Interpretation 제공이 가능하고 분석가의 검토 활동의 효율성 향상에 이바지할 수 있을 것으로 기대된다.

IV. 실험 결과

4.1 Dataset & Featuring Method

Yoo[11]등에 의하면 EDR(Endpoint Detection and Response)은 기업 시스템 환경에서 시스템 환경에서 실제 업무의 적용 단계인 단말(Entpoint)에서 발생하는 의심스러운 행위를 탐지하고 통제하는 가장 핵심적인 영역이라 말한다. 이렇듯 EDR 분야의 최종목적은 EndPoint에서 실시간으로 악의적인 활동을 탐지하고 즉각적인 대응을 수행하는 것이다. Sjarif[12] 등은 하루에도 수많은 데이터가 발생하는 환경에서의 AI 도입의 필요성은 매우 크며 EDR 분야에 AI 도입에 따라 얻을 수 있는 많은 이점을 소개한다. 하지만 실시간 대응을 중요시하는 EDR 분야에서 AI를 효과적으로 활용하기 위해서는 AI가 실시간 대응이 가능하도록 매우 빠른 처리를 수행해야 하는데 EDR에서 수집되는 정보는 매우 많고 이 정보를 모두 반영하기에는 AI 모델의 학습량도 많아지며 동시에 결과 산출을 위한 계산 량도 증가하기에 EDR의 핵심 목표인 실시간 대응이 어려워질 수 있다. 이를 해결하기 위해선 수집되는 정보 중 Anomaly 탐지에 중요한 역할을 하는 소수의 정보만 선정하여 활용함으로써 학습량을 줄이면서도 Anomaly를 잘 탐지할 수 있는 핵심정보를 선정하는 작업이 필요하다.

또한, EDR에서 수집되는 정보들은 수치화된 값일 수 있으나 문자형태의 정보가 존재할 수 있다. 이러한 정보는 수치화된 정보로 학습을 수행하는 AI 모델에 원본 데이터 그대로 적용할 수 없다. 따라서 문자형태의 데이터를 AI가 학습할 수 있게 수치화시켜주는 NLP(National Language Processing) 기법을 활용해야 하며, 원본 데이터가 지닌 정보가 수치로도 잘 반영되도록 적절한 NLP 기법을 설계해야 한다.

우리가 실험에 활용할 데이터는 EDR 분야인 MONSTER Event Collector에 의해 기록된 총 1058개의 한글 워드프로세서의 동작 과정이 담긴 원본 로그를 전 처리한 데이터이다. 해당 원본 로그에는 새로운 Hwp 프로세스가 생성될 때마다 생성된 Process에 대한 주요 정보가 기록되며, 원본 로그의 예시는 Fig.7과 같다. 로그에 기록되는 주요 정보로는 Process 생성 시간, 고유 식별정보인 process_guid, 부모 프로세스의 식별정보인 parent_guid, 여러 flag 정보, 프로세스 접근 권한 등급, 환경 변수 문자열, 전자 서명자 이름, 서명 검증 결과 등 공격 탐지에 효과적일 수 있는 39개의 정보를 담고 있으며, 원본 로그 데이터 라벨 구성은 Table 1과 같다. 1054개의 Normal Process에는 평상시에 나타나는 행위에 따라 중심 프로세스인 Hwp.exe 하위로 Hpdf.exe, HimTrayIcon.exe, HncLogUploader.exe 와 같이 일반적으로 Hwp 사용 시 정상적으로 연계되는 프로세스들이 주로 연계되는 것을 확인할 수 있었고, 반대로 4개의 Malware Process에서는 Choe[13] 등이 언급한 대로 한글 문서 형 악성코드 실행을 위해 eps 공격이 취약한 한글 8.6 버전에서 실행되었으며, 공격을

Fig. 7. Original Log - MONSTER Event Collector

Table 1. Configuration of Dataset

Dataset	Malware	Normal	Total
Train/ Test	4	1,054	1,058

위해 Normal일 때는 나타나지 않는 gswin32c.exe, Powershell.exe, conhost.exe와 같이 평상시에는 연계되지 않는 Anomaly 한 Process들과 취약 버전의 프로그램들이 연계되는 것을 식별할 수 있었다.

위와 같은 현황이 나타나는 것에 집중하여 연계되는 프로세스정보를 기반으로 Anomaly Detection을 수행하는 Featuring 기법을 한다. Featuring을 위해 각 Process의 연계정보를 파악할 수 있는 Parent_guid, Process_guid 필드와 Process Name 추출을 위해 Image_path를 필드를 활용한다. 따라서 제안하는 기법은 기존 원본 로그에 존재하는 39개의 정보 중 3개의 필드만 활용하여 계산량을 대폭 줄임으로 경량화 된 Anomaly Detection 방안을 설계하고, 경량화 된 데이터 세트를 기반으로 EDR에서 추구하는 실시간 대응이 가능한 AI 산출을 도모한다.

