이 일 구 교수 지도

석사학위 청구논문

악성 암호화 트래픽 분류를 위한 주성분 분석 기반 적응형 특징 선택 기법

2023

성신여자대학교 대학원

미래융합기술공학과

이 유 림

악성 암호화 트래픽 분류를 위한 주성분 분석 기반 적응형 특징 선택 기법

이 일 구 교수 지도

이 논문을 석사학위 논문으로 제출함

2022년 11월

성신여자대학교 대학원
미래융합기술공학과
이 유 림

인 준 서

이유림의 석사학위 논문으로 인준함

2022년 11월

심사위원장 박 새 롬 (서명 또는 인)

심 사 위 원 이 일 구 (서명 또는 인)

심 사 위 원 김 성 민 (서명 또는 인)

성신여자대학교 대학원

논문개요

최근 전 산업에서 데이터 보호가 중요해지면서, 안전한 데이터의 송수신을 위하여 암호 기술을 필수적으로 적용하게 되어, 암호화 트래픽의 비중이 급증하고 있다. 그러나 패킷 내용을 확인할 수 없으므로 Network Intrusion Detection System (NIDS)와 같은 탐지 시스템으로 페이로드를 분류할 수 없다는 한계점을 악용한 공격이 등장하였다. 이러한 문제를 해결하기 위해 Deep Packet Inspection (DPI)과 인공지능을 활용한 대응 연구가 수행되었으나, 프라이버시 문제와 실시간 탐지가 불가능한 한계로 인하여 여전히 실제 산업에서의 사용은 어려운 실정이다. 또한, 학습 모델을 교란시키기 위하여 학습 데이터에 의도적으로 노이즈를 주입하는 중독 공격으로 인하여 네트워크 트래픽 학습 및 분류 난이도가 증가하고 있다. 본 연구에서는 악성 암호화 트래픽을 효율적으로 분류하기 위하여 Principal Component Analysis (PCA) 기반 최적의 차원 정보와 탐지율 그래프의 기울기 정보를 활용한 특징 선택 기법을 제안한다. 노이즈 수준이 달라지는 환경에서 1차원부터 27차원까지 데이터의 차원을 축소한 뒤, 탐지율의 변화를 확인한다. 이때 탐지율 그래프의 기울기가 가파른 지점에서 설명 분산 점수가 높은 특징을 활용하거나 차원을 축소한 특징을 선택적으로 추가하여 적용하는 adaptive Feature Selection based F1-score Gradient (adaptive FSFG)를 제안한다. 실험 결과에 따르면 PCA를 사용했을 때보다 탐지율 측면에서 24.74% 개선되었고, 학습 시간은 35% 개선되었다. 그리고 특징 선택 기법을 수행하지 않았을 때보다 32.82%의 탐지율이 개선되었고, 학습 시간은 48% 개선되었다. 추가로, 제안방안을 정적으로 적용하였을 때와 동적으로 적용하였을 때를 비교하여 제안하는 adaptive FSFG가 약 8.67%만큼 탐지율이 개선되는 효과가 있음을 증명하였다.

목 차

표 차 례

그림 차례

Ⅰ. 서 론

빅데이터의 활용도가 높아짐에 따라 개인정보를 포함한 중요 데이터 보호에 대한 요구가 증가하고 있다. 안전한 데이터의 송수신을 위하여 암호화 기술이 도입되었으며, SSL/TLS 기술을 활용한 암호화 트래픽의 수가 증가하고 있다 [1]. Gartner에 따르면, 2019년 기준 80% 이상의 트래픽이 암호화되어있으며, 구글은 2021년 기준 세계적으로 많이 사용되는 사이트 100개 중 97%가 암호화 기술을 사용하고 있다고 보고했다 [2].

암호화 트래픽의 양이 증가함에 따라, 암호화 트래픽의 데이터 식별 불가능성을 악용한 지능형 사이버 범죄가 등장하였다 [3]. VMWare에 따르면, 2020년에 발생한 사이버 공격의 70%가 암호화 기술을 활용하고 있으며, 점차 악성 암호화 트래픽 공격이 더욱 증가할 것으로 예상된다 [3]. 암호화 트래픽은 평문의 내용을 확인하기 어려우므로 종래의 네트워크 침입 탐지 시스템으로 탐지하기 어렵다는 문제가 있다. 이러한 암호화 기술을 활용한 악성 트래픽 공격에 대응하기 위하여, 심층 패킷 분석 기술과 인공지능을 활용한 연구들이 수행되었다 [4, 5, 6, 7]. 그러나 심층 패킷 분석 기술은 암호화 데이터의 패킷 내용을 확인하는 방식이므로 프라이버시 침해 요소가 포함되어 있다 [4]. 또한, 인공지능 기술은 여러 산업 분야에서 활용되고 있음에도 불구하고, 데이터 오버헤드 문제로 인해 주로 오프라인에서 탐지하고, 실시간 탐지가 어려운 실정이다 [8].

인공지능 기술이 여러 분야에 사용되면서, 학습데이터에 의도적으로 노이즈를 추가하여 학습 모델을 교란시키는 중독 공격이 등장했다 [9, 10]. 중독 공격은 간단한 방법으로 수행될 수 있으나, 인공지능 모델 학습에 직접적인 영향을 주기 때문에 피해 파급력이 매우 높다. 노이즈가 추가된 데이터를 학습할 경우 분류 모델의 성능이 저하될 수 있으며, 잘못 학습된 모델을 활용

한 자율주행차를 운행한다면 차량의 속도나 경로를 악의적인 방향으로 유도하여 사람의 안전에 피해를 입힐 수 있다 [11, 12, 13]. 또한, 데이터의 중앙 집중형 저장 방식으로 인한 오버헤드 문제를 피하기 위하여 등장한 연합 학습에 적용될 경우, 각 단말이 악성 모델을 업데이트 하도록 유도할 수 있다 [14]. 따라서 중독 공격에 대한 영향을 최소화함으로써 피해를 줄이고, 악성 행위를 정확하게 탐지할 수 있는 방안이 연구되어야 한다.

본 연구에서는 악성 암호화 트래픽을 효율적으로 분류하기 위하여 차원 축소 기법인 주성분 분석 (PCA, Principal Component Analysis) 기반의 특징 선택 기법을 제안한다. 또한, 학습 모델을 교란시키기 위해 학습데이터에 의도적으로 추가한 노이즈를 고려하여 노이즈의 수준에 따른 적응형 데이터 처리 기법을 제안한다. 실험에서는 DNS 쿼리를 암호화한 DNS over HTTPS(DoH) 트래픽으로 구성된 CIRA-CIC-DoHBrw-2020 데이터 셋을 사용하였다 [15]. 전체 데이터 셋 중 정상 DoH과 악성 DoH를 분류하기 위한 2 계층 데이터 셋을 활용하였으며, 일부 데이터에서 값이 비어있는 특징은 삭제한 뒤 사용하였다. 전처리를 거친 데이터 셋을 PCA를 사용하여 차원을 축소한 뒤, 축소된 차원에 따른 인공지능 분류 모델의 탐지율 변화를 확인하였다. 이때 탐지율 그래프의 기울기를 분석하여 탐지율에 영향을 주는 특징을 선별하였다. 추가로, 노이즈가 증가하는 환경에서 동일한 실험을 진행하여, 노이즈 수준에 따라 다른 특징 선택 기법을 적용하는 Adaptive Feature Selection based F1 Score Gradient (Adaptive FSFG)을 실험하여 탐지율, 메모리 사용량, 학습 시간을 측정하였다.

논문의 주요 기여점은 다음과 같다.

1) 악성 암호화 트래픽을 효율적으로 분류하기 위하여 차원 축소와 탐지율 그래프 기울기 기반 특징 선택 기법을 제안한다.

2) 탐지율 그래프의 기울기가 가파른 지점의 특징과 탐지율 기반 최

적의 차원을 활용하여 노이즈 수준에 따른 적응형 특징 선택 기법을 제안하였으며, 학습 시간, 정확도, 메모리 효율성을 개선하였다.

3) 제안하는 기법을 통해 악성 암호화 트래픽을 복호화하지 않고도 탐지할 수 있으며, 중독 공격으로 인해 노이즈가 포함된 데이터를 정확하고 효율적으로 분류할 수 있다.

논문은 다음과 같이 구성된다. 2장에서 관련 연구에 대해 분석하고, 3장에서는 제안하는 탐지율 기울기 기반 적응형 특징 선택 기법에 관해 설명한다. 4장에서는 실험 방법에 관해 설명하고 결과를 분석한다. 마지막으로, 5장에서 결론을 맺는다.

Ⅱ. 관련 연구

본 장에서는 악성 암호화 트래픽 분류와 관련된 선행연구를 분석했다. 기존 악성 암호화 트래픽 분류 연구는 전통적인 인공지능 알고리즘을 활용하여 단순 분류한 연구가 대부분이다. 악성 암호화 트래픽은 악성 특징이 암호화 되어 있어 학습이 어렵고, 중요도가 낮은 특징들이 함께 학습되면서 분류 모델을 교란시킬 수 있다. 그러나 기존 연구들은 암호화 트래픽의 특성을 고려하지 않았으며, 추가적인 보안 조치나 환경적 특성으로 인해 추가된 노이즈를 고려하지 않았다. 본 연구는 기여도가 높은 특징을 선별한 후 이를 활용하여 악성 암호화 트래픽을 분류 및 탐지하는 기법을 제안한다.

1. 암호화 트래픽 개요

전자 상거래, 인터넷 뱅킹 등이 활성화되면서 온라인 상에서 공유되는 중요 정보의 양이 증가하고 있다. 사용자의 정보를 보호하기 위하여, 종래 웹 서비스 프로토콜인 HTTP (Hypertext Transfer Protocol)에 암호화 기술을 적용한 HTTPS가 등장하였다. HTTPS는 송수신 데이터를 보호하기 위한 프로토콜로, SSL(Secure Socket Layer)/TLS(Transport Layer Security)를 활용하여 보안성 있는 통신을 보장한다[16]. HTTPS의 핸드셰이크 과정은 FIGURE 1과 같다.

클라이언트는 지원하는 암호화 알고리즘, 난수, SNI (Server Name Indication) 필드, 세션 ID를 포함하여 “ClientHello” 메시지를 서버에 전송한다. 서버는 클라이언트가 보낸 암호화 알고리즘 중 사용할 암호화 알고리즘을 선택한 뒤, 해당 암호화 알고리즘, 세션 ID, 난수, 인증서, SNI를 포함하여 “ServerHello” 메시지로 응답한다.

