Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.34, NO.3, Jun. 2024 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2024.34.3.417

그래프 기반 이기종 위협정보 분석기술 연구*

이 예 은,[1†] 이 태 진[2‡]
[1,2]호서대학교 (대학원생, 교수)

A Study on Graph-Based Heterogeneous Threat Intelligence Analysis Technology*

Ye-eun Lee,[1†] Tae-jin Lee[2‡]
[1,2]Hoseo University (Graduate student, Professor)

요 약

현대 기술의 발전과 인터넷의 보급이 확대되면서 사이버 위협도 증가하고 있다. 이러한 위협에 효과적으로 대응하기 위해 CTI(Cyber Threat Intelligence)의 활용에 대한 중요성이 커지고 있다. 이러한 CTI는 과거의 사이버 위협 데이터에 기반하여 새로운 위협에 대한 정보를 제공하지만, 데이터의 복잡성과 공격 패턴의 변화 등 다양한 요인으로 인해 분석의 어려움을 겪고 있다. 이러한 문제를 해결하기 위해, 본 연구는 다차원적 관계를 포괄적으로 나타낼 수 있는 그래프 데이터의 활용하고자 한다. 구체적으로는 악성코드 데이터를 대상으로 이기종 그래프를 구축하고, metapath2vec의 노드 임베딩 방법을 활용하여 사이버 공격 그룹을 더 효과적으로 식별하고자 한다. 결론적으로 토폴로지 정보를 기존 악성코드 데이터에 추가로 활용하였을 때 탐지성능에 미치는 영향을 분석함으로써, 사이버 보안 분야에 새로운 실질적 적용 가능성을 제시하며, CTI 분석의 한계를 극복하는 데 기여하고자 한다.

ABSTRACT

As modern technology advances and the proliferation of the internet continues, cyber threats are also on the rise. To effectively counter these threats, the importance of utilizing Cyber Threat Intelligence (CTI) is becoming increasingly prominent. CTI provides information on new threats based on data from past cyber incidents, but the complexity of data and changing attack patterns present significant analytical challenges. To address these issues, this study aims to utilize graph data that can comprehensively represent multidimensional relationships. Specifically, the study constructs a heterogeneous graph based on malware data, and uses the metapath2vec node embedding technique to more effectively identify cyber attack groups. By analyzing the impact of incorporating topology information into traditional malware data, this research suggests new practical applications in the field of cyber security and contributes to overcoming the limitations of CTI analysis.

Keywords: Cyber Attack, Metapath2Vec, Attack group, Decision support, unsupervised model

I. 서 론

전 세계적으로 기술이 발전하고 인터넷 사용량이 증가함에 따라, 개인, 정부, 기업을 포함한 모든 주체의 활동이 점차 사이버 공간으로 옮겨가게 되면서 이는 사이버 위협의 증가로 이어지게 되었다. 이러한 이유로 사이버 위협에 대응할 수 있는 전략의 중요성이 높아지게 되면서 과거에 기반해 새로운 사이버 위협에 대한 깊이 있는 이해를 제공하는 CTI의 활용이 강조되고 있다. CTI는 일반적으로 공격에 사용된 URL, HASH, IP, Email, Malware, Domain 등의 IoC(Indicators of Compromise)와 공격

Received(01. 08. 2024), Modified(1st: 03. 08. 2024, 2nd: 04. 26. 2024), Accepted(04. 26. 2024)
* 이 논문은 2023년도 호서대학교의 재원으로 학술연구비 지원을 받아 수행된 연구임(20230316)
† 주저자, judiaye4477@gmail.com
‡ 교신저자, kinjecs0@gmail.com(Corresponding author)

방법에 대한 TTPs(Tactics, Techniques, and Procedures) 정보를 포함하고 있다. 이러한 정보를 분석하는 과정은 공격그룹 및 공격 기술의 식별 그리고 위협환경 전반의 분석에서 중요한 단계이다. 그러나 데이터의 복잡성이 증가하고 공격 패턴이 지속적으로 변화하면서 정확한 분석이 어려워 효과적인 위협탐지와 대응 전략 수립에 제약을 주고 있다. 예를 들어, 다단계로 정교하게 특정 대상을 지속해서 공격하는 APT(Advanced Persistent Threat) 공격과[1] 알려지지 않은 취약점을 이용하는 Zero-day 공격처럼[2] 침해 공격은 정상적인 활동과 악성 활동이 연속적으로 발생하기 때문에 일반적으로 데이터 구조가 복잡하여 분석의 어려움을 주게 된다.

이와 같은 복잡한 침해 공격에서 나타나는 연결 패턴과 행동 간의 상관관계를 효과적으로 인식하고 해석하려는 방안의 중요성이 점차 높아짐에 따라 AI(Artificial Intelligence)를 활용하는 많은 연구가 진행되고 있다. 그러나 대부분의 AI 모델들은 2차원 데이터에 중점을 둔 학습방식을 채택하고 있어, APT, Zero-day 공격과 같은 복잡한 구조의 데이터를 분석할 때 한계를 보인다. 따라서 이를 보완하고자 데이터의 구조적 정보를 포함하여 높은 표현력을 제공할 수 있는 그래프 데이터를 활용한 방안을 제안하고자 한다. 그래프 데이터는 다양한 관계적 정보를 종합적으로 표현할 수 있으며, 네트워크 내 숨겨진 패턴을 드러내고 다차원적인 데이터 관계를 통해 정확한 분석과 예측을 할 수 있다는 장점이 있어 최근 주목받고 있는 기술이다.

본 논문에서는 10개 공격그룹을 라벨로 하는 악성코드 데이터를 대상으로, 두 가지 버전의 이기종 그래프를 구축하여 실험하였다. 첫 번째 버전의 그래프는 악성코드와 공격 기술의 두 가지 노드를 포함하며, 두 번째 버전의 그래프는 악성코드, 공격 기술, 그리고 공격그룹의 세 가지 노드를 포함한다. 본 연구에서는 특히 악성코드와 관련된 IoC(Malware) 및 특정 TTPs와 관련된 정보(공격 기술, 공격그룹)를 중점적으로 분석하여 사이버 공격 그룹 식별 성능 향상을 위한 방안을 제안한다. 또한, 이기종 그래프에 meta-path 기법을 활용하여 유의미한 임베딩 값을 산출 및 분석함으로써, 앞으로의 활용 가치를 보이고자 한다.

이 접근 방법은 기존의 CTI 분석 방법과 구별되며, 특히 STIX(Structured Threat Information eXpression) 표준을 활용하는 기존 연구들과의 차이점을 갖는다. STIX는 CTI를 구조화하여 표현하는 데 사용되는 언어로서, CTI 정보의 공유와 호환성을 높이는 데 중요한 역할을 한다. 그러나 본 연구에서 제안하는 이기종 그래프 기반 방법은 다차원적 관계 분석, 임베딩 기반 분석, 그리고 시각화 및 분석 유연성 측면에서 STIX2의 그래프 표현 방법과 비교하였을 때 추가적인 분석 용이성과 데이터 관계의 깊이 있는 이해를 가능하게 한다. 결론적으로 그래프 임베딩을 통한 토폴로지 정보가 탐지성능에 미치는 영향과 다양한 관계를 유의미하게 해석할 수 있음을 보여줌으로써 보다 세밀한 위협 식별 및 대응 전략 수립에 기여하는 새로운 기술적 접근법을 제시하고자 하였다. 더불어, CTI의 복잡한 데이터를 그래프를 통해 명확하게 표현하고, 상호 연결된 위협 요소 간의 관계를 분석함으로써, 향후 CTI 분석에서 위협 패턴 분석이나 위협탐지와 같은 분석 유형을 지원할 수 있는 기술임을 보여주고자 한다.