4.2 AI 기반 침입분석 결과

Process는 크게 자신이 하위로 발생시키는 Process가 있고 반대로 상위에서 자신을 실행시킨 Process가 존재하는데 자신을 실행시킨 Process를 Parent Process라고 하며, 반대로 자신이 실행시킨 Process를 Child Process라고 한다. 이렇듯 Process는 부모 노드와 자식 노드가 존재하는 자료형태인 트리와 비슷한 형태로 연계가 이루어지며, 특정 행위를 수행하기 위해서는 해당 행위를 수행할 수 있는 Process가 하위에 연계되어야 한다. 따라서 공격 프로세스에는 악성 스크립트를 실행시키기 위해 정상에는 연계되지 않는 Powershell.exe와 같은 Anomaly Process가 연계되고 해당 정보가 log 데이터로써 기록된다.

본 탐지 기법에서는 위에서 언급하였듯이 Attack일 때는 Anomaly Process 연계과정이 나타난다는 특징에 집중하여 Process 연계정보를 Featuring의 대상으로 지정하였다. 이러한 Process 연계정보는 Process의 식별자인 Process_guid 정보를 재귀적으로 탐지하여 선택한 Process에 대한 전체적인 연계정보를 추출할 수 있으며, 추후 이렇게 산출된 각 Process 별 전체적인 연계정보를 Process chain이라 칭하기로 한다.

앞서 언급된 MONSTER Event Collector로 수집된 위협 헌팅에 필요한 프로세스 행위정보가 담긴 원본 Event log들을 대상으로 학습데이터 세트 구축 즉, 로그 데이터 전처리를 수행한다. 제안하는 기법은 Fig.8과 같다. 우선 Process의 식별자인 Process guid를 통해 Parent_Guid와 비교해가며 재귀적 탐색을 통해 Process chain 정보를 추출한다. 이후 추출된 Tree와 같은 구조를 지니는 Process chian에 Depth를 3으로 제한해줌으로써 산출되는 Normal일 때와 Attack일 때 추출되는 Process chain의 정규화 및 경량화를 추구하였다. 이렇게 산출된 Process chain 정보를 토대로 Featuring을 수행하며 Table 2와 같이 Featuring은 크게 2가지 대상에 대하여 수행하여

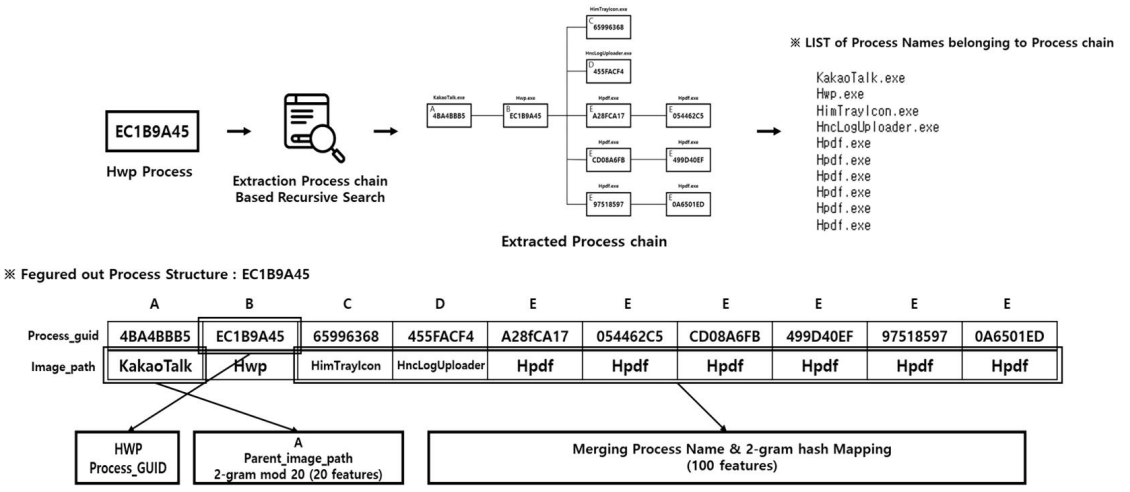


Fig. 8. Process chain Featuring Framework

Table 2. Configuration of Featuring

Target	Parent Process	Child Process	Total
Feature count	20	100	120

총 120개의 Feature 들이 산출된다.

첫 번째 Featuring 대상은 Parent Process의 Image_path이다. 본 Featuring 방법에서는 Process의 연계과정을 학습데이터로써 Featuring을 수행하였으며, 그중 Parent Process의 Image_path 정보 즉, 현재 프로세스가 어떤 경로에 있는 Process에 의하여 실행되었는지를 Anomaly Detection 기반 모델에서 주요한 정보로 활용될 것으로 판단하였다. 따라서 해당 정보를 "\"를 경계로 토큰화시켰으며 해당 정보를 다시 2-gram hash 20 mod mapping을 해줌으로써 기존 문자열 형태인 Parent_Process의 image_path 정보를 총 20개의 Feature 들로 전처리를 수행하였다.