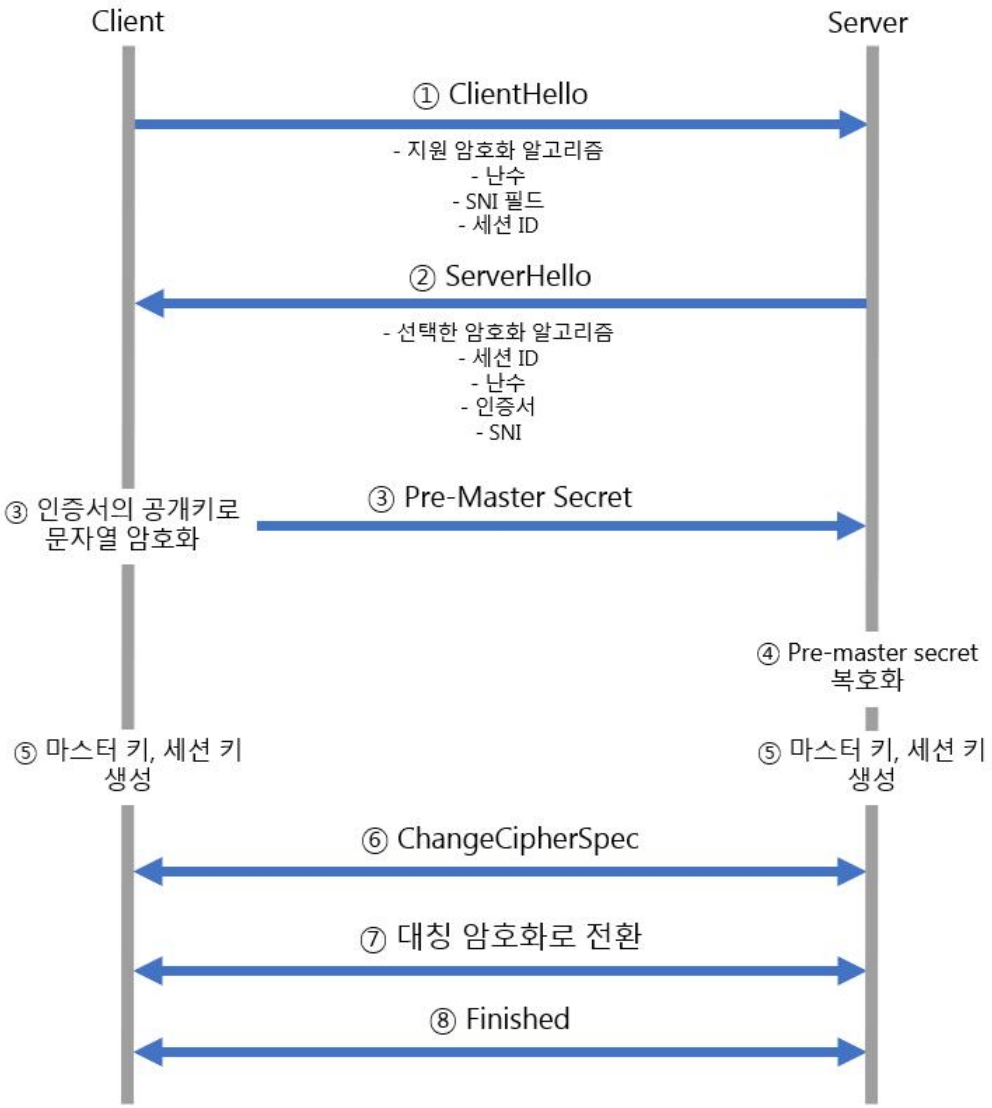

FIGURE 1. Handshake of HTTPS ([16] 그림을 재구성)

클라이언트는 수신한 인증서의 공개키를 활용하여 임의의 문자열인 “Pre-Master Secret”을 암호화하여 서버에게 전송한다. 본 과정은 비대칭 키 암호화의 일종으로, 클라이언트가 서버로부터 수신한 인증서를 확인하는 과정이다. 이후, 서버는 자신의 개인 키를 사용하여 “Pre-Master Secret”을 복호화하여 확인한다. 그리고 통신에 참여 중인 클라이언트와 서버는 대칭 암

호화에 사용될 마스터 키와 세션 키를 생성한다. 마지막으로, “ChangeCipherSpec” 메시지를 교환함으로써 통신 암호화 방식을 대칭 암호화로 전환하고, “Finished” 메시지를 교환하는 것으로 TLS 핸드셰이크가 종료된다.

2. 인공지능 기반 악성 트래픽 분류

과거의 악성 암호화 트래픽 분류 연구에서는 심층 패킷 분석 기술을 활용하여 암호화 트래픽을 복호화한 뒤, 악성 여부를 분석하였다. 그러나 심층 패킷 분석 기술을 활용한 탐지 방식은 암호화된 데이터의 평문에 접근할 수 있는 방식이므로 프라이버시 문제가 발생한다 [3]. 따라서 최근 연구들에서는 인공지능 분류 모델을 활용하여 악성 암호화 트래픽을 탐지하고자 하는 시도를 많이 하고 있다. TABLE I은 인공지능을 활용한 악성 암호화 트래픽 분류 관련 선행연구를 정리한 표이다.

TABLE I
Method and limitation of related works using artificial intelligent

Papers	Methods	Limitation
Matthew Behnkle et al [17]	특징 선택 기법을 적용한 뒤, 머신러닝을 활용하여 순차 순방향 선택 방식으로 악성 DoH 트래픽 분류	• 새로운 기법을 제안하는 연구가 아니라, 종래 방안들을 조합한 뒤 최적의 머신러닝 분류 모델을 파악하는 연구임 • 노이즈를 고려하지 않았음
K. Li et al [18]	4-tuple을 활용하여 트래픽에 대한 추가적인 정보를 얻은 뒤, 딥러닝 모델을 사용하여 악성 암호화 트래픽 분류	• 데이터 전처리가 복잡하지만, 전처리 시간을 고려하지 않았음 • 트래픽의 암호화 여부를 식별하는 과정이 없어 실제 환경에서 활용되기 어려움 • 노이즈를 고려하지 않았음
Zhang X et al [19]	5-tuple을 활용하여 암호화 트래픽을 이미지화한 뒤 분류하는	• 실시간 탐지가 불가능함 • 노이즈를 고려하지 않았음

본 연구와 동일하게 CIRA-CIC-DoHBrw-2020 데이터 셋을 사용한 연구에서는 다양한 머신러닝 모델을 비교하여 각 레이어의 악성 트래픽 분류에 효과적인 모델을 확인하였다 [17]. 해당 연구에서는 카이제곱검정, 피어슨 상관계수를 활용하여 특징을 선택한 뒤, 의사결정 나무, 랜덤 포레스트, LightGBM, XGBoost 모델을 사용하여 악성 DoH 트래픽을 분류하였다. 그리고 그 중 성능이 가장 좋은 모델에 순차 순방향 선택 기법을 적용하여 악성 암호화 트래픽을 재분류하였다. 그러나 해당 연구에서는 악성 암호화 트래픽을 효과적으로 분류하기 위한 새로운 방안을 제안하고 있지 않으며, 전통적인 머신러닝 알고리즘의 분류 성능을 개선하기 위하여 등장한 종래 기법들을 조합하였다. 또한, 메모리 효율성 측면에서의 효과를 보여주고 있지 않으며, 실제 환경의 노이즈를 고려하지 않았다는 한계가 있다. 또 다른 연구에서는 SSL/TLS 세션을 통해 통신할 경우 다른 세션과는 달리 클라이언트가 해당 세션에서 새로운 포트를 할당받는 특징을 활용하여 악성 암호화 트래픽을 분류하였다 [18]. 해당 연구에서는 같은 source IP, destination IP, destination port, transport layer protocol을 가지는 세션 그룹을 4-tuple로 표현하며, 4-tuple은 어떤 클라이언트가 하나의 서버를 하나의 포트로 통신하는 모든 flow를 포함하는 것으로 정의한다. 실험에서는 4-tuple에서 세션이 재개되지 않은 flow의 바이트 스트림과 애플리케이션 데이터 레이어의 사이즈 시퀀스 특징을 활용하였다. 그리고 딥러닝 분류 모델을 활용하여 악성 암호화 트래픽을 분류하였다. 해당 연구에서는 트래픽 데이터 중 초기 바이트만을 선정하고, 4-tuple을 추출한 뒤 필요한 특징을 분석하는 과정이 필요한데, 전체 과정에 대한 소요시간을 측정하지 않았다. 또한, 입력된 트래픽에 대한 암호

화 여부를 확인하지 않는 것으로 보아, 모든 트래픽이 암호화 트래픽이라는 가정에 따라 진행된 것으로 보인다. 그러나 실제 환경에서는 암호화 트래픽과 일반 트래픽이 혼재되어 들어오기 때문에, 제안하는 내용을 활용하기 어렵다. 마지막으로, 해당 연구에서도 노이즈를 고려하지 않았다는 한계가 있다. 위 연구와 유사하게 5-tuple 정보를 활용한 연구에서는, 암호화 트래픽을 이미지화한 뒤 분류하는 DF-IDS를 제안하였다 [19]. 5-tuple은 source IP, source port, destination IP, destination port, transport layer protocol 정보로 구성된다. DF-IDS는 SSL/TLS 특징에 따라 원본 트래픽 데이터를 쪼갠 뒤, 해당 데이터에서 SNI 정보를 제거하였다. 그리고 2차원 이미지로 재가공한 뒤, 인공지능 알고리즘을 활용하여 분류하였다. 그러나 트래픽 데이터를 이미지화한 후 분류하는 방식은 실시간 탐지 방식으로 활용되기 어렵다는 한계가 있다. 또한, DF-IDS 대비 RAM 사용량이 현저히 낮은 모델이 존재하며, 노이즈를 고려하지 않았다는 한계가 있다.

3. 탐지율 개선을 위한 특징 선택

인공지능 분류 알고리즘의 성능 개선을 위하여 파라미터 최적화, 데이터 처리 등 다양한 기법이 활용되고 있다. 특히, 데이터 처리 오버헤드를 줄이고 탐지율을 개선하는 방안으로 특징 선택 기법이 많이 사용되고 있다. TABLE II는 효율적인 악성 트래픽 분류를 위하여 특징 선택 기법을 사용한 선행연구를 정리한 표이다.