II. 관련 연구

2.1 이기종 그래프 분석기술 동향

일반적인 AI 모델은 유클리드 공간에 정의된 데이터를 활용하여 학습하게 된다. 이미지나 텍스트 데이터와 같은 전통적인 데이터는 이러한 유클리드 공간에서 처리되며, 이 데이터들은 고정된 차원에 배열되고 각 차원이 독립적인 값을 가지는 구조를 통해 데이터 간의 거리를 이용한 유사도 측정이나 예측값 추론 과정에 활용된다. 그러나 시간이 흐름에 따라 데이터의 양이 폭발적으로 증가하고, 그 크기와 복잡성이 커지면서 데이터의 지엽적인 특성을 넘어 구조적 정보와 특성을 모델에 어떻게 반영할지에 관한 연구가 진행되기 시작하였다[3, 4].

이러한 문제를 해결하기 위해서 그래프 신경망(Graph Neural Network, GNN)과 그래프 컨볼루션 네트워크(Graph Convolutional Network, GCN)와 같은 AI 모델이 제안되었다. 이 모델들은 데이터의 구조적 특성을 반영할 수 있도록 설계되어, 데이터의 수치적 특성을 넘어 그 구조와 관계를 깊이 이해할 수 있으며, 이를 기반으로 더욱 정밀하고 복잡한 예측을 가능하게 한다[5,6]. 그래프의 기본 구조는 G = (V, E)로, 여기서 V(Vertex)는 노드의 집합이고, E(Edge)는 노드 간의 연결을 나타내는 선들의 집합을 의미한다. 이와 같은 그래프를 활용하

기 위해서는 각 노드가 특정한 벡터값을 가지도록 하는 노드 임베딩 과정이 필수적이다. 그 이유는 각 노드가 갖는 유의미한 임베딩 값이 활용 가치를 결정하기 때문이다.

본 연구에서는 그래프 신경망 모델들을 직접적으로 활용하지 않았기에 이 장에서는 AI 모델 없이 보안 도메인에서의 데이터 분석을 위한 노드 임베딩과 관련된 연구를 중심으로 다루고자 한다.

2.1.1 그래프 모델 학습이 없는 노드 임베딩 기술

노드 임베딩은 그래프의 요소를 d 차원 벡터로 변환하는 과정을 의미한다[7]. 이를 위해 노드 간의 관계와 특성을 벡터 공간에 표현하며, 유사한 노드들은 이 공간에서 서로 가까운 위치에 매핑 되어지도록 표현된다. 즉, 노드 임베딩의 주된 목적은 유사한 특징을 갖는 각 노드가 임베딩 공간에서 가깝게 위치하도록 하는 것이다. 예를 들어, 소셜 네트워크에서 서로 친밀한 노드들은 벡터 공간에서 가깝게 위치하게 된다. 이와 같은 노드 임베딩을 위해서 일반적으로 AI 모델을 사용할 수 있지만, 별도의 AI 모델 학습 없이 그래프 구조만을 활용해 노드 임베딩 값을 산출할 수 있다.

랜덤워크 기반 접근법은 Fig. 1과 같이 그래프상에서 무작위로 노드를 이동하며 노드 시퀀스를 생성한 후 각 노드의 벡터 표현을 학습한다. B. Perozzi 등의 연구[8]에서는 랜덤워크와 자연어 처리의 word2vec 방법론을 결합한 알고리즘으로 deepwalk 방법을 제안했다. 이 기법은 노드를 단어로, 노드 시퀀스를 문장으로 간주하며, skip-gram 과정을 통해 학습된다. 따라서 확률적 방식으로 노드 간의 장거리 이웃 정보를 포착할 수 있으며, 랜덤워크로 나타나는 노드들에 기반한 임베딩 산출로 인해 효율적이라는 장점이 있다. 그러나 임의로 랜덤워크가 생성되기 때문에, 강한 이웃 관계를 명확히 파악하기 어렵고, 네트워크 구조를 충분히 반영하지 못할 수 있는 한계점이 존재하였다. 이후, A.Grover 등[9] 연구에서는 기존 deepwalk의 단점을 보완하기 위해 'p'와 'q'라는 두 매개변수를 도입한 node2vec 알고리즘을 제안했다.

이 매개변수는 너비 우선 탐색(Breadth-First Search, BFS)과 깊이 우선 탐색(Depth-First Search, DFS) 사이의 균형을 조절함으로써, 노드의 다양한 이웃 관계를 더욱 효과적으로 탐색할 수 있어 해당 노드 임베딩 방법의 핵심적인 부분이다.

그러나 앞선 방법들은 주로 동종(homogeneous) 그래프에서 사용할 수 있는 노드 임베딩 방식으로 비교적 구조가 단순한 동종 그래프 분석에서는 용이하지만, 실제 사회의 복잡한 시스템이나 다양한 유형의 데이터 관계 표현에서는 한계점을 갖는다. 이러한 이유로 동종 그래프가 아닌 다양한 유형의 노드와 엣지를 포함하는 이기종(heterogeneous) 그래프에 적용 가능한 노드 임베딩 방법이 필요하게 되었다.

이를 위해 Y.Dong 등 연구[10]에서는 metapath2vec 이란 노드 임베딩 기법을 제안했다. 해당 노드 임베딩은 사전에 설계된 meta-path를 따라 이동하며 노드 시퀀스를 생성 후 랜덤워크 기반 주변 노드를 예측하게 된다. 이때 meta-path는 이기종 그래프에서 다양한 유형의 노드와 엣지를 통해 복잡한 관계를 정의하는 경로를 의미한다. 예를 들어, '저자-논문-주제' 경로는 특정 저자가 연구하는 주제 간의 관계를 탐색할 수 있게 해주는 경로를 나타낸다. 이런 과정을 통해 명시적인 모델 학습 과정 없이 노드 간의 상대적인 위치를 고려한 임베딩을 생성할 수 있으며, 노드 쌍이 동시에 등장하는 정보를 임베딩 벡터 조정에 활용한다. 이 방법은 비교적 적은 시간과 비용으로 복잡한 네트워크 데이터의 구조를 파악할 수 있는 장점이 있어 그래프 분석 시 유

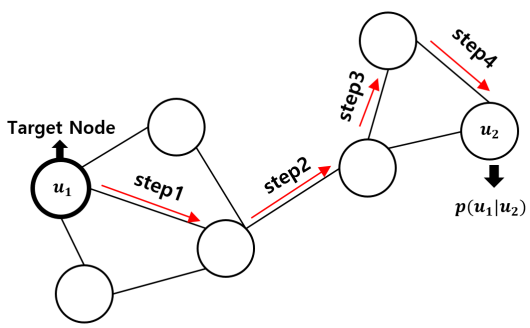

Fig. 1. Example of random walk

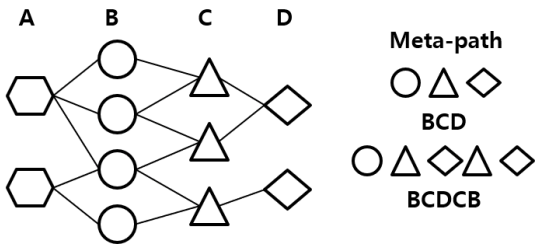

Fig. 2. Example meta-path

용하게 사용하고 있다.