Featuring 두 번째 대상으로는 하위로 연계된 child process들의 Process name 정보이다. 해당 정보들을 하나로 이어붙인 후 char 단위 2-gram hash 100 mapping을 수행하여, 연계된 Process의 정보가 총 100개의 Feature 들로 산출된다. child process들의 이름을 하나로 이어붙인 후 2-gram을 수행하기에 Process 연계 중간에 평소에 나타나지 않는 Process가 실행되거나 평소보다 많은 Process가 연계된다면, 특정 자릿수의 Feature의 수치가 커지거나 정상일 때는 Mapping 발생하지 않는 Feature 자리에 Mapping이 수행되는 등 제안하는 Featuring 기법을 통해 각 Process에서 연계되는 Process chain 정보를 수치화하여 120개의 Feature에 반영할 수 있게 된다. 앞서 산출된 데이터 세트를 통해 Anomaly Detection을 수행하기 위해 본 논문에서는 Oh[14]들도 위협탐지를 위해 비 지도학습을 기반으로 심층 신경망 결과에 Softmax 함수를 연결한 Autoencoder 모델을 사용한다. 모델 세부 계층은 Encoder, Decoder가 각각 3개의 계층으로 이루어져 있어 총 6계층을 지니는 Autoencoder 이다. 각 계층은 Input data의 Feature 수에 따라 Encoder에서 중간 노드를 전체 Feature 수의 100%, 75%, 50%, 25%로 압축시킨 후 Decoder

	process_guid	Process_chain_Name	Parent_path_feature_0	Parent_path_feature_1	...	Child_path_feature_98	Child_path_feature_99	Label	rmse
1054	{AB30096E-24DC-43C8-97BC-7EC104EAC4E0}	Hwp.exe	0	1		2	6	Execution	0.489122
1056	{93776A11-3728-4237-8FD4-4504437104B6}	Hwp.exe	0	1		2	6	Execution	0.489122
1055	{2C9904A9-84FC-4B27-82F6-24A40DF8916C}	Hwp.exe	0	1		2	6	Execution	0.489122
1057	{8A8F05C3-9713-4B44-B479-B085A09CDCFC}	Hwp.exe	0	1		0	0	Execution	0.336468
470	{EC1B9A45-8914-4311-A134-0381BF37EEE4}	Hwp.exe	0	0		0	0	normal	0.216727
...	...	...	...	...	...	...	...	...	...
627	{6E3FC930-9409-4024-841A-57DDE4A48788}	Hwp.exe	0	1		0	0	normal	0.013638
625	{652866EB-5E5B-4E3A-8EC2-7B39A2125540}	Hwp.exe	0	1		0	0	normal	0.013638
622	{B253B671-F455-470F-AEF0-2D15815B6824}	Hwp.exe	0	1		0	0	normal	0.013638
621	{6E949BCB-ABB2-49EC-844A-681C756A5C9F}	Hwp.exe	0	1		0	0	normal	0.013638
529	{98514A53-9D9C-4FBD-A9AA-CAC69E210EE8}	Hwp.exe	0	1		0	0	normal	0.013638

Fig. 9. Result of Anomaly Detection

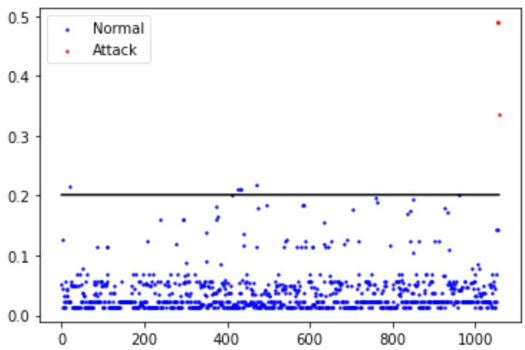


Fig. 10. Result(Plot) of Anomaly Detection

에서 반대로 수행되어 원본 데이터를 복원한다. 각 데이터에 대한 Anomaly Score는 mse를 산출한 후 루트를 적용한 rmse를 사용하였다.

본 논문에서 실험할 1058개의 Hwp.exe 프로세스에 대한 Anomaly Detection 수행결과는 Fig.9, Fig.10과 같다. 산출결과 1058개의 Process에 존재하는 4개의 공격 Process가 Autoencoder에 의하여 모두 최상위에서 탐지되는 것을 확인할 수 있다.