TABLE II
Method and limitation of related works using feature selection

Papers	Methods	Limitation
Onur Barut et al [20]	4가지 특징 선택 기법을 사용하여 메타데이터와 TLS 특징을 추출한 뒤, 성능이 우수한 2가지 특징 선택 기법을 활용하여 악성 암호화 트래픽 분류	• 종래 특징 선택 기법 비교에 가까움 • 노이즈를 고려하지 않았음
Muhammad Shafiq et al [21]	두 가지 특징 선택 기법에서 공통적으로 선택된 특징을 활용하여 악성 IoT 트래픽 분류	• 특징 선택 기법의 효과를 확인하기 위한 것이 아니라, 적합한 분류 모델을 파악하기 위한 성능 평가를 수행하였음 • 노이즈를 고려하지 않았음
Mohammad reza MontazeriS	TLS 세그먼테이션과 IP 단편화 프로세스에서 흩어진 애플리케이션	• 라벨에 영향을 미치는 특징을 선별하는 것이 아니라, 특성에 맞는 특징을 추출하는 방식이므로

hatoori et al [15]	트래픽을 찾아 결합한 뒤, 악성 암호화 트래픽 분류	다양한 애플리케이션에 적용되기 어려움 • 여러 패킷을 합친 뒤, 합친 결과를 바탕으로 탐지를 수행하는 방식으로, 탐지 시간이 오래 걸릴 것으로 예상됨 • 노이즈를 고려하지 않았음

종래 특징 선택 기법 중 효과적인 기법을 분석한 뒤, 이를 활용하여 악성 암호화 트래픽을 분류했던 연구에서는 머신러닝 모델 중 랜덤 포레스트, KNN(K-nearest neighbor) 알고리즘에 포함된 특징 선택 기법과 PCA, 상관관계를 사용하였다 [20]. 그리고 이 중 메타데이터와 TLS 특징을 선택하는데 효과적인 두 가지 특징 선택 기법을 활용하여 악성 암호화 트래픽 분류를 수행하였다. 그러나 본 선행연구에서는 종래 특징 선택 기법을 활용해 성능을 개선하는 방안을 제안하기보다는 특징 선택 기법을 비교하는 것에 그쳤다. 또한, 머신러닝 및 딥러닝으로 분류를 수행할 때, 노이즈를 고려하지 않았다는 한계가 있다. 또 다른 연구에서는 두 가지 특징 선택 기법을 활용하여, 두 기법에서 공통적으로 선택한 특징을 활용하여 악성 IoT 트래픽을 분류하였다 [21]. 본 선행연구에서는 종래 기법인 bijective soft set technique에 제안하는 CorrACC 함께 활용하였다. 제안하는 CorrACC는 피어슨 상관계수를 활용한 특징 선택 기법에 래퍼 방식으로 분류 모델의 정확도 정보를 결합하여 특징을 선택하는 기법이다. 그러나 본 선행연구에서는 각 특징 선택 기법을 개별적으로 사용했을 때와 함께 사용했을 때의 성능 평가를 수행하지 않았기 때문에 특징 선택 기법의 효과를 확인할 수 없다. 또한, CorrACC에서 각 특징의 상관계수와 같은 특징 선택 기법의 구체적인 결과를 보여주고 있지 않다는 한계가 있다. 그리고 노이즈를 고려하고 있지 않았

으므로 현실적인 결과라고 보기 어렵다. 본 연구에서 사용한 CIRA-CIC-DoHBrw-2020 데이터 셋을 사용해 특징 추출 기법을 제안한 연구에서는 여러 패킷을 합친 뒤, 합친 결과를 바탕으로 악성 암호화 트래픽을 분류 및 탐지하였다 [15]. 본 선행연구에서는 TLS 세그먼테이션과 IP 단편화 프로세스에서 흩어진 애플리케이션을 찾아 결합한 뒤, 악성 암호화 트래픽 분류를 수행하였다. 그러나 본 선행연구에서 수행한 특징 추출 기법은 라벨에 영향을 미치는 특징를 선별하는 방식이 아닌, 특성에 맞는 특징을 추출하는 방식이므로 다양한 애플리케이션에 적용되기 어렵다는 한계가 있다. 또한, 여러 패킷을 합치는 데 시간이 오래 걸리고, 노이즈를 고려하지 않았다는 한계가 있다.

4. 노이즈 환경에서의 공격 탐지

인공지능을 활용하여 공격을 탐지할 때, 데이터에 포함된 노이즈를 고려하여 공격을 탐지하기 위한 연구들이 여러 차례 수행되었다. TABLE III 악성 행위를 탐지하기 위하여 노이즈를 처리하는 기법을 제안한 선행연구를 정리한 표이다.

TABLE III
Method and limitation of related works considering noise

Papers	Methods	Limitation
Michal Piskozub et al [22]	Denoising auto-encoder를 사용하여 노이즈를 제거한 뒤 멀웨어 탐지	• 데이터에 의도적으로 추가한 노이즈가 아닌 트래픽 수집 과정에서 추가되는 양성 flow에 포함된 악성 트래픽의 특징을 노이즈로 간주하고 있음 • 여러 파라미터 별 분류 성능은 제시하고 있으나, 선행연구나 종래 방안과의 성능 비교가 없음
Chuanpu Fu et al [23]	주파수 도메인 분석을 사용하여 네트워크 트래픽의 순차 정보를 추출한 뒤 분석하여 공격 탐지	• 노이즈가 많은 현실 세계를 배경으로 하고 있으나, 정상 데이터만으로 클러스터링을 했다고 신뢰하는 것에 기반하여 성능 평가를 수행하였음
Jinchi Huang et al [24]	딥러닝의 과소적합과 과적합의 전이를 활용하여 노이즈가 포함된 라벨 값을	• 실시간으로 공격을 탐지하는 데 활용되기에는 무리가 있음 • 충분히 과소적합과 과적합을 반복하기 위하여 오랜 시간이

3단계의 분류 단계를 통해 멀웨어를 분류하는 연구에서는 이진 분류 단계에서 악성 트래픽의 특징 추출을 위하여 denoising auto-encoder를 사용하였다 [22]. 제안하는 방안에서는 Deep neural network(DNN)을 이진 분류기로 사용하여 트래픽의 악성 여부를 파악한 뒤, 멀웨어의 유형과 패밀리를 분류하였다. 그러나 본 연구에서 활용하는 노이즈는 학습 데이터 공격을 위하여 의도적으로 추가한 노이즈로 특징들에 노이즈가 추가된 반면, 선행연구에서는 감염된 호스트에 의해 생성된 트래픽에 포함되는 노이즈로써 양성 flow에 포함된 악성 트래픽의 특징을 노이즈로 간주하기 때문에 라벨 값에 노이즈가 추가되었다. 따라서 노이즈가 추가되는 방식이 본 연구와 다르다. 또한, 본 선행연구에서는 여러 파라미터 별 분류 성능을 제시하고 있으나, 종래 방안과의 성능 비교는 수행하지 않았다는 한계가 있다. 노이즈를 고려한 또 다른 연구에서는 주파수 도메인을 분석하여 네트워크 트래픽의 순차 정보를 추출한 뒤, 이를 분석하여 공격을 탐지하였다[23]. 제안하는 주파수 도메인 추출 단계는 3단계로 구성되며, 각 패킷 당 시퀀스를 벡터로 인코딩한 뒤, 인코딩된 벡터를 나누고 각 프레임에 이산 푸리에 변환을 수행한다. 마지막으로, 이산 푸리에 변환으로 인해 생성된 주파수 도메인의 모듈러스 로그를 변환한다. 이때, 정상 애플리케이션에서 생성된 다양한 패킷을 공격 트래픽에 추가하였고, 이를 노이즈로 간주하여 회피 공격을 수행하였다. 그러나 본 선행연구는 노이즈가 많은 현실적인 실험 환경을 배경으로 하고 있음에도 불구하고, 정상 데이터 셋만으로 클러스터링을 수행하였다고 신뢰하는 것에 기반하여 성능 평가를 수행하였다는 한계가 있다. 또한, 본 연구에서는 입력 데이터에 노이즈가 추가된 환경인 반면, 선행연구에서는 라벨 값에 노이즈가 있는 형태이므로 본 연구에서 고려하는 노이즈와 주입 형태가 다르며, 탐지 시간

을 측정하지 않았다는 한계가 있다. 노이즈가 포함된 라벨 값을 탐지하기 위하여 딥러닝을 활용한 연구에서는, 딥러닝 모델의 전체 학습 과정 중 복잡하고 어려운 샘플들이 학습되는 후반부에 노이즈가 포함된 라벨이 학습된다는 점을 활용하였다 [24]. 보통 과소적합에서 과적합으로의 상태 전이는 전체 학습 과정 중 한 번만 발생하며, 노이즈가 포함된 라벨이 한번 학습되면 그에 대한 손실이 빠르게 감소되기 때문에 과적합 시기를 확인하기 어렵다. 따라서 과적합에서 과소적합으로의 전이가 반복적으로 발생하도록 유발한 뒤, 언제 노이즈가 포함된 라벨이 과적합되는 지를 파악하였다. 그리고 각 샘플들에 의도적으로 랜덤 노이즈를 추가하여 노이즈가 포함된 라벨을 탐지하였다. 그러나 본 선행연구에서 제안하는 방안은 데이터 셋 전처리 시 수행해야 하는 부분으로, 실시간으로 공격을 탐지하는 데 활용되기에는 무리가 있다. 또한, 충분히 과소적합과 과적합을 반복하기 위하여 오랜 시간이 소요될 것으로 예상되지만, 시간을 측정하지 않았다.

Ⅲ. 탐지율 기울기 기반 적응형 특징 선택 기법

본 절에서는 많은 양의 네트워크 트래픽으로 인한 간섭 및 노이즈의 수준에 따라 악성 암호화 트래픽을 효율적으로 분류하기 위한 적응형 특징 선택 기법을 제안한다. 제안하는 특징 선택 기법에서는 PCA에 기반하여 데이터의 차원을 1차원부터 27차원까지 축소한다. 그리고 각 차원에 따른 탐지율을 확인한 뒤, 탐지율 그래프의 기울기에 영향을 미치는 차원을 파악하여 특징을 선택한다. 제안하는 adaptive feature selection based f1 score gradient (adaptive FSFG)는 탐지율 그래프를 기준으로 특징을 선택하는 Feature selection based f1 score gradient-less feature (FSFG-LF)과 최적의 차원 정보를 활용하여 특징을 추가로 사용하는 Feature selection based f1 score gradient-more feature (FSFG-MF)으로 구분되며, 노이즈의 수준에 따라 두 가지 방식 중 한 가지를 선택할 수 있다.