2.1.2 그래프 모델 학습을 통한 노드 임베딩 기술

이 항에서는 이기종 그래프 대상 그래프 AI 모델 학습을 통한 노드 임베딩 방식을 간략하게 소개하고자 한다. 그래프 모델 학습을 통한 노드 임베딩은 그래프 데이터의 복잡성을 효과적으로 포착하고 노드 간의 잠재적인 패턴과 특성을 표현할 수 있다. M. Guan 등의 연구[11]는 이기종 그래프에 대한 노드 임베딩 과정에서, meta-path를 기반으로 생성된 서브 그래프를 대상 그래프 모델을 통해 노드 임베딩을 수행하는 Fig. 3과 같은 방법을 제안하였다. 이 방법론에서는, 먼저 이기종 그래프에 기반하여 meta-path를 디자인하고, 이를 통해 생성된 서브 그래프를 AI 모델의 입력으로 사용하였다. 이 과정에서 GATConv(Graph Attention Network Convolution)와 HetGCN(Heterogeneous Graph Convolutional Network) 모델을 통해 노드 임베딩이 수행되었다. 이후 각 서브 그래프별로 생성된 임베딩 값은 semantic attention 메커니즘을 통해 하나의 벡터로 통합 후 노드 분류, 클러스터링, 링크 예측 작업을 수행하였다. 결과적으로, 이 접근방식은 복잡한 이기종 그래프를 meta-path 를 통해 작은 서브 그래프 단위로 분해함으로써, 동종 및 이기종 이웃을 구분하여 맞춤형 노드 임베딩을 적용할 수 있다는 점을 보여준 연구임을 알 수 있다.

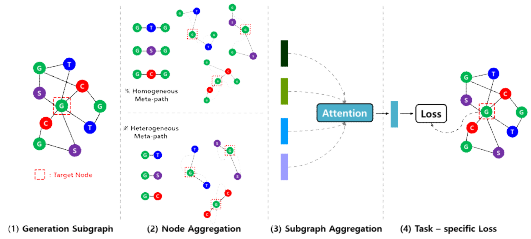

Fig. 3. Example of meta-path based heterogeneous graph analysis

2.2 그래프 기반 CTI 분석기술 동향

2.2.1 IoC 기반 임베딩 분석기술

이 항에서는 IoC를 기반으로 한 복잡한 그래프를 사용하는 위협 분석 방법 연구에 관해 다루고자 한다. 사이버 보안 분야에서 CTI의 중요성이 커지고 있지만, 단독으로 존재하는 IoC 정보만으로는 전반적인 위협을 완전히 파악하기는 어렵다. J. Zhao 등 연구[12]에서는 공격자, 취약점, 악성코드, 디바이스, 플랫폼, 공격유형 등을 포함하는 6개의 노드 유형과 9개의 연결 유형을 가진 이기종 그래프를 사용하여 공격자 중심 분석을 수행하였다. 이때 이기종 그래프는 다양한 소스에서 수집한 위협 관련 데이터를 BIO(Begin, Inside, Outside) 태깅 방식으로 6개 노드 유형에 해당하는 키워드를 추출한 후 이를 딥러닝 모델 학습을 통해 구축하였다. 이후 공격자 노드를 중심으로 17개의 meta-path를 정의하고, 그래프 컨볼루션 네트워크 모델을 사용해 노드 임베딩을 수행하였다. 이 과정을 통해, 연결정보를 고려하여 서로 다른 유형의 노드들에 대한 최종 노드 임베딩 값이 생성되었다. 이후 공격자 노드 대상 분류를 수행한 결과 meta-path 별 서로 다른 성능을 보였는데, 유의미한 관계 정보를 갖는 meta-path 일수록 분류의 성능이 높게 나오는 것을 알 수 있었다. 이때 공격자 노드가 아닌 다른 유형인 취약점 노드의 임베딩 값을 분석한 결과 DBSCAN(Density-Based Spatial Clustering of Applications with Noise)을 통해 클러스터링 된 임베딩 값들이 동일한 취약점 타입을 가지는 것을 볼 수 있었다. 결론적으로, 특정 노드를 대상으로 임베딩이 되는 과정에서 다른 유형의 노드도 유의미하게 임베딩 됨을 알 수 있다. 이를 통해 이기종 그래프에서의 meta-path의 중요성을 알 수 있다.

2.2.2 CTI 기반 임베딩 분석기술

Y. Gao등 연구[13]에서는 CTI를 활용하는 이기종 그래프를 통해 위협 유형을 식별하는 방법론을 연구하였다. 이 연구에서는 악성코드, 주소, 도메인, 이메일 등 4가지 유형의 노드와 5가지 연결로 구성된 이기종 그래프를 구축하고, GCN 모델을 사용해 도메인 노드의 분류 작업을 수행하였다. 도메인 노드 분석에는 도메인 이름의 평균 길이, 문자와 숫자의 분포, 엔트로피, 활성시간, 업데이트 빈도 등 다양한 특성이 활용되었다. 또한, 도메인 노드를 중심으로 한 12개의 meta-path를 사용하여 사이버 위협들 사이의 관계를 모델링 하였다. 이때 연결정보는 노드 간의 유사성을 측정하는 MIIS(meta-graph instances-based threat Infrastructure

similarity) 방법을 통해 meta-path 별로 계산되었으며, 노드 간의 연결 수가 많을수록 유사도가 증가하고, 외부로의 연결이 많을수록 유사도가 감소하는 방식으로 학습된 값과 도메인 특징을 함께 AI 모델의 학습데이터로 사용하였다. 그 결과 도메인 노드의 위협 유형을 식별하였고, 다양한 분류 알고리즘을 통해 성능이 향상되었다. 이 연구 결과는 meta-path의 효과적인 활용이 성능향상에 기여할 뿐만 아니라 분석 대상이 아닌 다른 유형의 노드도 유의미하게 학습된다는 점을 확인할 수 있었다. 결론적으로 이 연구를 통해 사이버 보안 분야에서 다양한 위협정보 간의 관계를 그래프로 구축하고, 목적에 맞는 meta-path의 설계와 적용이 중요하다는 사실을 알 수 있었다.

III. 공격그룹 식별 제안 방법

이번 장에서는 이기종 그래프를 기반으로 공격그룹의 식별 정확도를 향상하려는 방법과 meta-path 기법을 활용한 분석 전략을 제안하고자 한다. 3.1절과 3.2절에서는 단순히 악성코드 데이터만을 활용하는 것보다 토폴로지 정보를 추가로 활용하였을 때 공격그룹 분류 성능이 향상될 수 있는지를 보여주며, 3.3절과 3.4절에서는 다양한 목적으로 설계된 meta-path에 따른 임베딩 결과를 분석 및 검증함으로써 이 기법의 향후 활용 가능성을 탐색하고자 한다.