4.3 Reference 기반 해석

4.3절에서는 4.2절에서 수행된 1058개의 Hwp.exe 프로세스에 대한 Anomaly Detection의 최상위 결과에 대해 Anomaly 적절성 검증을 수행한다. 우선 최상위 Anomaly 데이터에 대한 Reference 수치를 산출하고 원본 데이터와 비교를 수행하여 Anomaly 판단결과에 대한 적절성을 검증한다. 이후 산출된 Process chain 정보를 가시화하여 Reference 기반 해석의 적절성과 AI 산출결과에 대한 적절성을 검증한다. 최상위 결과는 1054

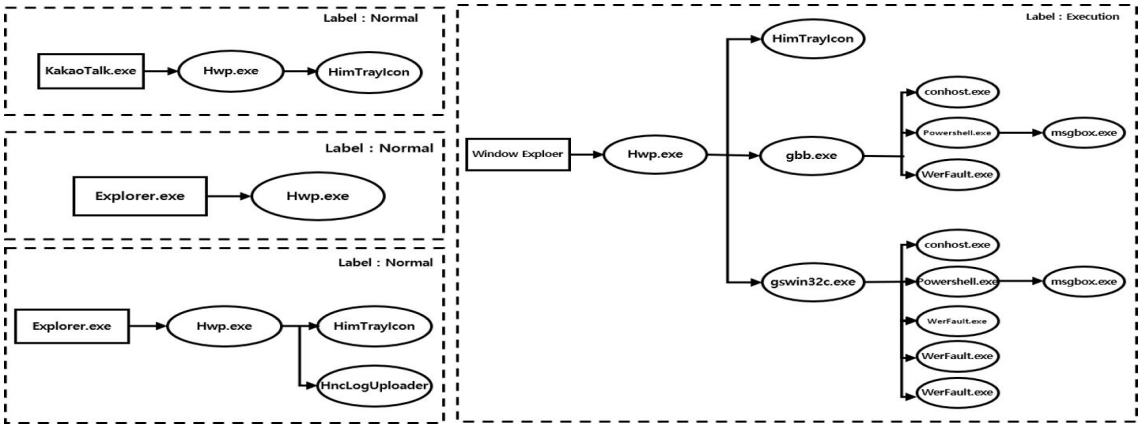


Fig. 11. Attack Process - Anomaly Process chain

Hwp 프로세스이며 해당 프로세스에 대한 Process chain 정보는 Fig.11과 같다.

Fig.11의 우측에 보이는 Attack Process는 좌측의 Normal Process와 확연한 차이를 보이는 Process Chain을 지닌다. 이는 공격을 위해 평소에는 나타나지 않는 Attack 관련 프로세스들이 연계되었기 때문이다. 즉, 공격에서만 나타나는 이러한 Anomaly Process chain 정보는 앞서 제안한 Fig.8의 2-gram 기반 hash mapping Featuring 기법을 통해 전 처리되어 Feature에 Anomaly한 수치는 Value로 Mapping 된다. 따라서 1054 Hwp Process는 Anomaly 연계정보가 Feature에 반영되었기에 Fig.9와 Fig.10처럼 최상위에서 Anomaly로 탐지되었음을 식별할 수 있으며, 1054 Hwp 프로세스에 대한 Reference 산출결과는 Fig.12, Fig.13과 같다.

Fig.12는 Reference 산출결과를 Table 형태로 제공한 결과이다. Feature Description은 Feature Name을 의미하며 가장 우측인 Local Importance를 기반으로 상위 Feature들이 정렬되어 제공된다. Value in Anomaly는 1054번 Hwp Anomaly Data의 원본 Value, Value in Reference는 산출된 Reference의 Value이다. 각 Feature Value의 의미는 다음과 같다. 우선 1054 Hwp Process에서 연계된 Child Process들의 Name을 이어 붙인 후 2-gram으로 분할한다.

이후 분할된 길이 2의 문자열들을 sha-256를 통해 hash 처리 후 산출된 16진수의 문자열을 10진수로 변환하고 산출된 10진수 수치에 mod 100을 적용하면 앞서 2-gram으로 분할된 길이 2의 문자열들은 각각 0~99 사이의 정수인 수치로 전 처리된다. 즉, Child_feature_90이 1이라는 것은 1054 Hwp Process에서 연계된 Process의 이름을 이어 붙인 후 2-gram으로 분할된 길이 2의 문자열 중 hash 처리한 후 10진수로 변환하여 mod 100을 적용하였을 때 산출결과가 90인 문자열이 1개라는 것이다. 따라서 Anomaly한 Process가 연계되면 2-gram으로 평소에는 나타나지 않는 문자열이 생성

Visualize Interpretation (Table View)