1. Adaptive FSFG 시스템 구성도

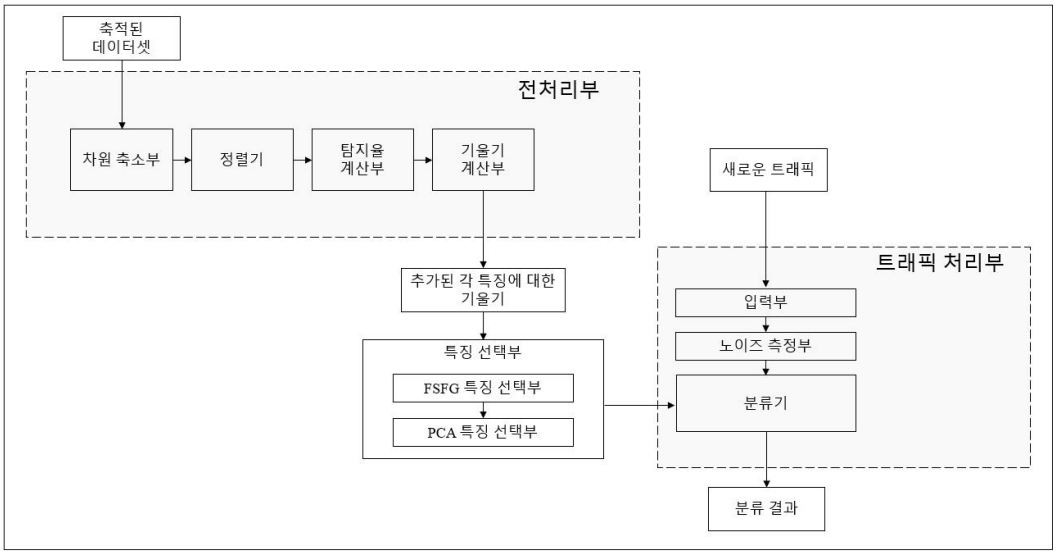

FIGURE 2. System architecture of adaptive FSFG

FIGURE 2는 adaptive FSFG의 시스템 구성도로, 본 시스템은 데이터 셋을 축적할 수 있고 분류를 수행할 수 있는 모든 노드에 적용할 수 있다. 시스템은 새로운 트래픽이 유입되기 전에 서버에서 전처리를 수행하는 전처리부와 새로 유입된 트래픽을 처리하는 트래픽 처리부로 나뉜다. 전처리부에서는 이미 누적된 트래픽 데이터 셋을 활용하며, 차원 축소부는 PCA를 기반으로 데이터 셋의 차원을 축소한다. 정렬기는 각 차원에서의 설명 분산 점수를 바탕으로 특징들을 정렬하고, 탐지율 계산부는 축소한 차원이 증가함에 따라 탐지율을 계산한다. 그리고 기울기 계산부는 축소한 차원이 증가함에 따라 변화한 탐지율인 기울기를 계산한 뒤, 이를 특징 선택부로 전송한다. 특징 선택부는 FSFG 특징 선택부와 PCA 특징 선택부로 나누어진다. FSFG 특징 선택부는 기울기 계산부에서 기울기가 큰 지점의 차원에서 설명 분산 점수가 큰 특징만 선택하고, PCA 특징 선택부는 PCA를 기반으로 축소한 특징의 차원에 따라 특징을 추가 선택한다. 새로운 트래픽이 유입되면, 트래픽 처리부는 이를 입력받은 뒤, 노이즈를 측정한다. 그리고 노이즈가 많으면 FSFG 특

징 선택부만 거친 특징만을 활용하고, 노이즈가 적으면 PCA 특징 선택부까지 모두 거친 특징을 활용한다. 마지막으로 분류기는 선택된 특징들을 활용하여 머신러닝 분류 알고리즘으로 분류를 수행하고 분류 결과를 출력한다.

2. Adaptive FSFG 동작 원리

특징 선택을 위하여 서버에서 사전에 수행되는 데이터 전처리 프로세스는 FIGURE 3과 같다.

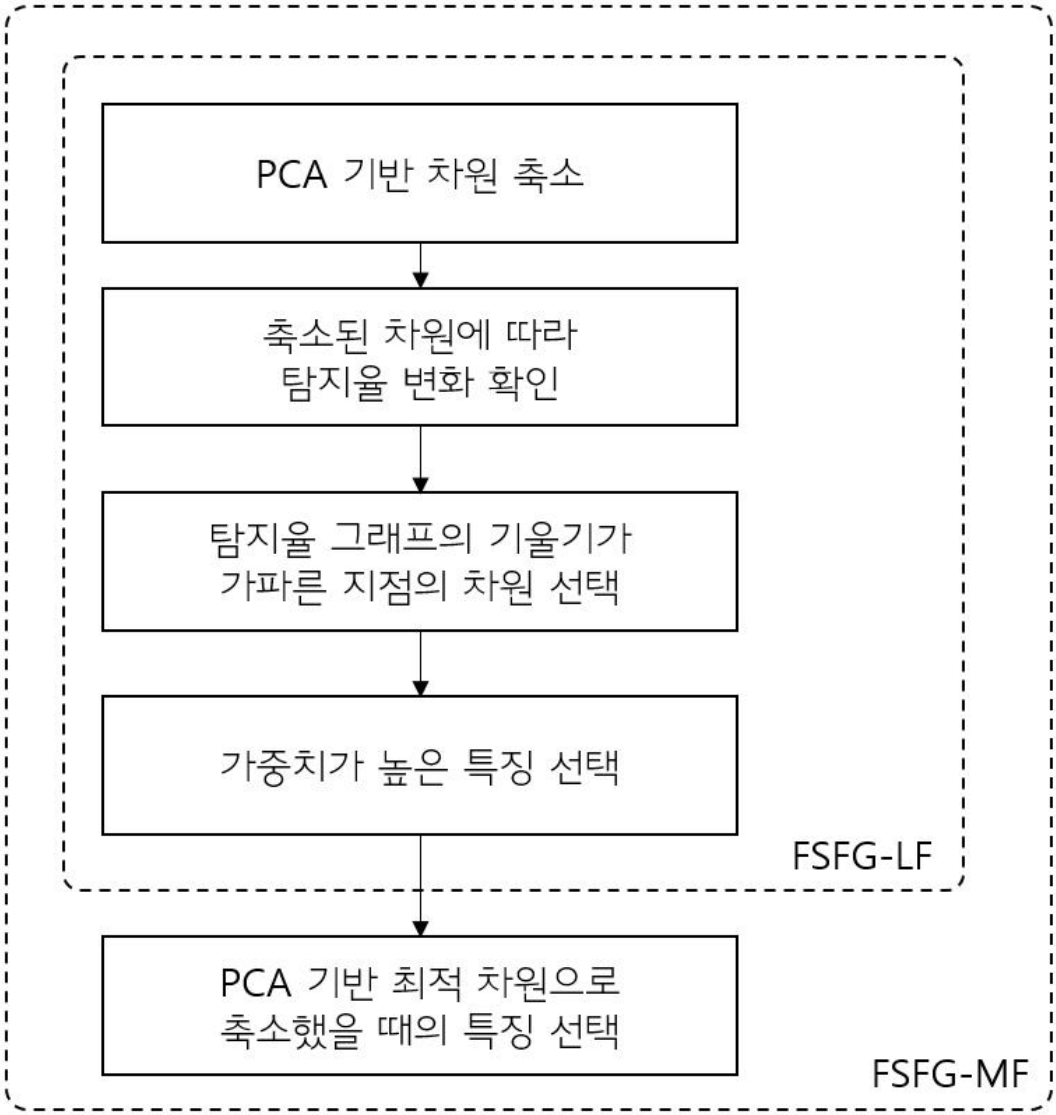

FIGURE 3. Data preprocessing for adaptive FSFG

PCA 알고리즘은 데이터의 분산을 최대화하면서 데이터의 차원을 축소하는 알고리즘으로, 다차원 데이터의 분포를 가장 잘 표현하는 특징을 찾아주기

때문에 특징 선택 기법으로 널리 사용된다. 따라서 PCA 알고리즘을 기반으로 데이터 셋의 차원을 축소한다. 그리고 차원을 최소 차원인 1차원부터 전체 특징 개수만큼 축소하면서, 분류 모델의 탐지율을 확인한다. FSFG-LF는 탐지율 그래프의 기울기 정보를 활용해 특징을 선택하는 기법으로, 탐지율 그래프의 기울기가 크게 변하는 지점에서의 차원을 선택하고, 설명 분산 점수가 높은 특징들을 탐지율에 큰 영향을 주는 특징으로 간주하여 선택한다. FSFG-MF는 축소된 차원에서 특징을 선택하는 기법으로, PCA 기반 최적의 차원과 FSFG-LF에서 선택된 특징들을 함께 사용하는 방식이다. 따라서 FSFG-MF에는 FSFG-LF에서 선택한 특징들이 모두 포함되어 있으며, FSFG-LF보다 많은 특징을 선택한다.

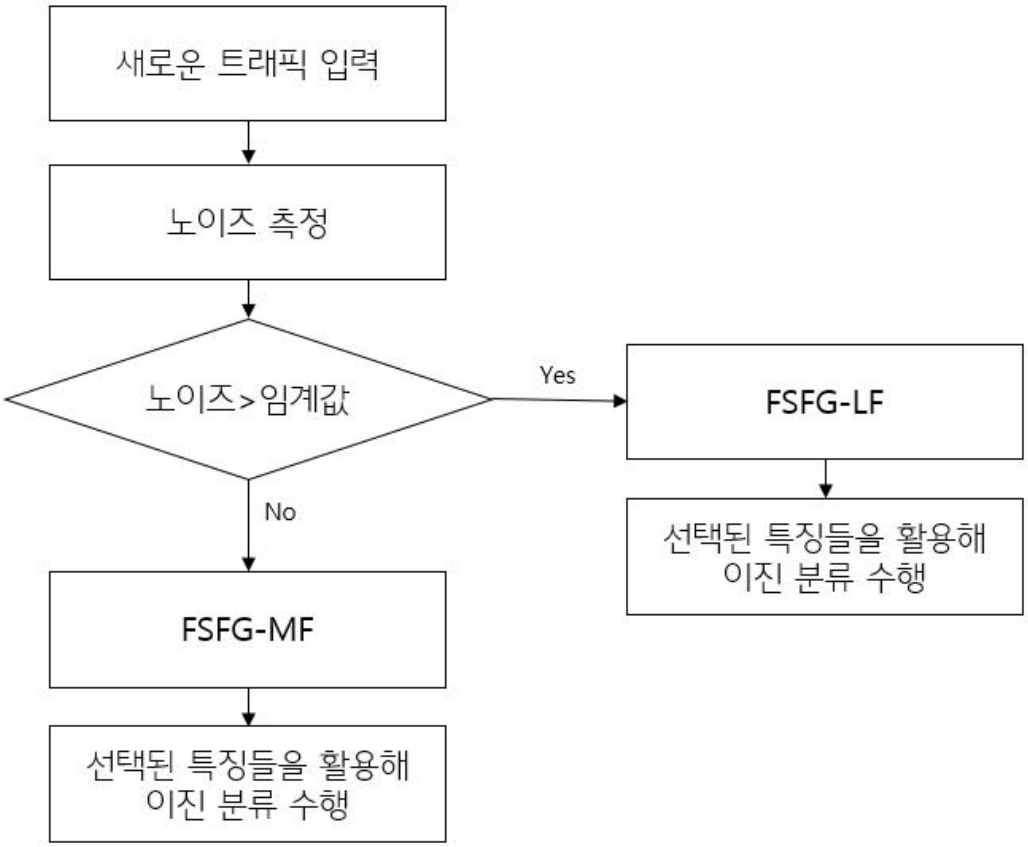

FIGURE 4. Mode selection process of adaptive FSFG

FIGURE 4는 새로운 네트워크 트래픽이 유입되었을 때, 노이즈 수준에 따라 적응형으로 특징 선택 모드를 선택하는 프로세스이다. 먼저, 현재 클라이언트의 데이터 수신 환경의 노이즈 수준을 측정한다. 이때 노이즈는 학습 데이터에 의도적으로 주입한 노이즈이며 데이터 셋 중 특징 값에서 측정한다. 노이즈가 임계값보다 높은 경우에는 FSFG-LF, 그렇지 않은 경우에는 FSFG-MF를 선택한다. 그리고 선택된 특징들을 활용하여 머신러닝 분류 알고리즘으로 이진분류를 수행한다.