3.1 공격그룹 식별 전체 프레임워크

Fig. 4는 공격그룹 식별성능을 향상하기 위한 제안된 방법의 전체 구조를 보여준다. 이 방법은 두 주요 영역으로 구분된다. 첫 번째 영역에서는 악성코드인 소프트웨어 노드와 해당 악성코드에 의해 실행된 공격 기술 노드를 포함하는 이기종 그래프 데이터를 생성한다. 두 번째 영역에서는 이 생성된 그래프를 기반으로, 악성코드가 사용한 공격 기술 간의 연결정보를 포함하는 임베딩 값을 metapath2vec을 사용하여 도출한다. 이후, 이 임베딩 정보를 활용하여 악성코드 데이터에 추가함으로써 10개의 공격그룹을 분류하는 과정을 수행한다. 이때 실험을 위해 metapath2vec을 사용한 주된 이유는 실제 환경에서 대용량 데이터가 주로 라벨 없이 제공되기 때문이다. 라벨 없는 데이터에 대해 효과적으로 그래프 모델 학습을 수행하기 위해서는 일반적으로 높은 비용

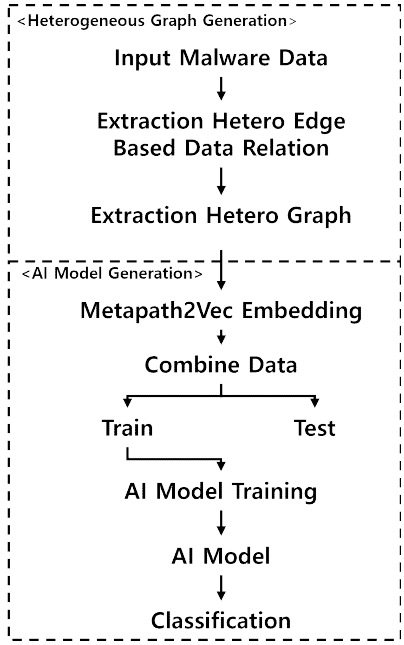

Fig. 4. Proposed Framework(1)

과 많은 시간이 소요되나, metapath2vec는 별도의 AI 모델을 학습시키지 않고도 저렴한 비용과 짧은 시간 내에 복잡한 네트워크 구조를 분석할 수 있다는 효율성 때문에 본 연구에서는 해당 노드 임베딩 방법을 선택하였다. 첫 번째 영역에서 생성된 이기종 그래프 데이터를 바탕으로, metapath2vec 방법을 적용하기 위해 'software-technique-software' 형태의 meta-path를 설계하였다. 이 meta-path는 악성코드 노드 주변의 정보를 구조화하여 임베딩 결과를 도출하는 데 사용되었다. 도출된 임베딩 결과는 t-SNE(t-distributed Stochastic Neighbor Embedding) 기법을 적용해 2차원으로 축소 후 시각화하였으며, 이 임베딩 값을 기존 악성코드 데이터에 추가 특성으로 포함하여 AI 모델 학습을 통해 공격그룹을 식별하는 데 활용하였다. 이러한 접근방식은 복잡한 네트워크 구조 속에서도 효과적인 공격그룹 식별 가능성을 제시함으로써, 보안 분야에서의 응용이 가능할 것으로 기대된다.

3.2 공격그룹 식별모델 생성

다음은 10개의 공격그룹을 식별하기 위한 모델 생성 과정에 관해 설명하고자 한다. 공격그룹 식별을 위한 모델은 XGBoost(Xtreme Gradient

Boosting)를 사용하고자 하였다. 그 이유는 XGBoost는 그래디언트 부스팅 프레임워크를 기반으로 개발되어, 여러 결정 트리를 결합함으로써 강력한 예측 모델을 만들 수 있으며, 현재 다양한 분류 및 회귀 문제에 폭넓게 적용되고 있기 때문이다. 이와 같은 모델의 성능을 최대화하기 위해 데이터에서 정보를 충분히 활용하는 것이 중요하므로, 특징 전처리는 이 과정에서 중요한 역할을 한다. 따라서 각 특징이 지닌 본연의 의미를 살릴 방안을 신중하게 설계해야 한다. 따라서 공격그룹 분류 시 공격자의 패턴을 잘 반영할 수 있는 특징을 선별하여 모델의 성능을 높이고 잡음을 최소화하고자 한다.

이때, 실제 환경에서 수집되는 데이터에는 다양한 원인으로 인해 결측값이 자주 발생하게 된다. 이러한 결측값은 모델 학습에 영향을 미치기 때문에 적절한 처리 방법이 필요하다. 본 연구에서는 결측값 처리를 위한 여러 방법 중 극단 값에 의해 크게 영향을 받지 않고 전체 데이터의 중심을 잘 나타낼 수 있는 중앙값을 이용해 결측값을 대체하는 방식을 선택하였다.

이러한 접근방식을 통해, 모델 학습 과정에서 정보의 손실을 최소화하고, 공격그룹을 더욱 정확하게 식별할 수 있는 기반을 마련하고자 하였다.

IV. Meta-path 임베딩 분석 제안 방법

4.1 Meta-path 임베딩 분석 전체 프레임워크

Fig. 5는 별도의 AI 모델 학습 없이 사전에 정의된 4개의 meta-path를 사용하여 노드 임베딩 결과를 시각화하고 분석하기 위한 전체 과정을 보여준다. 제안된 방법은 두 가지 주요 영역으로 나뉜다. 첫 번째 영역에서는 악성코드, 공격 기술, 공격그룹이라는 세 가지 유형의 노드로 구성된 이기종 그래프 데이터를 생성하고, 두 번째 영역에서는 이 그래프를 기반으로 4개의 meta-path를 설정하고, metapath2vec을 적용하여 노드 임베딩을 수행한다. 이후 이 임베딩 결과를 시각화하여 분석 및 검증하고자 한다. 이러한 방식을 통해, 우리는 단순한 분석을 넘어서 공격그룹 간의 상호 관계와 기술적 방법론 간의 유사성을 탐색하고자 한다.

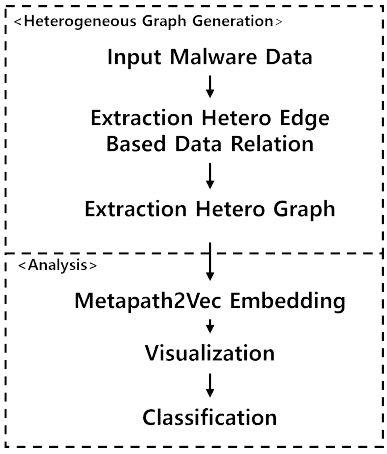

Fig. 5. Proposed Framework(2)

4.2 Meta-path 구성

Fig. 6은 악성코드인 소프트웨어를 중심으로 한 이기종 그래프에서 세 가지 노드 타입을 포함하는 4가지 meta-path를 나타낸다. meta-path는 네트워크 내의 복잡한 관계를 나타내며, 구성에 따라 네트워크의 구조적 특성과 상호작용을 다양한 관점에서 분석할 수 있게 한다. 첫 번째 Case 1은 소프트웨어와 공격 기술 간의 연결을 통해 소프트웨어가 공통으로 사용하는 기술적 특성을 파악할 수 있다. 두 번째, Case 2는 소프트웨어와 공격그룹 간의 관계를 통해 공격그룹의 행동양식과 선호하는 소프트웨어 유형을 이해할 수 있다. 세 번째, Case 3은 소프트웨어와 연관된 공격그룹, 해당 그룹이 사용하는 기술, 그리고 다른 공격그룹과의 관계를 통해 기술적인 유사성 분석이 가능하다. 마지막으로 Case 4는 소프트웨어, 공격 기술, 공격그룹 간의 복잡한 상호작용을 드러낼 수 있다. 이러한 meta-path를 통해, 우리는 네트워크 내 세 가지 주요 노드 유형 간의 다양한 관계를 깊이 있게 이해하고 분석하고자 한다.