Feature Description	Value in Anomaly	comp.	Value in Reference	Local Importance
Child_path_feature_90	1.0	>	0.791	0.044
Child_path_feature_19	1.0	>	0.792	0.043
Child_path_feature_5	1.0	>	0.794	0.043
Child_path_feature_97	1.0	>	0.795	0.042
Child_path_feature_63	2.0	>	1.592	0.042
Child_path_feature_34	1.0	>	0.796	0.041
Child_path_feature_81	1.0	>	0.797	0.041
Child_path_feature_67	1.0	>	0.798	0.041
Child_path_feature_36	2.0	>	1.598	0.04
Child_path_feature_16	2.0	>	1.598	0.04

Fig. 12. Reference Interpreter(Table View)

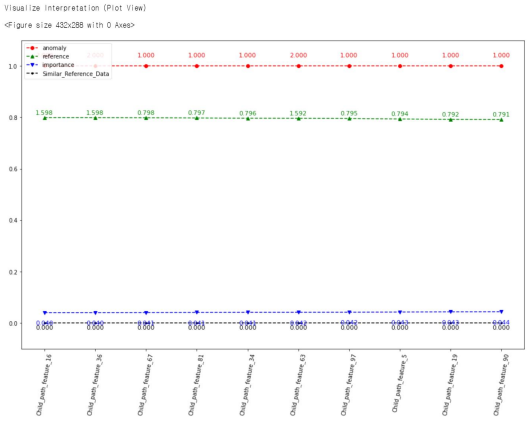


Fig. 13. Reference Interpreter(Plot View)

될 것이고 이는 즉, 정상일 때는 Mapping이 되지 않는 Feature에 값이 Count 되는 것을 의미한다. 기존 기법에서는 앞선 2가지의 수치만 제공하여 비교함으로써 Interpretation을 제공하였으나 본 논문에서는 Local Importance를 추가로 제공하여 더욱 명확한 Interpretation을 제공한다.

Local Importance는 각 Feature 별 원본 Anomaly Data의 Value와 산출된 Reference Value의 오차를 기반으로 산출된다. 각 Feature는 서로 다른 Feature Range를 지니기에 형평성을 위해 Feature 별로 Minmax Scaling을 수행한 후 오차를 산출하고 이를 제곱함으로써 Local Importance가 산출하였다. 따라서 Local Importance는 오차가 클수록 높은 수치를 지니게 되며 해당 수치를 Local Importance로 활용한 이유는 Feature Value가 Anomaly 할수록 모델은 제대로 된 복원을 수행하지 못하기에 오차가 커지기 때문이다. 즉, Local Importance가 높다는 것은 Anomaly 판단에 크게 기여한 Feature를 의미한다. 앞서 정리한 개념을 바탕으로 산출결과를 해석하면 Local Importance가 높은 Child_path_90, Child_path_19, Child_path_5 등의 Feature가 Anomaly 판단에 주요한 영향을 미친 Feature임을 판단할 수 있다. 산출된 Reference value를 보면 각각 0.79 언저리의 값이 산출된다. 정상 범주 중 Anomaly와 가장 가까운 Value라는 Reference 의미에 의거 하여 해석을 수행하면, Child_path_90, Child_path_19, Child_path_5는 산출된 Reference Value가 각각 0.791, 0.792, 0.794 으로 해당 수치들이 정상으로 판단될 수 있는 가장 큰 값 수치임을 의미한다.

본 Attack 데이터인 1054 Hwp Process는 Child_path_90, Child_path_19, Child_path_5가 모두 1이기에 산출된 Reference 수치를 모두 초과하였음으로 Anomaly로 판단되었다는 명확한 해석을 가시적으로 보여준다. 하지만 본 데이터에 적용한 Featuring 기법은 2-gram Hash Mapping이다. 따라서 Mapping 된 횟수를 Count 하여 featuring이 수행되기에 Feature value는 0, 1, 2, ..., n 등 정수 값을 지니게 된다.

하지만 앞서 산출된 Reference Value는 실수 값으로 이는 제안하는 Featuring 방법으로 산출될 수 없는 데이터임을 식별할 수 있다. 그 원인으로는 Reference Value는 SGD Optimizer에 의해 최적화되어 Normal 으로 판단되는 Decision Boundary 주변의 추정치로 산출되기에 점진적 최적화 과정에서 Loss를 줄여가는 방향으로 Feature Value를 최적화 시 실수 단위로 갱신되기 때문이다. 따라서 산출된 Reference 수치는 제안하는 Featuring 기법으로 인해 산출될 수 없는 실수 Value를 지니는 데이터가 산출된 것으로 짐작할 수 있다. Reference 기반 XAI 기법의 가장 주된 목적은 비 지도학습에서 비교 대상으로 Reference로 산출한 후 원본 Anomaly와 비교를 통해 해석 가능성을 제공하는 것이다. 하지만 실제로 Reference를 산출한 결과 산출된 Reference가 Real Data가 아니라 Optimizer에 의해 탐색 된 추정치이기 때문에 상대적으로 해석 가능성이 떨어지게 되는 것을 식별할 수 있다. 따라서 본 실험에서는 Reference를 대체할 수 있는 Reference와 가장 유사한 실존 데이터를 L_2 Norm을 기반으로 Train 데이터 세트에서 탐색하고 이를 Nearest Real Data라는 항목으로 추가하여 해석 가능성을 향상시키는 기법을 추가로 제안한다.