본 연구에서는 로지스틱 회귀 모델을 활용하여 이진 분류를 수행하였다. 로지스틱 회귀는 독립적인 특징 값과 라벨 값 사이의 관계 수준에 기반한 회귀 모형으로, 독립된 변수들 간의 관계 수준을 측정하거나 중요도를 파악하는 데 주로 사용된다 [25]. 또한, 다른 머신러닝 모델들 대비 학습 시간이 짧고 정확도 개선이 쉽다는 특성이 있어 공격 분류를 위한 효율적인 솔루션으로 평가받고 있다 [26]. 시그모이드 함수는 로지스틱 회귀의 예측 결과를 특정 라벨에 속하는 지에 대한 조건부 확률로 변환하기 위하여 사용되는 함수로, 특징 x에 대한 시그모이드 함수는 다음과 같다 [26].

$$Sigmoid\,(x) = \frac{1}{1+e^{-x}} \qquad (1)$$

시그모이드 함수를 통해 변환된 예측 값은 0과 1 사이의 값을 가지며, 임계값을 기준으로 임계값 미만인 경우 0, 임계값을 초과할 경우 1로 분류된다 [27].

Ⅳ. 성능 평가

1. 시뮬레이션 환경 설정

1) 데이터 셋

제안하는 특징 선택 기법은 jupyter notedbook에서 Scikit-Learn을 활용하여 파이썬으로 실험하였다. 실험은 Ubuntu 20.04 LTS에서 수행하였으며 RAM 4GB, 하드디스크 60GB, 4개의 core processor를 할당하였다.

악성 암호화 트래픽 분류를 위하여 DNS 쿼리를 암호화한 DoH 트래픽으로 구성된 CIRA-CIC-DoHBrw-2020 데이터 셋을 활용하였다 [15]. 해당 데이터 셋 중 정상 DoH과 악성 DoH를 분류하기 위한 2계층 데이터 셋을 활용하였다. CIRA-CIC-DoHBrw-2020 데이터 셋의 악성 트래픽은 dns2tcp, loadine 등의 DNS 터널링 도구를 사용하여 생성되었으며, firefox, chrome 등에 연결되었던 DoH 트래픽을 수집하였다.

CIRA-CIC-DoHBrw-2020 데이터 셋을 특징 중요도에 따라 내림차순으로 나열한 특징들과 각 특징의 의미는 TABLE IV와 같다.

TABLE IV
Dataset configuration

Features	Description
Duration	Duration
FlowBytesSent	Number of flow bytes sent
FlowSentRate	Rate of flow bytes sent
FlowBytesReceive d	Number of flow bytes received
FlowReceivedRate	Rate of flow bytes received
PacketLengthVaria	Variance of Packet Length

nce	
PacketLengthStan	Standard Deviation of Packet Length
dardDeviation	
PacketLengthMean	Mean Packet Length
PacketLengthMedi	Median Packet Length
an	
PacketLengthMode	Mode Packet Length
PacketLengthSkew	Skew from median Packet Length
FromMedian	
PacketLengthSkew	Skew from mode Packet Length
FromMode	
PacketLengthCoeff	Coefficient of Variation of Packet Length
icientofVariation	
PacketTimeVarian	Variance of Packet Time
ce	
PacketTimeStanda	Standard Deviation of Packet Time
rdDeviation	
PacketTimeMean	Mean Packet Time
PacketTimeMedian	Median Packet Time
PacketTimeMode	Mode Packet Time
PacketTimeSkewF	Skew from median Packet Time
romMedian	
PacketTimeSkewF	Skew from mode Packet Time
romMode	
PacketTimeCoeffic	Coefficient of Variation of Packet Time
ientofVariation	
ResponseTimeTim	Variance of Request/response time difference
eVariance	
ResponseTimeTim	Standard Deviation of Request/response time
eStandardDeviatio	difference
n	
ResponseTimeTim	Mean Request/response time difference

eMean ResponseTimeTim eMode	Mode Request/response time difference
ResponseTimeTim eSkewFromMode	Skew from mode Request/response time difference
ResponseTimeTim eCoefficientVariati on	Coefficient of Variation of Request/response time difference

CIRA-CIC-DoHBrw-2020 데이터 셋에서 악성 트래픽의 공격 종류는 구별되어 있지 않으며 악성 또는 정상으로 구성되어 있으므로 실험에서도 이진분류를 수행하였다. 또한, 데이터 셋을 학습, 시험, 검증 데이터 셋으로 6:2:2 비율로 나누어 사용하였으며, 학습 데이터 셋에 대해 공격과 정상 데이터를 11841개로 동일하게 사용하였다.

2) 평가 방법

실험에서는 탐지율, 학습 시간, 메모리 사용량을 측정하였다. 탐지율은 F1 score를 사용하였으며, F1 score는 다음과 같이 계산할 수 있다.

$$\mathrm{Pr}ecision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1\,score = 2 \times \frac{\mathrm{Pr}ecision \times Recall}{\mathrm{Pr}ecision + Recall} \quad (4)$$

TP (True Positive)는 실제 값이 양성일 때 이를 옳게 분류한 경우이다. FP (False Positive)는 실제 값이 음성일 때 이를 잘못 분류한 경우, FN (False Negative)는 실제 값이 양성일 때 이를 잘못 분류한 경우이다. 실험에서는 F1 score 결과를 백분율로 환산하여 사용하였다.

학습 시간의 경우, 파이썬의 time() 모듈을 사용하여 측정하였다. 그리고 메모리 사용량의 경우, 데이터 셋의 크기를 측정하였다.

3) 탐지율 기울기 기반 특징 선택

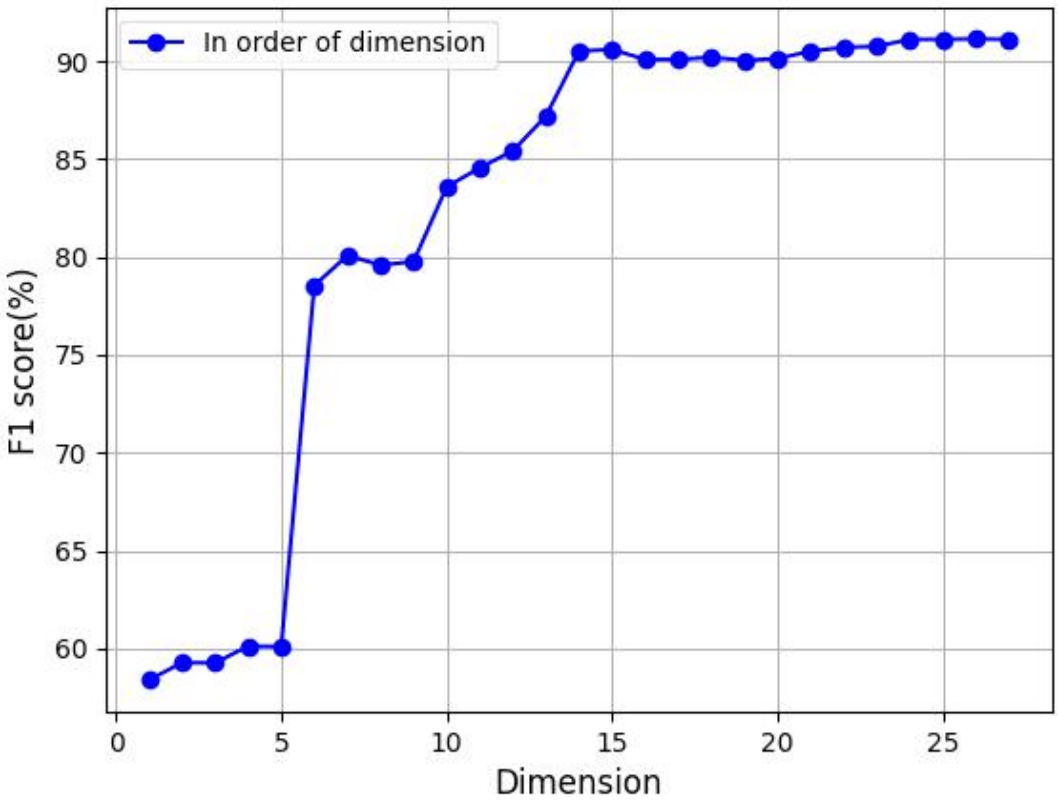

FIGURE 5. F1 score depending on dimension of dataset

FIGURE 5는 데이터의 차원에 따른 탐지율 변화를 확인한 결과이다. PCA를 기반으로 데이터의 차원을 축소했으며, 탐지율 관점에서의 최적의 차원을 확인하기 위하여 실험하였다. 일반적으로 PCA를 활용하여 특징을 선택할 때, 고유 값의 변화가 크게 변화하는 지점에서 차원을 선택한다. 하지만 본 연구에서는 악성 행위를 탐지할 때 효과적인 차원과 특징을 선별 활용하기 위하여 탐지율을 기반으로 차원과 특징을 선택하였다. 실험 결과, 특정 차원에서 탐지율이 크게 증가하였다. 또한, 15차원 이상이 되면서 탐지율이 유지

되었다.

TABLE V은 FIGURE 5에서 표현한 탐지율 그래프의 기울기가 가파르거나 평평한 지점에서의 데이터 차원과 그때의 탐지율 개선 수준을 정리한 표이다.

TABLE V
Dimension when the gradient of the detection rate graph is flat or steep

Dimension of dataset	Gradient of detection rate graph
1	-
2	0.89
6	18.45
7	1.53
8	-0.5
9	0.17
10	3.81
11	0.98
12	0.87
13	1.78
14	3.32
15	0.08
16	-0.51
22	0.21
23	0.05
24	0.36
25	0.0
26	0.01
27	-0.01

TABLE V에 따르면, 탐지율 그래프의 기울기가 가파르거나 평평할 때의 차원은 총 19개이며, 그 중 기울기가 0.9 이상으로 큰 특징들은 6개였다. 따라서 탐지율 그래프의 기울기가 큰 차원을 활용하는 것, TABLE V에 표현한 차원에서 설명 분산 점수가 높은 특징들을 모두 활용하는 것, 축소된 차원을 활용하는 것의 효과를 확인하기 위하여 다양한 특징 조합을 만들었다.

4) 특징 조합 별 성능 평가

특징 선택을 위한 전처리 과정을 설명하기 위하여, TABLE V의 특징들을 조합하여 특징을 4개, 5개, 6개, 11개, 19개를 사용하였을 때 탐지율, 학습 시간, 메모리 사용량을 비교하였다. FIGURE 6은 노이즈 수준에 따른 특징 조합의 탐지율이다.