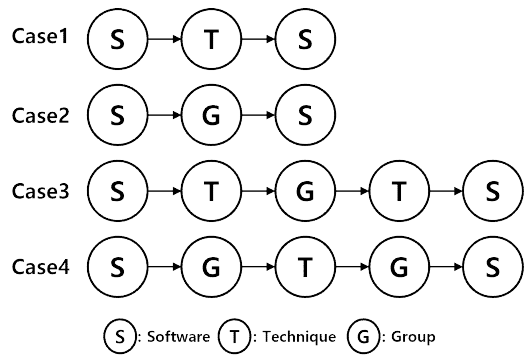

Fig. 6. Meta-path Design

V. 실험 및 결과

5.1 공격그룹 탐지성능

5.1.1 데이터 셋

본 논문에서는 KISA(한국인터넷진흥원)에서 수집 및 제공한, 10개 공격그룹을 라벨로 하는 총 4,755개의 악성코드 데이터를 활용하고자 한다. 공격그룹별 이름과 악성코드 수는 Table. 1과 같다. 이때 원본 데이터의 총 특징 수는 604개이며, 이 데이터를 학습용으로 사용하기 위해 아래와 같은 특징 전처리 작업을 수행하였다. 우선 숫자 형태의 특징(Numeric features)은 StandardScaler를 통해 정규화하였고, 이진 특징(Binary features)은 참/거짓 값을 0 또는 1로 변환하였다. 문자열 값(String feature)을 가진 특징은 n-gram 기법을 적용하여 처리하였으며, 범주형 특징(Categorical features)은 고유한 값들을 숫자로 표현하였다. API 함수 사용 여부를 나타내는 문자열 이진 특징(String binary features)은 0 또는 1로 변환하였고, 악성코드의 백신 탐지명을 나타내는 AVClass 특징은 고윳값을 숫자로 변환하는 처리를 하였다. 이 과정을 거친 전체 특징 중에서도 탐지율에 크게 기여하는 특징들을 선별하여 최종적으로 181개를 선택하였다. 선택한 특징 중 '4-gram' 및 '10-gram' 특징은 악성코드 바이트 열을 연속적인 n 바이트의 시퀀스로 분해한 특정 패턴사용 횟수로 데이터의 특정 부분을 특성화하는 데 사용될 수 있는 정보이다. 또한 'entropy' 특징은 데이터의 복잡성을 측정하여 악성 가능성을 의미하는 척도로 많이 사용되는 특징 중 하나이다.

Table 1. Dataset

Attack Group	Label	Number of data
APT28	0	752
Lazarus	1	1,856
Twisted Panda	2	19
KONNI	3	164
APT29	4	520
APT41	5	235
TA505	6	316
Turla	7	450
Kimsuky	8	265
APT37	9	178
Total	**10**	**4,755**

모델 학습을 위해서는 데이터 세트는 8:2의 비율로 학습데이터와 테스트 데이터를 분할 하였다. 학습 데이터를 기반으로 그래프를 구축하기 위해, 악성코드가 사용한 공격 기술의 ID 값을 포함하는 데이터만을 선별하여 사용하고자 하였다. 따라서 전체 4,755개 악성코드 데이터 중 MITRE ATT&CK의 Tactics에서 12개에 Tactics에 해당하는 공격 기술 ID를 활용하고자 하였다. 이 기준에 해당하는 최소 하나 이상의 공격 기술 ID 값을 포함하는 데이터 총 2,520개를 활용하고자 하였다. 이때 사용한 데이터에 공격 기술에 대한 결측값이 많아 사용할 수 있는 데이터의 수가 적다는 한계점이 있었다. 그럼에도 본 연구에서는 metapath2vec 같은 복잡한 네트워크 분석 기법을 사용함으로써, 작은 데이터 셋에서도 유의미한 패턴과 상관관계를 추출할 수 있음을 보여주고자 하였다. 이후 이 데이터를 기반으로 소프트웨어 노드인 악성코드와 공격 기술 노드를 갖는 그래프를 구축하였다.

5.1.2 주요 결과

악성코드와 공격 기술 간의 토폴로지 정보를 얻기 위해 Fig. 6의 meta-path 중 A를 사용하여 metapath2vec 노드 임베딩을 수행하였다. 이 임베딩 과정을 통해 얻은 벡터값들은 고차원 공간에 위치하게 되며, 이를 직관적으로 이해하기 위해서는 적절한 차원 축소 기법이 필요하다. 이에 따라 t-SNE 방법을 적용하여 임베딩 벡터들을 2차원으로 축소하였고 이후 시각화 도구를 활용하여 2차원 평면에 표현하였다. 그 결과 유사 공격 기술이 사용되어 연관성이 높은 소프트웨어끼리 그룹화되는 것을 확인할 수 있었고, 다수의 동일한 소프트웨어를 사용한 공격 기술도 서로 군집이 형성됨을 확인할 수 있었다. 이러한 토폴로지 정보는 소프트웨어 간, 그리고 공격 기술 간의 관계를 해석하는 데 중요한 역할을 수행할 수 있게 된다.

이후 공격그룹 탐지성능 비교를 위한 데이터는 두 가지로 첫 번째는 악성코드 정보만 존재하는 데이터이고 두 번째는 기존 악성코드 데이터에 Metapath2Vec 수행 후 2차원으로 표현된 토폴로

지 정보를 결합한 데이터이다. 이 두 데이터를 활용해서 10개 공격그룹 대상 식별성능을 비교하고자 하였고 그 결과 Table. 2와 Table. 3과 같다. Table. 2는 토폴로지 정보를 활용하지 않고 악성코드 데이터만을 사용하여 공격그룹 탐지를 수행한 결과로 86%의 정확도를 보였고 Table. 3은 토폴로지 정보를 반영한 데이터의 성능 결과로 약 88%로 탐지율에서 향상이 나타났음을 볼 수 있다. 이를 통해 토폴로지 정보의 사용이 공격그룹 탐지의 정확도와 효율성 향상에 중요한 역할을 수행할 수 있다는 가능성을 볼 수 있다. Label 0을 보면 node 특징만 활용하였을 때 precision(0.85), recall(0.78), f1-score(0.82)로 측정되었는데 이후 토폴로지 정보를 반영한 성능에서는 precision(0.92), recall(0.80), f1-score(0.86) 으로 향상됨을 보였다. 특히 Label 5에서는 node feature만 사용하였을 때 성능이 precision(0.74), recall(0.71), f1-score(0.73)로 측정되었었는데 토폴로지 정보를 반영 후 precision(0.89), recall(0.86), f1-score(0.87)로 큰 성능 폭으로 향상됐음을 알 수 있다. 이러한 성능향상의 이유는 Label 5 공격그룹이 사용한 소프트웨어와 공격 기술 관계에서 유의미한 특징이 보였기에 탐지에서 향상이 나타났다고 볼 수 있다. 이때 macro avg와 weighted avg에서도 토폴로지 특징을 추가로 활용한 모델에서 성능 향상이 나타났다. macro avg는 모든 클래스에 대한 성능을 동등하게 고려하며, 이는 불균형한 클래스 분포를 가진 데이터 셋에서 중요한 지표로 사용된다. 악성코드 데이터만을 사용했을 때의 macro avg가 precision(0.85), recall(0.79), f1-score(0.81)였던 것에 비해, 토폴로지 정보를 반영한 데이터를 사용했을 때는 macro avg가 precision(0.88), recall(0.80), f1-score(0.83)으로 상승하였다. 다음 weighted avg는 각 클래스의 샘플 수를 고려한 평균으로, 모델의 전반적인 정확도에 대한 더 정확한 평가를 제공한다. 이때 악성코드 데이터만 사용했을 때는 precision(0.86), recall(0.86), f1-score(0.86) 이었던 반면, 토폴로지 정보를 추가하였을 때 precision(0.88), recall(0.88), f1-score(0.88)으로 개선되었다.