Nearest Real Data 데이터는 학습된 Autoencoder가 판단했을 때 정상이라고 판단한 데이터를 대상으로만 탐색을 수행하였으며, Fig.14와 같이 산출된 Nearest Real Data의 Value를 Table Interpreter에 Nearest Real Data 항목을 추가하여 제공하였다. Nearest Real Data를 산출 후 향상된 Interpretation 정보를 해석한 결과는 아래와 같다. 1054 Hwp Process의 Child_path_feature_90 value는 1이며, 산출된 Reference value는 0.791이다. 제안하는 Featuring 기법으로 인해 모든 feature value가 정수 값을 지니는 현황에서 실수 값으로 산출된 Reference는 비교분석을 통해 Interpretation을 제공하기에는 그 의미가 명확하지 않다. 그에 비해 Nearest Real Data의 값은 해당 Feature가 0임

Visualize Interpretation (Table View)

Feature Description	Value in Anomaly	comp.	Nearest Real Data	Local Importance	Value in Reference
Child_path_feature_90	1.0	>	0.0	0.044	0.791
Child_path_feature_19	1.0	>	0.0	0.043	0.792
Child_path_feature_5	1.0	>	0.0	0.043	0.794
Child_path_feature_97	1.0	>	0.0	0.042	0.795
Child_path_feature_63	2.0	>	0.0	0.042	1.592
Child_path_feature_34	1.0	>	0.0	0.041	0.796
Child_path_feature_81	1.0	>	0.0	0.041	0.797
Child_path_feature_67	1.0	>	0.0	0.041	0.798
Child_path_feature_36	2.0	>	0.0	0.04	1.598
Child_path_feature_16	2.0	>	0.0	0.04	1.598

Fig. 14. Interpretation added Nearest Real Data

을 확인 할 수 있다. 즉, Child_path_feature_90은 Normal일 때는 항상 0이라는 수치를 지니고 공격일 때만 Anomaly Process가 연계되어 Count되는 Feature임을 짐작할 수 있고 이는 즉, Attack에서만 나타나는 Powershell, Cmd 같은 Process가 2-gram hash Mapping을 통해 평소에는 나타나지 않는 자릿수에 Mapping 되어 나타나는 Feature임을 짐작할 수 있다.

즉, Child_path_feature_90은 Anomaly Normal 판단에 중요한 역할을 하는 feature이며, 그 밖에도 산출된 상위 Feature들의 Value를 분석해보면 90번 Feature와 동일하게 Attack을 위해 연계된 Anomaly Process들이 2-gram hash mapping에 의해 Featuring 되어 평소에는 0인 Feature들에 Mapping count가 발생한 것을 확인할 수 있다. 이렇듯 기존 추정치로 산출되는 Reference만으로는 명확한 비교를 통한 분석이 모호했던 기존 Interpretation 방법에 Nearest Real Data 항목을 추가함으로써 비교분석을 통한 해석 효율성을 향상시킬 수 있었다.

Fig.15는 원본 Anomaly Data의 Process chain 정보와 산출된 Nearest Real Data와의 chain 정보를 비교한 그림이다. 산출결과 원본 Anomaly Data의 Process chain과 Nearest Real Data의 Process chain의 차이는 매우 크다. 즉, Reference 기반으로 산출된 Nearest Real Data도 Reference와 같이 원본 Anomaly Data와의 확고한 차이를 보여줌으로써 의도에 맞게 산출되었음을 확인할 수 있었다. 또한, Fig.16은 전체 데이터의 분포 정도와 산출된 Nearest Real data가 정말로 정상 중에서는 Anomaly와 가장 유사한지를 확인하기 위해 DBSCAN(eps = 7.5, samples = 5)을 통해 클러스터링하여 cluster 정보를 추출한 후 plot으로 나타내기 위해 2차원으로 T-SNE를 수행한 결과이다.