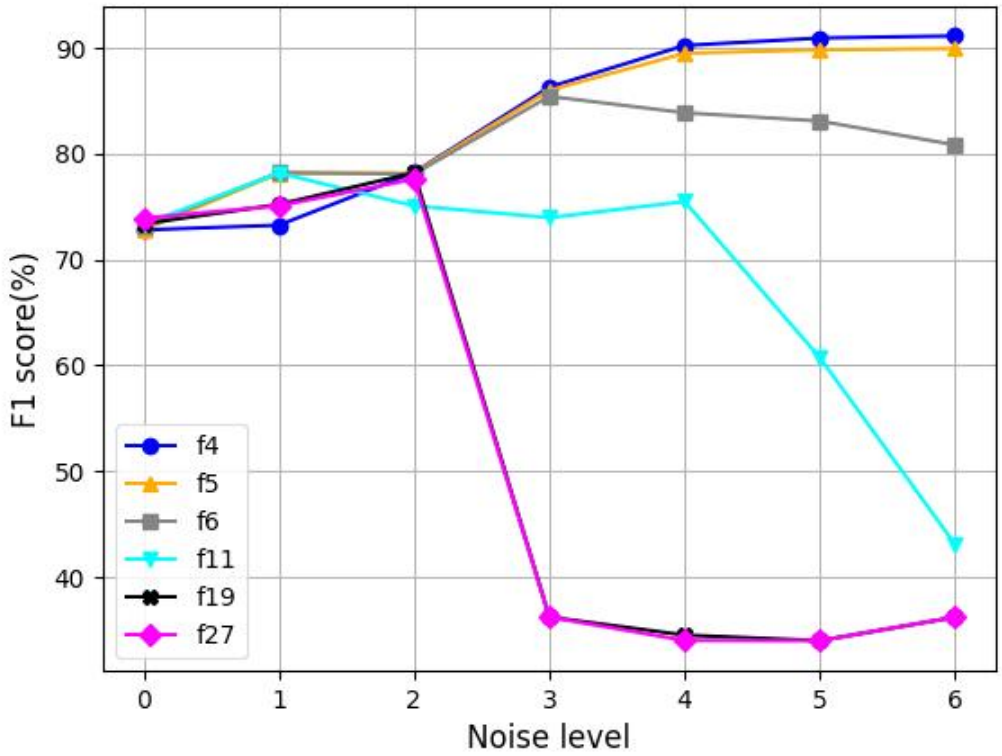

FIGURE 6. F1 score of feature combinations depending on noise level

FIGURE 6의 f4, f5, f6은 탐지율 그래프의 기울기가 가파른 차원에서 설명 분산 점수가 큰 특징 4개, 5개, 6개를 선택한 것이고, f11은 f6에 탐지율이 높은 특징 5개를 추가로 사용한 것이다. f19는 TABLE V에 표현한 특징을 모두 사용한 것이며, f27는 특징 선택 기법을 수행하지 않고 데이터 셋의 모든

특징을 사용한 것이다. 또한, 노이즈 수준의 경우, 각 노이즈 수준이 평균이고 표준편차가 0.1인 랜덤 노이즈를 생성한 뒤 데이터 셋에 추가하였으며, 노이즈가 추가된 데이터 셋을 머신러닝 분류 모델로 분류하였다. 실험 결과에 따르면 노이즈 수준이 2일 때를 기점으로 탐지율 추이가 달라진 것을 확인할 수 있다. 특징을 적게 사용한 f4, f5, f6의 경우, 노이즈 수준이 높아짐에도 불구하고 탐지율이 개선되는 효과를 보였다. 이는 특징을 적게 사용할 때, 노이즈가 가중치와 같은 효과를 주어 특징 활용도가 좋아지는 것으로 보인다. 그에 반해, 특징을 많이 사용한 f11, f19, f27의 경우, 노이즈가 방해요소로 작용하면서 노이즈가 커짐에 따라 탐지율이 급격하게 떨어지는 것을 알 수 있었다. 또한, 더 많은 특징을 사용할수록 큰 노이즈에 더욱 취약한 것으로 보였다. 이는 노이즈와 함께 특징 개수가 늘어나면서 과적합이 더 많이 유도된 것으로 분석된다. FIGURE 6을 통해 본 실험에서의 노이즈 임계치는 2이며, 노이즈가 많을 때는 특징을 4개 사용하였을 때 성능이 가장 우수하고, 노이즈가 적을 때는 특징을 11개 사용하였을 때 성능이 가장 우수한 것을 확인하였다.FIGURE 7은 특징 조합에 따른 학습 시간과 메모리 사용량이다.

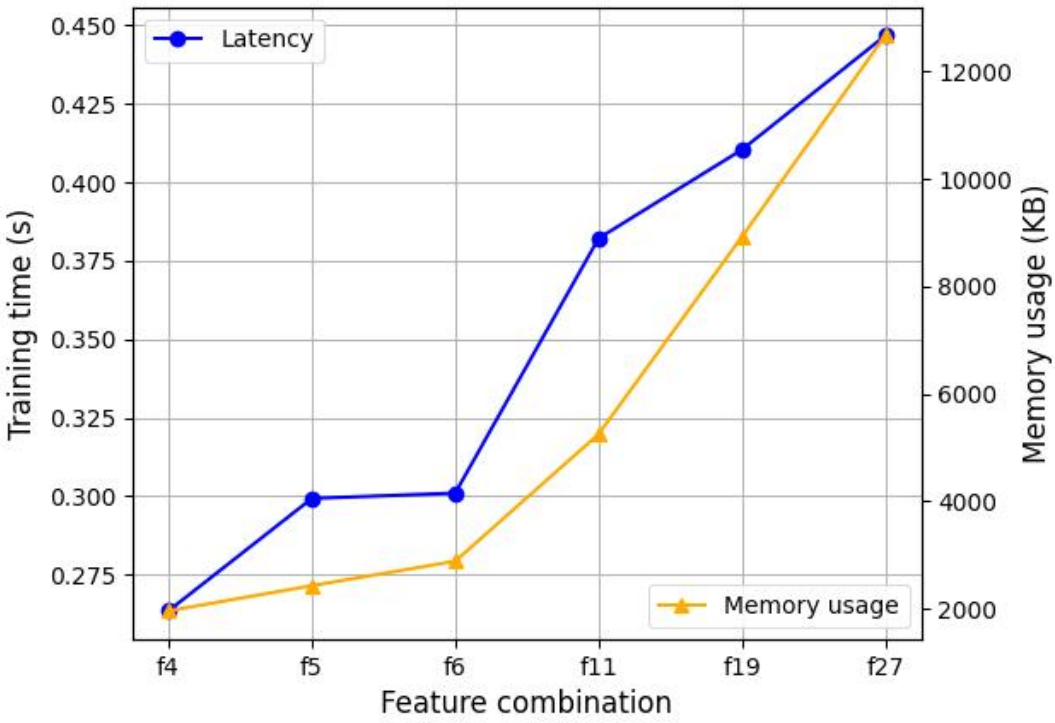

FIGURE 7. Training time and memory usage of feature combinations

학습 시간과 메모리 사용량의 경우에는 노이즈 수준에 따라 크게 변화하지 않았으므로 노이즈가 없는 환경에서 x축에 표현한 특징 개수에 따라 측정하였다. 실험 결과, 많은 양의 특징을 사용할수록 학습 시간이 증가하고 메모리를 많이 사용하였다. 또한, 학습 시간과 메모리 사용량의 증가 추이가 유사한 것을 확인하였다.

2. 실험 결과 및 분석

제안하는 adaptive FSFG의 성능을 탐지율 개선 효과와 효율성 측면에서 평가했다. 머신러닝 분류 모델은 로지스틱 회귀을 사용하였으며, 제안하는 방안은 데이터 셋의 모든 특징을 사용했을 때 (w/o feature selection), PCA를 기반으로 동일한 차원의 특징을 사용하였을 때와 비교하였다. 실험에서는 제안하는 방안과 비교하는 두 가지 대상을 선행연구들에 기반해 모델링 하였다. PCA는 데이터의 차원을 축소한 뒤, 로지스틱 회귀 모델을 사용하여 악성 안드로이드 트래픽 데이터 셋인 CICAndMal2017을 분류하는 연구에 기반하여 모델링 하였다 [28]. W/o feature selection은 다중 머신러닝 분류 모델을 사용하여 VPN/non-VPN을 분류한 연구에서 로지스틱 회귀를 사용한 부분과 유사하게 모델링 하였다 [29].

IV-1장에 따르면, 실험에서 사용한 데이터 셋에 대해, 노이즈가 2 미만으로 적은 환경에서는 특징 11개를 사용하고, 노이즈가 2 이상으로 큰 환경에서는 특징 4개를 사용하는 것이 효과적이었다. 따라서 본 장에서는 최적의 특징 조합을 사용했을 때에 대한 실험 결과를 비교한다.

1) 정적 기법 간 비교

FIGURE 8은 노이즈 수준에 관계없이 제안하는 특징 선택 기법을 정적으로 적용한 FSFG-LF, FSFG-MF와 동일한 수의 특징을 정적으로 사용한 PCA 기법의 탐지율을 비교한 결과이다.

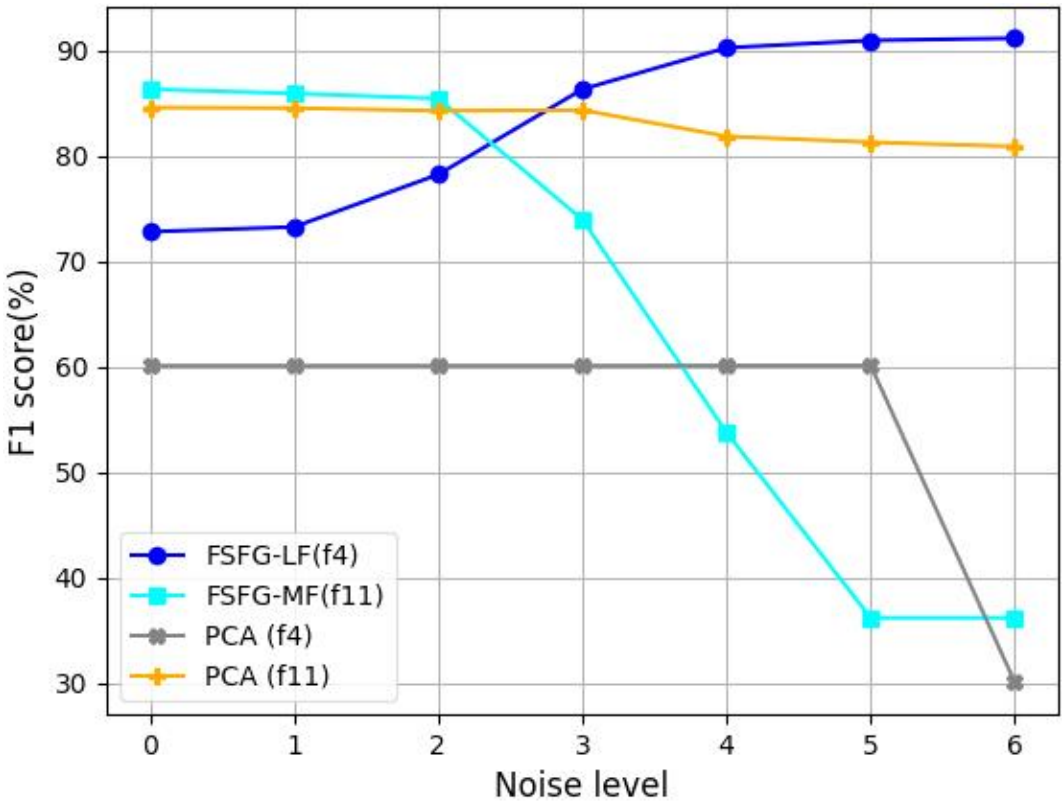

FIGURE 8. F1 score of PCA and proposed method using static feature selection

실험 결과에 따르면 PCA의 경우 특징을 4개 사용했을 때보다 11개를 사용했을 때 탐지 성능이 월등히 개선되었다. FSFG-LF, PCA (f4)는 동일한 개수의 특징을 사용했음에도 불구하고, FSFG-LF에서 탐지율이 약 32.95% 개선되었다. 그리고 FSFG-LF가 PCA (f11)보다 특징을 7개 덜 사용했음에도 불구하고, 탐지율이 더욱 우수한 것을 확인하였다.