Table 2. Node feature result

XGBoost Accuracy		0.8611	
attack group	precision	recall	f1-score
0	0.85	0.78	0.82
1	0.84	0.95	0.89
2	1.00	1.00	1.00
3	0.79	0.79	0.79
4	0.93	0.88	0.90
5	**0.74**	**0.71**	**0.73**
6	0.83	0.83	0.83
7	0.96	0.91	0.93
8	0.93	0.54	0.68
9	0.62	0.45	0.53
macro avg	0.85	0.79	0.81
weighted avg	0.86	0.86	0.86

Table 3. Node + Topology feature result

XGBoost Accuracy		0.8789	
attack group	precision	recall	f1-score
0	0.92	0.80	0.86
1	0.84	0.97	0.90
2	1.00	1.00	1.00
3	0.79	0.79	0.79
4	0.94	0.89	0.92
5	**0.89**	**0.86**	**0.87**
6	0.88	0.78	0.82
7	0.96	0.91	0.93
8	0.94	0.58	0.71
9	0.62	0.45	0.53
macro avg	0.88	0.80	0.83
weighted avg	0.88	0.88	0.88

이는 해당 공격그룹의 악성코드와 공격 기술의 특성이 특히 다양하고 복잡하여, 임베딩 과정에서 추출된 토폴로지 정보만으로는 충분히 특징을 잘 표현하지 못했기 때문으로 분석하였다. 또한, 이러한 metapath2vec과 같은 임베딩 기법을 통해 산출된 토폴로지 정보의 활용이 모든 경우에 동일한 효과를 보장하지 않을 수 있음을 보여주며, 공격그룹별 특성에 따른 맞춤형 접근방식이 중요하다는 점을 보여준다.

결론적으로, 단순 노드 특징만 존재하는 데이터에서는 구조적인 관점이 포함된 포괄적인 정보가 담겨있지 않았지만, metapath2vec의 노드 임베딩을 통해 산출된 데이터에는 구조적인 특성이 나타나기

때문에 모델 성능에서 시너지 효과가 나타나는 것으로 판단할 수 있다.

5.2 IoC 연관관계 분석

5.2.1 데이터 셋

이 장은 이기종 그래프 데이터에 4가지 meta-path를 적용하여, 사전에 설정한 목표에 따라 적절한 임베딩이 이루어지는지 확인하고자 한다. 이를 위해 사용한 그래프 데이터는 공격그룹, 공격기술, 소프트웨어의 세 가지 노드 유형과 이들 사이의 다양한 관계를 나타내는 엣지로 구성된다. 엣지 유형에는 '공격그룹-공격 기술(GT)', '소프트웨어-공격 기술(ST)', '공격그룹-소프트웨어(GS)'의 세 가지가 있으며, 이는 각각 그룹과 공격 기술, 소프트웨어와 공격 기술, 그룹과 소프트웨어 간의 관계를 지칭한다. 이러한 데이터에 4개의 meta-path를 적용한 결과를 통해, 실제 네트워크 내에서의 상호작용을 얼마나 정확하게 반영하는지 각 케이스 별 평가를 보이고자 한다.

5.2.2 Meta-path 별 주요 결과

먼저, Case 1은 동일한 공격 기술을 사용하는 소프트웨어가 서로 유사하게 임베딩되도록 설계된 meta-path이다. 이 meta-path의 목적은 같은 공격 기술을 사용하는 서로 다른 소프트웨어, 또는 같은 소프트웨어를 사용하는 서로 다른 공격 기술 간에 유사성을 부여하는 것이다. Fig. 7은 이러한 관계를 시각화한 것으로, 노란색은 소프트웨어를 푸른색은 공격 기술을 나타내며 각 노드 유형을 같은 임베딩

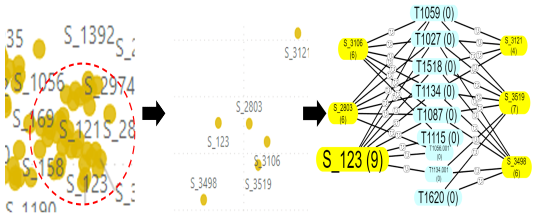

Fig. 8. (Case 1) Software node graph analysis

공간에 나타내었다. 이 결과 내 공격 기술과 연결되지 않은 소프트웨어는 왼쪽에 크게 클러스터가 형성됨을 볼 수 있었다.

더욱 구체적인 분석을 위해, 소프트웨어와 공격 기술 노드를 분리하여 분석을 수행하였다. 이 과정에서, 푸른색 박스로 표시한 부분에 해당하는 유사하게 임베딩된 소프트웨어의 일부를 대상으로 실제 데이터에서 검증한 결과, 동일한 공격 기술을 사용하는 다양한 소프트웨어 간의 실제 연결이 있었음을 Fig. 8에서 확인할 수 있었다. 이러한 결과는 사전에 설정하였던 Case1의 meta-path 의도대로 임베딩이 의미 있게 되었음을 알 수 있다. 반대로 Fig. 8에서 보이는 9개의 공격 기술도 유사하게 임베딩 됐음을 보였고 이러한 결과를 통해, 기존 평면 데이터에서는 알 수 없었지만, 만약 soft_1, soft_2가 있을 때 이러한 meta-path를 통해소 이 두 소프트웨어 사이에 유사한 특징이 존재한다는 새로운 분석관점을 제시할 수 있으며, 두 소프트웨어가 같이 쓰일 확률이 높다는 의미임으로 이 데이터를 가지고 모델 학습을 한다면 분류에 유용하게 사용할 수 있게 된다.

Case 2는 동일한 공격그룹에 의해 사용된 소프트웨어들 사이의 유사성을 탐색하기 위해 설계된 meta-path이다. 이 meta-path는 같은 공격그룹

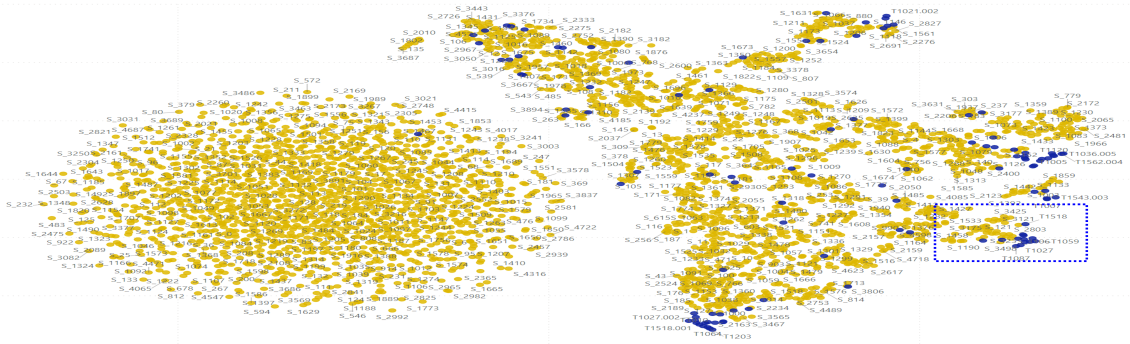

Fig. 7. (Case1) All node type embedding

에 의해 사용된 서로 다른 소프트웨어, 혹은 서로 다른 공격그룹에서 사용된 동일한 소프트웨어 간의 유사성을 포착하려는 목적이 있다.

이를 통해, 각 소프트웨어와 공격그룹 사이의 연결성이 임베딩 과정에서 반영되어, 유사한 소프트웨어 또는 공격그룹이 임베딩 공간 내에서 서로 가까이 위치하게 된다. Fig. 9는 이러한 접근방식의 결과를 시각화한 것으로, 소프트웨어와 공격그룹 노드 유형이 동일한 임베딩 공간 내에서 어떻게 잘 클러스터링 되는지를 나타낸다. 10개의 클러스터별 색상은 본 실험에서 사용한 10개의 공격그룹을 나타낸다. 이러한 결과는 소프트웨어를 대상으로 한 공격그룹 분류의 유효성을 검증한다. 현재 설계된 meta-path에 따라 소프트웨어 노드에 공격그룹 정보가 포함되어, 각 소프트웨어가 특정 공격그룹과 1대1로 매칭되었기에 의미있게 임베딩된 것을 확인할 수 있다.