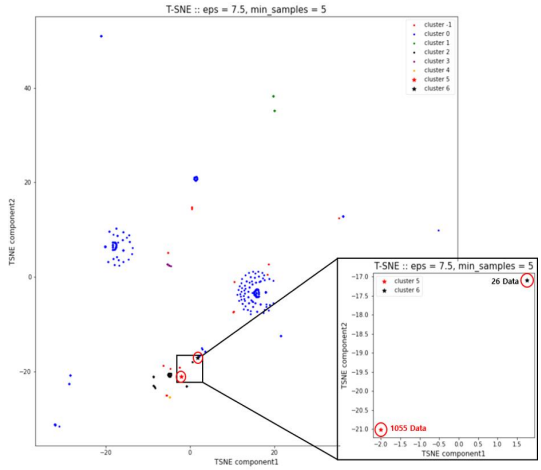


Fig. 16. Verification of each Data(Distance)

해당 plot에 원본 Anomaly Data와 Nearest Real Data의 위치를 확인한 결과 두 데이터는 다른 Normal data들에 비해 가까운 Distance를 보이는 것을 식별할 수 있었다. 즉, Reference의 기본 개념인 정상 중에서 Anomaly와 가장 가까운 데이터를 Reference와의 L_2 Norm으로 산출된 Nearest Real Data도 만족하기에 Nearest Real Data 정당성을 부여할 수 있었고 해당 데이터로 Reference를 대변하여 더욱 효과적으로 Reference 기반 Explainable에 활용할 수 있을 것으로 기대된다.

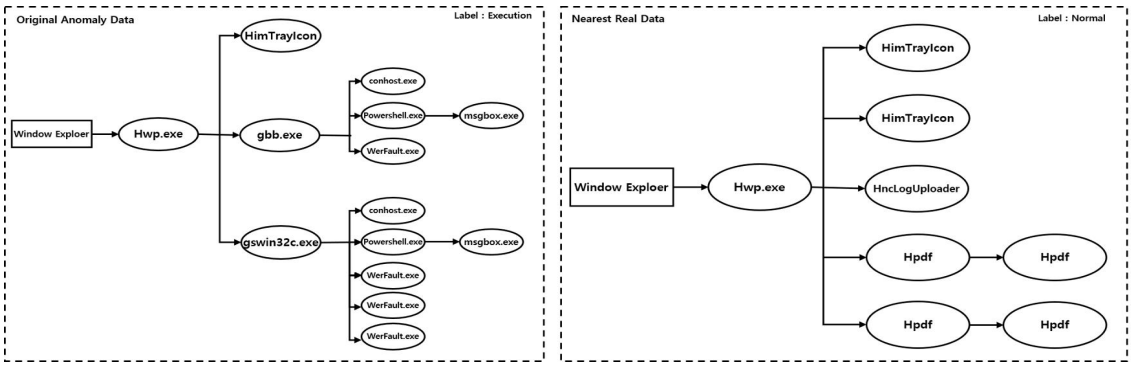


Fig. 15. Verification of each Data(Process chain)

V. 결 론

AI의 성능이 향상되고 좋은 결과를 산출되는 만큼 AI는 Black Box 성격을 지니게 되어 산출된 결과에 대한 해석이 어려워지고 있다. 이러한 성격은 중요한 의사 결정을 내리는 상황이나 의료, 보안 분야와 같이 오 탐지 리스크가 큰 분야에서는 AI 도입의 걸림돌이 되는 상황이다. 따라서 AI 판단에 대한 Interpretation을 제공하는 XAI(eXplainable Artificial Intelligence) 기법이 활발히 연구되고 있다. 하지만 선행연구 된 대부분의 XAI 기법은 지도학습 및 모델의 복잡한 산출 로직을 이해하여 시각적으로 Interpretation을 제공하는 것에 초점을 맞추어 개발되었다. 따라서 비 지도학습 모델 기반의 이상 탐지를 수행하며, 정상과 악성의 차이점을 통해 악성을 판단해야 하는 보안 분야에는 선행연구들을 적용하기에는 부적합하다. 이러한 이유로 인해 보안 분야 현업에서 AI 도입하기에는 어려움이 존재한다. 따라서 비 지도학습 모델에서도 해석 가능성을 지원할 수 있고 정상과 악성 사이의 차이를 통해 효과적인 Interpretation을 제공할 수 있는 Reference 기법을 활용한다.

본 논문의 실험에서는 Reference를 기반으로 비 지도학습에서의 Interpretation을 제공하는 최신 XAI 기법을 발전시켜 더욱 효과적인 해석 가능성 제공을 도모하고자 하였다. 실험 결과 기존 EDR 분야에서의 실시간 대응을 위해 경량화된 Anomaly Detection 산출결과에 적용 시 기존에는 불가능했던 비 지도학습에서의 Interpretation 제공이 가능했다. 산출되는 Reference, Nearest Real Data, Local Importance 등 해석을 위한 주요 정보들을 종합하여 Interpreter를 통해 Anomaly Data에 대한 명확한 Interpretation을 제공할 수 있었으며, 본 논문에서 제안한 Nearest Real Data에 대한 적절성 검증 결과 Reference를 대체 가능한 것으로 판단되었다. 따라서 기존 추정치를 기반으로 비교분석을 진행할 때보다 실존 데이터를 통해 비교분석을 수행함으로써 직관적인 해석 가능성을 제공할 수 있었다. 해당 연구를 통해 기존에는 불가능했던 이상 탐지모델에서의 Interpretation을 효과적으로 제공할 수 있을 것으로 보인다. 향후 연구로는 산출된 Feature와 value들을 더욱 효과적으로 활용하기 위해 Rule로 추출하여 이상 탐지의 효율을 높이는 등 여러 후행 연구를 연계할 예정이다.