2) 동적 기법 간 비교

FIGURE 9는 특징 선택 기법을 수행하지 않았을 때, 종래 방안을 적응형으로 사용하였을 때와 제안하는 adaptive FSFG의 탐지율을 비교한 결과이다.

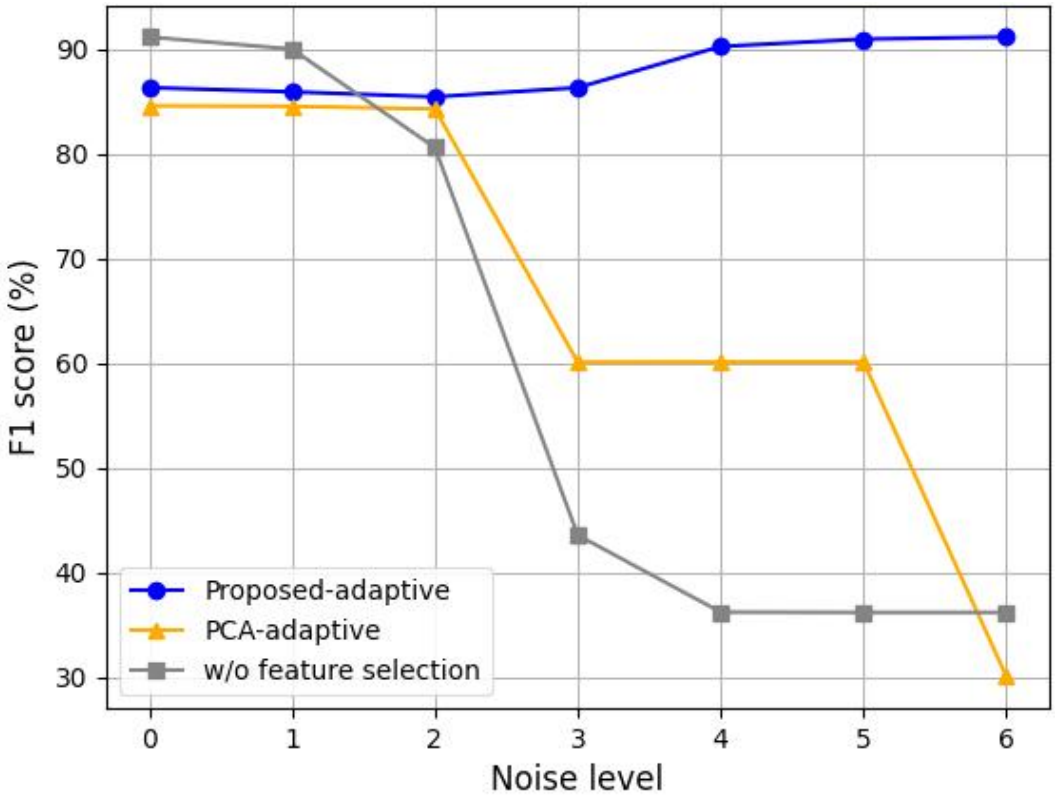

FIGURE 9. F1 score of adaptive FSFG, PCA(adaptive), w/o feature selection

실험 결과에 따르면 제안하는 adaptive FSFG가 노이즈의 수준에 관계없이 가장 안정적이고 우수한 탐지 성능을 보였다. PCA의 평균 탐지율은 66.25%, w/o feature selection의 평균 탐지율은 59.13% 인 반면, adaptive FSFG의 평균 탐지율은 88.03%로 종래 방안 대비 24.74%, 32.82% 만큼 크게 개선된 효과를 보였다. 또한, 종래의 두 가지 방안의 경우 노이즈가 커짐에 따라 탐지율이 급격하게 저하되는 반면에 adaptive FSFG는 노이즈의 수준과 무관하게 우수한 탐지 성능을 보였다.

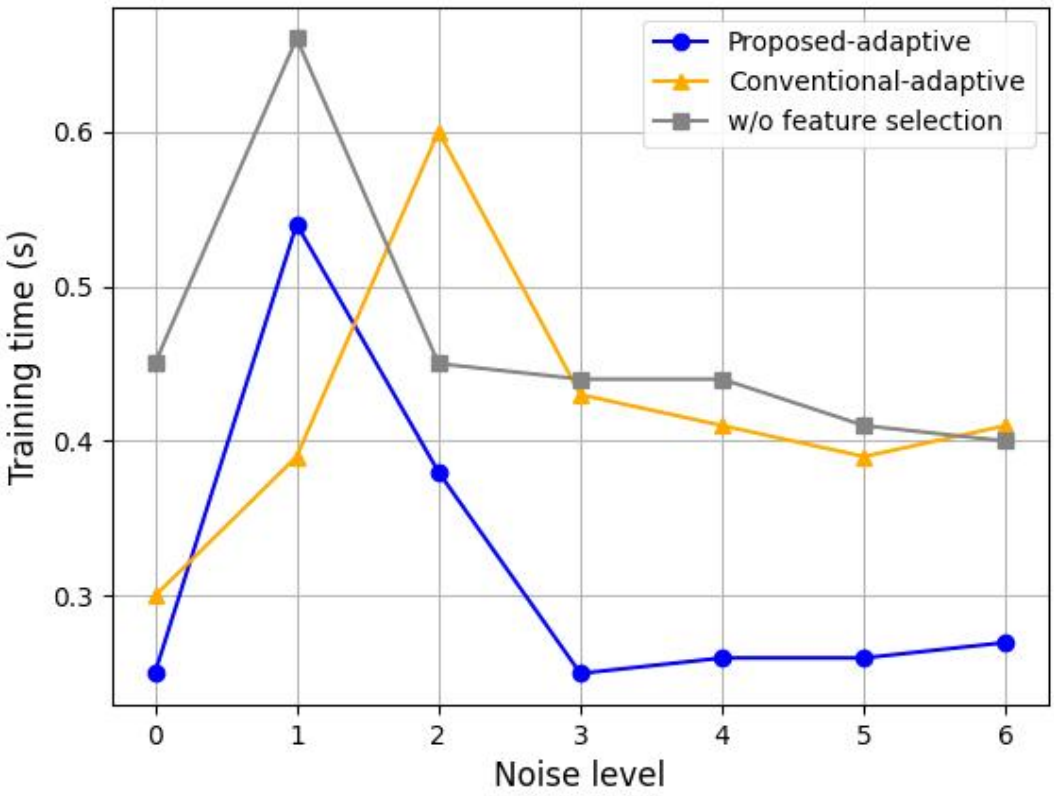

FIGURE 10. Training time of adaptive FSFG, PCA(adaptive), w/o feature selection

FIGURE 10은 특징 선택 기법을 수행하지 않았을 때, 종래 방안을 적응형으로 사용하였을 때와 제안하는 adaptive FSFG의 학습 시간을 비교한 결과이다. 실험 결과에 따르면 제안하는 adaptive FSFG가 노이즈의 수준과 관계없이 시간 지연이 가장 적었다. w/o feature selection의 경우에는 27개의 특징을 모두 사용했으므로 모든 특징들을 처리하는 데 오랜 시간이 걸린 것으로 분석된다. 또한, PCA의 경우에는 adaptive FSFG와 동일한 개수의 특징을 사용했으나, adaptive FSFG 보다 악성 트래픽을 탐지하는 데 기여도가 낮은 특징들을 선택했기 때문에 데이터를 분석하는 데 더 오랜 시간이 걸린 것으로 예상된다. 따라서 adaptive

FSFG가 w/o feature selection 대비 48%, PCA 대비 35%의 학습 시간 개선 효과를 보였다.

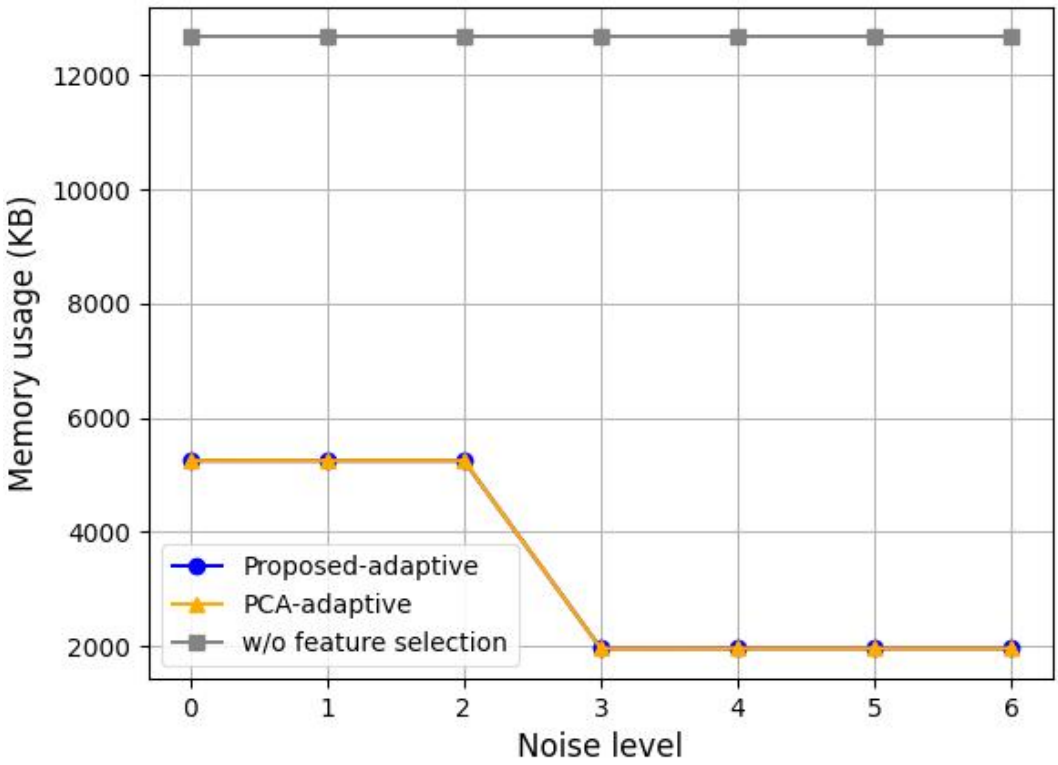

FIGURE 11. Memory usage of adaptive FSFG, PCA(adaptive), w/o feature selection

FIGURE 11은 특징 선택 기법을 수행하지 않았을 때, 종래 방안을 동적으로 사용하였을 때와 제안하는 adaptive FSFG의 메모리 사용량을 비교한 결과이다. 실험 결과, PCA 기법과 제안하는 adaptive FSFG는 동일한 특징 개수를 사용하는 환경이므로 메모리 사용량이 동일한 것으로 나타났다. 또한, 특징 선택 기법을 사용하지 않았을 때는 모든 특징들을 사용했으므로 노이즈 수준에 관계없이 메모리를 가장 많이 사용하였다. 따라서 제안하는 adaptive FSFG에서 특징 선택 기법을 사용하지 않았을 때

대비 메모리 사용량이 71.17% 감소하는 효과를 보였다.