Case 3은 동일한 공격 기술을 사용하는 공격그룹과 해당 그룹이 사용한 소프트웨어들 간의 유사성을 탐색하기 위해 설계된 meta-path이다. 이 meta-path는 서로 다른 공격그룹이 동일한 공격 기술을 사용하거나, 하나의 공격그룹이 여러 공격 기술을 사용하는 경우, 그리고 동일한 공격 기술을 사용한 다양한 공격그룹의 소프트웨어들이 서로 유사하게 임베딩되도록 하는 것을 목적으로 한다. Fig. 10은 이러한 관계를 시각화한 것으로, 노란색은 소프트웨어를 나타내면 붉은색은 공격그룹, 푸른색은 공격 기술 노드로 서로 다른 노드 유형이 동일한 임베딩 공간 내에서 어떻게 표현되는지 나타낸다. 시각화 결과에서 10개의 공격그룹 정보가 포함된 소프트웨어들이 의미 있게 클러스터링 됐음을 볼 수 있다. 이 meta-path 디자인에 따른 임베딩 결과의 유효성을 실제 데이터를 통해 검증한 결과, 공통의 공격 기술을 사용하는 서로 다른 공격그룹들과 그 그룹들이 사용한 다양한 소프트웨어 간의 연결성을 확인할 수 있었다. 추가적으로, 공격 기술을 중심으로 한 분석을 수행한 결과, 공통으로 사용된 여러 공격 기술을 기반으로 하는 서로 다른 공격그룹들이 연결되어 있음을 발견하였으며, 각 공격그룹에 속한 다양한 소프트웨어들이 실제로 어떻게 연계되어 있는지를 확인할 수 있었다.

Case 4는 앞서 언급된 meta-path와 유사한 접근을 취하면서도, 반대 방향의 경로를 탐색한다. 이는 동일한 공격그룹에 의해 사용된 서로 다른 공격 기술들과 그 공격 기술들에 의해 사용된 소프트웨어들 사이의 유사성을 포착하기 위해 설계되었다. 이 meta-path의 목적은 서로 다른 공격 기술을 사용하는 동일한 공격그룹, 또는 동일한 공격 기술을 사용하는 서로 다른 공격그룹, 그리고 동일한 공격그룹에 의해 사용된 서로 다른 공격 기술에 의해 사용된 소프트웨어들이 서로 유사해지도록 하는 것이다. 이

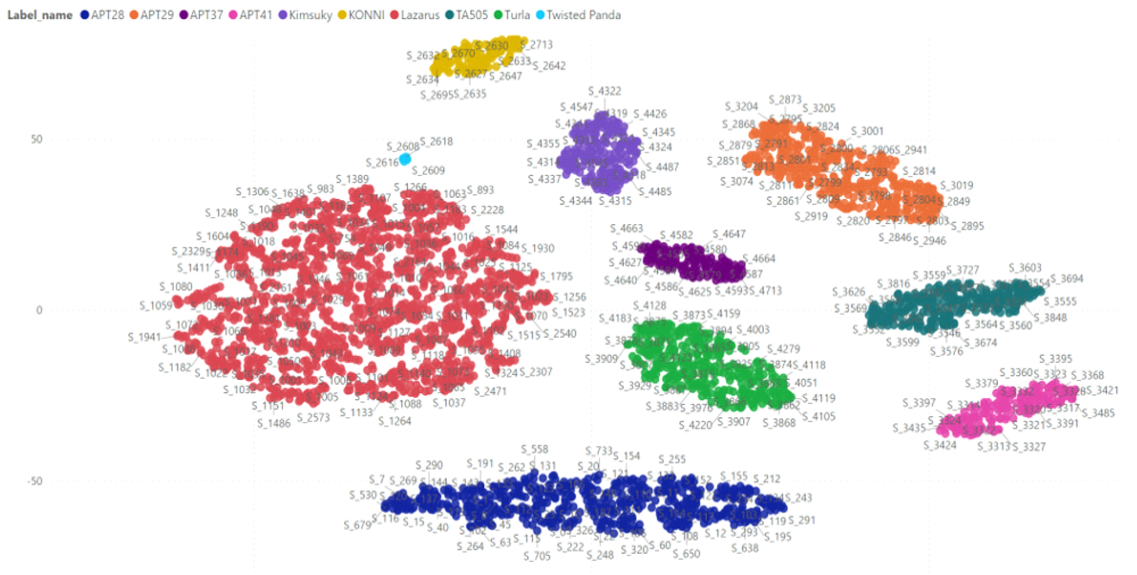

Fig. 9. (Case 2) Only software embedding

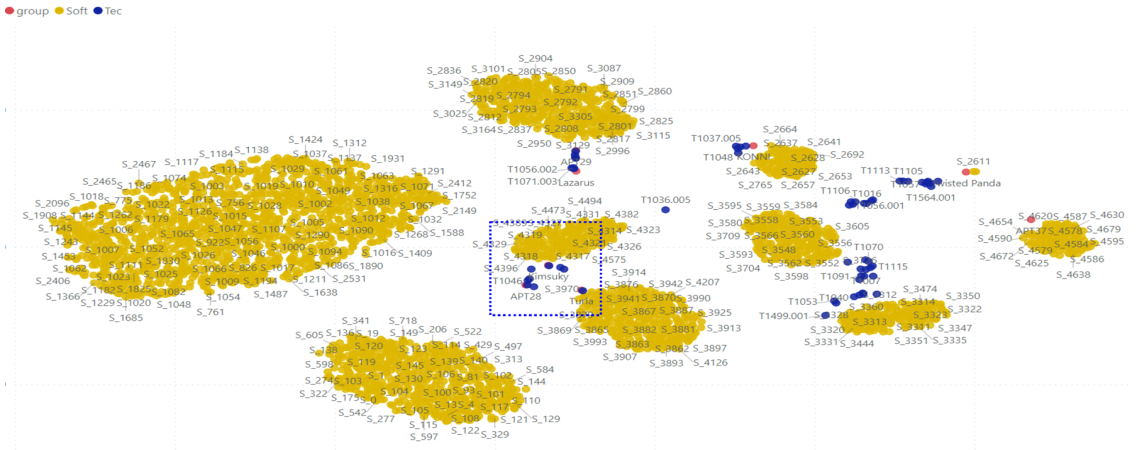

Fig. 10. (Case 3) All node type embedding

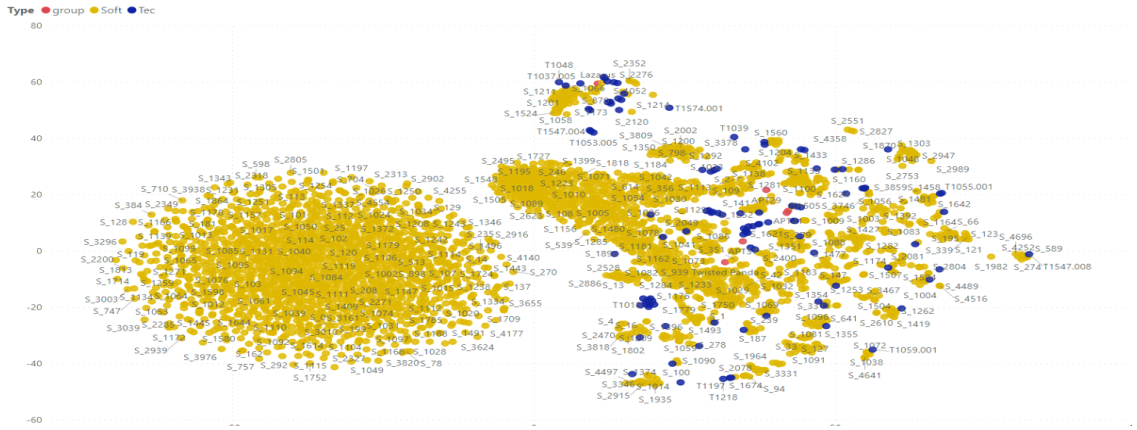

Fig. 11. (Case4) All node type embedding

러한 분석 목적에 따라 설계된 meta-path를 통해 소프트웨어, 공격그룹, 그리고 사용된 공격 기술 간의 상호작용을 효과적으로 분석할 수 있었으며, 이 과정에서 meta-path의 실질적인 적용 가능성과 분석적 가치를 확인할 수 있었다.