References

[1] Cam, Arif, Michael Chui, and Bryce Hall. McKinsey Analytics. "Global AI Survey: AI proves its worth, but few scale impacts", pp. 1-11, Nov. 2019

[2] Kyounga Shin, Yunho Lee, ByeongJu Bae, Soohang Lee, Heeju Hong, Youngjin Choi, Sangjin Lee. "Trustworthy AI Framework for Malware Response", Journal of the Korea Institute of Information Security & Cryptology, 32(5), pp. 1019-1034, Oct. 2022

[3] Adadi, Amina, and Mohammed Berrada. "Peeking inside the black-box: a survey on explainable artificial intelligence (XAI)" IEEE access 6, pp. 52138-52160, Sep. 2018

[4] Das, Arun, and Paul Rad. "Opportunities and challenges in explainable artificial intelligence (xai): A survey", arXiv preprint arXiv:2006.11371, pp. 1-24, Jun. 2020

[5] Yunsu Lee, Kyuil Kim, Sangsoo Choi, Jungsuk So, "AI/X-AI technology research trend for cyberattack detection based on cryptographic communication", REVIEW OF KIISC, 29(3), pp. 14-21, Jun. 2019

[6] Lundberg, Scott M., and Su-In Lee. "A unified approach to interpreting model predictions", Advances in neural information processing systems 30, pp. 1-10, May. 2017

[7] ZHOU, Bolei, et al. "Learning deep features for discriminative localization", In Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2921-2929. Jun. 2016

[8] Liat Antwarg, Ronnie Mindlin Miller, Bracha Shapira, and Lior Rokach. "Explaining anomalies detected by

autoencoders using shapley additive explanations", Expert Systems with Applications 186:115736, pp. 1-37, Dec. 2021

[9] Han, Dongqi, et al. "DeepAID: interpreting and improving deep learning-based anomaly detection in security applications", Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. Nov. 2021.

[10] Montavon, Grégoire, Wojciech Samek, and Klaus-Robert Müller. "Methods for interpreting and understanding deep neural networks", Digital signal processing 73, pp. 1-15, Feb. 2018

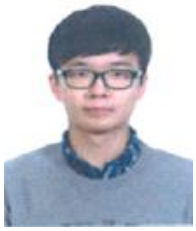
[11] Seung Jae Yoo, "Study on Improving Endpoint Security Technology", Journal of convergence security. 18(3) pp. 19-25, Sep. 2018.

[12] Sjarif, Nilam Nur Amir, et al. "Endpoint detection and response: Why use machine learning?", 2019 International Conference on Information and Communication Technology Convergence (ICTC). IEEE, pp. 283-288, Dec. 2019

[13] MinJi Choe, KangSik Shin, DongJae Jung. "Method to Generate Signature for HWP Malware Detection Based on Threat Factors", Proceedings of the Korean Information Science Society Conference, 49(1), pp. 1312-1314. Jun. 2022

[14] Sungtaek Oh, Woong Go, Mijoo Kim, Jaehyuk Lee, Kim Hong-Geun, SoonTai Park. "Study on IoT threat detection technology through artificial neural network algorithm", REVIEW OF KIISC, 29(6), pp. 59-65, Dec. 2019

〈저자소개〉



이 현 우 (Hyun-woo Lee) 학생회원
2023년 2월: 호서대학교 컴퓨터공학부 졸업
2023년 3월~현재: 호서대학교 정보보호학과 석사과정
〈관심분야〉 악성코드 분석, 침입 탐지, 이상 징후 분석, 정보보호, AI

한 태 현 (Tae-hyun Han) 학생회원
2018년 3월~현재: 호서대학교 컴퓨터공학부 학석사과정
〈관심분야〉 악성코드 분석, 정보보호, AI

박 영 지 (Yeong-ji Park) 학생회원
2020년 3월~현재: 호서대학교 컴퓨터공학부 학석사과정
〈관심분야〉 악성코드 분석, 정보보호, AI

이 태 진 (Tae-jin Lee) 종신회원
2003년 2월: 포항공과대학교 컴퓨터공학과
2008년 2월: 연세대학교 컴퓨터공학과 석사
2017년 2월: 아주대학교 컴퓨터공학과 박사
2013년 1월~2017년 2월: 한국 인터넷진흥원 팀장
2017년 3월~현재: 호서대학교 컴퓨터공학부 교수
〈관심분야〉 시스템 보안, 침해사고 대응, Trustworhty AI