5.2.3 종합해석 관련

이 장에서는 실험을 통해 얻은 주요 결과들을 종합적으로 검토하고 평가하고자 한다.

본 연구에서 관찰된 성능향상의 폭이 상대적으로 제한적이었던 주된 요인은 두 가지로 분석할 수 있다. 첫 번째 요인은 실험에 사용된 데이터의 양이 전반적으로 적고, 특히 공격 기술 관련 데이터가 부족했다는 점이다. 이는 분석 가능한 데이터의 범위를 축소해 모델이 학습할 수 있는 정보의 양이 제한되었기 때문이다. 두 번째 요인은 구조적 정보가 공격그룹 분류에 있어 노이즈로 작용했을 가능성을 고려할 수 있다. 이는 소프트웨어 노드의 엣지가 공격 기술의 사용 여부에 따라 결정되어, 소프트웨어의 본질적인 특성보다는 공격 기술과의 관계에 더 중점을 둔 결과로 나타났을 수 있다.

그런데도 일정 수준의 성능향상이 관찰된 것은, 엣지 설계 방향성과 임베딩 값을 활용이 부분적으로는 성공적이었음을 의미한다. 특히, 공격그룹 분류보다는 악성코드 관련 라벨을 소프트웨어 노드 임베딩에 적용할 경우, 성능향상 가능성이 더 커질 것으로 기대된다. 따라서 이 연구를 향후 더 개선함으로써 보다 향상된 분류 성능을 달성하고자 한다.

이기종 그래프에서 설정한 네 가지

meta-path(Case 1, Case 2, Case 3, Case 4)를 통한 metapath2vec 분석 결과는 목적에 부합하는 유의미한 정보를 제공한다. 이 데이터 클러스터링은 예상치 못한 유사성을 드러내며 새로운 관점을 제시한다. meta-path의 범위가 넓어질수록, 더 다양한 연결과 상호작용을 포함하여 분석을 광범위하게 확장할 수 있다. 이는 네트워크 내 다양한 관계와 패턴을 포괄적으로 이해하는 데 도움을 주는 사실을 알 수 있다.

각 meta-path 분석을 통해 식별된 소프트웨어, 공격 기술, 공격그룹 간의 관계는 사이버 보안 상황에서 실제 적용 사례로 연결될 수 있다. 이는 공격유형의 구체적 식별에 도움을 주며, 보안 분석가에게 탐지 전략을 수립하는 데 중요한 정보를 제공한다. 이러한 분석과 결과를 통해, 우리는 구조적 정보를 활용한 분석 방법의 유효성을 평가하고, 향후 연구 및 실제 적용을 위한 기반을 마련하고자 한다. 각 케이스별 임베딩 결과는 실제 사이버 보안 상황에서의 적용 가능성을 보여주었으며, 특정 공격그룹이나 기술과 관련된 소프트웨어를 더 정밀하게 식별하는 데 중요한 역할을 한다는 사실을 알 수 있다.

VI. 결 론

본 연구는 사이버 보안 환경 내에서 공격그룹 식별의 중요성에 주목하고, metapath2vec 기법을 활용하여 공격그룹, 소프트웨어, 공격 기술 간의 상호작용을 분석하는 새로운 방법을 제안하였다. 이 접근법은 공격 패턴을 정확히 식별하고, 공격그룹의 행동을 예측하며, 새로운 보안 위협에 대응하기 위한 전략을 마련하는 데 필수적인 역할을 할 수 있음을 보이고자 하였다. 따라서 실제 사례 분석을 통해 도출된 임베딩 결과는 이 연구가 실제 보안 문제해결에 적용 가능함을 시사한다.

향후 취약점, IP 주소, 도메인, CVE와 같은 추가 엔티티를 포함해 분석의 범위를 확장할 수 있으며. 이를 통해 특정 취약점을 활용한 공격 기술이나 CVE와 관련된 공격그룹을 더욱 심층적으로 분석할 수 있게 된다. 이러한 확장된 분석 방법은 사이버 위협의 지속적인 진화에 대응하면서 더 정밀하고 효율적인 위협 인식 및 대응 전략의 수립에 기여할 것으로 기대된다.

References

[1] H. Neuschmied, M. Stojanović, B. Hofer-Schmitz, K. Božić, and U. Kleb, "Apt-attack detection based on multi-stage autoencoders," Applied Sciences, vol. 12, no. 13, pp. 6816, July 2022.

[2] T. Zoppi, A. Ceccarelli, and A. Bondavalli, "Unsupervised algorithms to detect zero-day attacks: Strategy and application," IEEE Access, vol. 9, pp. 90603-90615, July 2021.

[3] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," IEEE International Joint Conference on Neural Networks, vol. 2, pp. 729-734, July 2005.

[4] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," arXiv preprint arXiv:1312.6203, Dec. 2013.

[5] F. Xia, K. Sun, S. Yu, A. Aziz, L. Wan, S. Pan, and H. Liu, "Graph learning: A survey," IEEE Transactions on Artificial Intelligence, vol. 2, no. 2, pp. 109-127, April 2021.

[6] Y. Zhu, W. Xu, J. Zhang, Q. Liu, S. Wu, and L. Wang, "Deep graph structure learning for robust representations: A survey," arXiv preprint arXiv:2103.03036, March 2021.

[7] M. Xu, "Understanding graph embedding methods and their applications," SIAM Review, vol. 63, no. 4, pp. 825-853, Dec. 2021.

[8] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery

and data mining, pp. 701-710, August 2014.

〔9〕 A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 855-864, August 2016.

〔10〕 Y. Dong, N.V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 135-144, August 2017.

〔11〕 M. Guan, X. Cai, J. Shang, F. Hao, D. Liu, X. Jiao, and W. Ni, "HMSG: Heterogeneous graph neural network based on metapath subgraph learning," Knowledge-Based Systems, vol. 279, pp. 110930, Feb. 2023.

〔12〕 J. Zhao, Q. Yan, X. Liu, B. Li, and G. Zuo, "Cyber threat intelligence modeling based on heterogeneous graph convolutional network," Proceedings of the 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020), pp. 241-256, Oct. 2020.

〔13〕 Y. Gao, X. Li, H. Peng, B. Fang, and S.Y. Philip, "Hincti: A cyber threat intelligence modeling and identification system based on heterogeneous information network," IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 2, pp. 708-722, Feb. 2022.

〈저자소개〉

이 예 은 (Ye-eun Lee) 학생회원
2019년 3월~현재: 호서대학교 컴퓨터공학부 학석사과정
〈관심분야〉 악성코드 분석, 침입 탐지, 이상징후 탐지, 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