3) 정적 기법과 동적 기법의 비교

FIGURE 12는 제안 방안을 적응형으로 사용하는 것의 효과를 확인하기 위하여 노이즈가 랜덤한 환경에서 제안하는 방안을 정적으로 적용했을 때와 동적으로 적용했을 때의 탐지율을 비교한 결과이다.

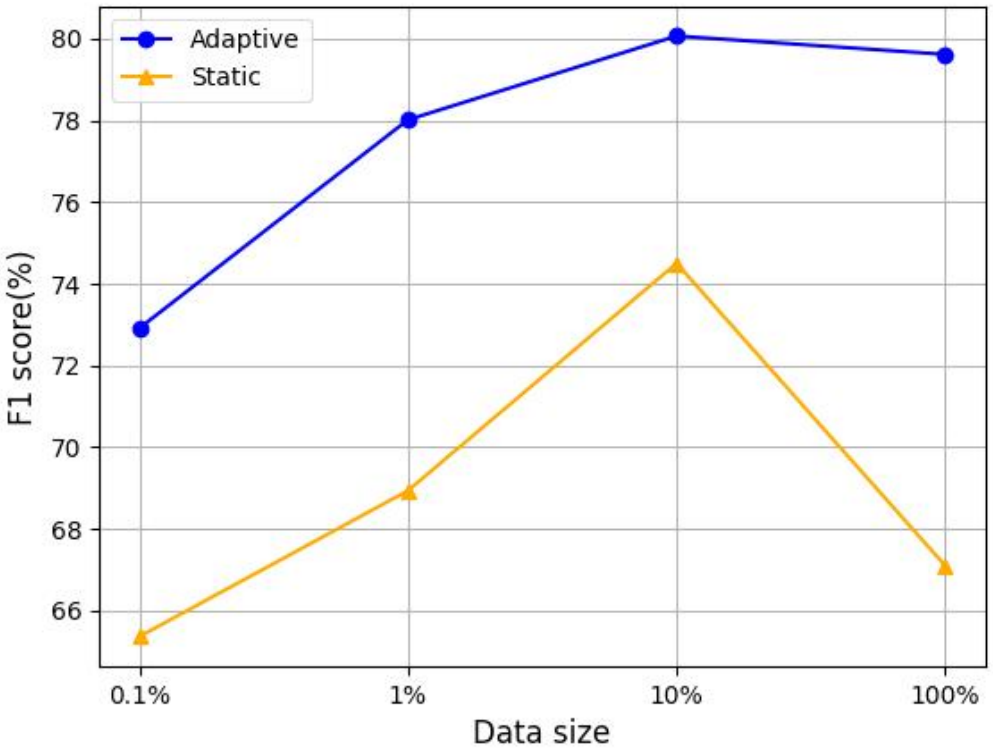

FIGURE 12. F1 score of static, adaptive method

기존 데이터 셋에서 랜덤하게 0.1%, 1%, 10%만큼 샘플링했을 때와 샘플링을 수행하지 않았을 때에 대해 각각 5,000번씩 반복하여 실험하였다. 실험 결과, 정적 방식과 동적 방식에서 사용한 특징의 개수가 동일하여 메모리 사용량과 학습 시간에는 뚜렷한 차이를 보이지 않았으며, 탐지율은 적응형 기법에서 평균 약 8.67%의 개선 효과가 있었다.

Ⅶ. 결 론

빅데이터의 높아진 활용도로 인해 중요 데이터 보호에 대한 요구가 증가하면서, 안전한 데이터의 송수신을 위하여 암호화 기술이 많이 사용되고 있다. 암호화 트래픽의 양이 증가하면서, 패킷 내용을 확인할 수 없고 네트워크 침입 탐지 시스템으로 공격이 쉽게 탐지되지 않는다는 점을 악용한 사이버 공격들이 등장하고 있다. 2020년에 발생한 사이버 공격의 70%는 암호화 기술을 사용하고 있으며, 악성 암호화 트래픽에 기반한 공격이 더욱 많아질 것으로 예상된다. TLS/SSL 기반 악성 트래픽 공격에 대응하기 위하여 심층 패킷 분석, 인공지능 등을 활용한 연구들이 많이 수행되었으나, 여전히 현실적으로는 사용할 수 없다는 한계가 있다.

본 연구에서는 효율적으로 악성 암호화 트래픽을 분류하기 위하여 PCA와 탐지율에 대한 기여도 정보를 활용하여 효율적으로 특징을 선택하고, 노이즈의 수준에 따라 적응형으로 특징을 선택하는 adaptive FSFG를 제안한다. 제안하는 adaptive FSFG에서는 데이터 셋의 차원을 축소한 뒤, 축소한 차원에 따라 탐지율을 확인한다. 이때, 탐지율 그래프에서 기울기가 가파른 지점에서 설명 분산 점수가 큰 특징을 탐지에 유의미한 특징으로 간주한다. 그리고 이렇게 선택한 특징을 활용하여 머신러닝 분류 모델을 학습시킨 뒤, 클라이언트가 위치한 환경의 노이즈가 많을 경우에 분류를 수행한다. 만약 노이즈가 적은 환경이라면, 이렇게 선택한 특징들에 탐지율 기반 최적의 차원에서의 특징을 추가로 활용하여 분류를 수행한다. 실험 결과, 제안하는 adaptive FSFG에서 PCA 기반 축소된 차원 정보만을 활용하였을 때보다 24.74% 개선된 탐지율을 보였으며, 학습 시간 측면에서는 35%의 단축율을 보였다. 또한, 특징 선택 기법을 수행하지 않았을 때 보다는 32.82%만큼 탐지율이 개선되었으며, 48%의 학습 시간 개선 효과를 보였다. 본 연구의 한

계는 데이터 셋을 활용한 시뮬레이션 결과라는 것이며, 추후 연구에서는 다양한 실험 조건에서 현실적인 테스트베드를 구축하여 실증할 것이다.

참고문헌

[1] Yingjie Wang, Guangquan Xu, Xing Liu, Weixuan Mao, Chengxiang Si, Witold Pedrycz, Wei Wang(2020), “Identifying vulnerabilities of SSL/TLS certificate verification in Android apps with static and dynamic analysis,” Journal of Systems and Software, Volume 167, 110609, ISSN 0164-1212, https://doi.org/10.1016/j.jss.2020.110609.

[2] Zihao Wang, Kar Wai Fok, Vrizlynn L.L. Thing(2022), “Machine learning for encrypted malicious traffic detection: Approaches, datasets and comparative study,” Computers & Security, Volume 113, 102542, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2021.102542.

[3] Ferriyan A, Thamrin AH, Takeda K, Murai J(2022), “Encrypted Malicious Traffic Detection Based on Word2Vec,” Electronics, 11(5):679. https://doi.org/10.3390/electronics11050679

[4] Bingfeng Xu, Gaofeng He, Haiting Zhu(2021), “ME-Box: A reliable method to detect malicious encrypted traffic,” Journal of Information Security and Applications, Volume 59, 102823, ISSN 2214-2126, https://doi.org/10.1016/j.jisa.2021.102823.

[5] D. F. Isingizwe, M. Wang, W. Liu, D. Wang, T. Wu and J. Li(2021), “Analyzing Learning-based Encrypted Malware Traffic Classification with AutoML,” 2021 IEEE 21st International Conference on Communication Technology (ICCT), pp. 313-322, doi: 10.1109/ICCT52962.2021.9658106.

[6] Yong Fang, Kai Li, Rongfeng Zheng, Shan Liao, Yue Wang(2021), “A

communication-channel-based method for detecting deeply camouflaged malicious traffic," Computer Networks, Volume 197, 108297, ISSN 1389-1286, https://doi.org/10.1016/j.comnet.2021.108297.

[7] S. Liu, Y. Han, Y. Hu and Q. Tan(2021), "FA-net: Attention-based Fusion Network For Malware HTTPs Traffic Classification," 2021 IEEE Symposium on Computers and Communications (ISCC), 2021, pp. 1-7, doi: 10.1109/ISCC53001.2021.9631419

[8] Chuanpu Fu, Qi Li, Meng Shen, and Ke Xu(2021), "Realtime Robust Malicious Traffic Detection via Frequency Domain Analysis," In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21), 2021, Association for Computing Machinery, New York, NY, USA, 3431 - 3446. https://doi.org/10.1145/3460120.3484585

[9] J. Shi, Y. Lin, Z. Zhang and S. Yu(2021), "A Hybrid Intrusion Detection System Based on Machine Learning under Differential Privacy Protection," 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), 2021, pp. 1-6, doi: 10.1109/VTC2021-Fall52928.2021.9625540.

[10] J. Zhao, Y. Chen and W. Zhang(2019), "Differential Privacy Preservation in Deep Learning: Challenges, Opportunities and Solutions," in IEEE Access, vol. 7, pp. 48901-48911, 2019, doi: 10.1109/ACCESS.2019.2909559.

[11] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu(2021), "Accumulative poisoning attacks on real-time data," In NeurIPS.

[12] Junhao Zheng, Patrick P.K. Chan, Huiyang Chi, Zhimin He(2022), “A concealed poisoning attack to reduce deep neural networks’ robustness against adversarial samples,” Information Sciences, Volume 615, Pages 758-773, ISSN 0020-0255, https://doi.org/10.1016/j.ins.2022.09.060.

[13] Chen Wang, Jian Chen, Yang Yang, Xiaoqiang Ma, Jiangchuan Liu(2022), “Poisoning attacks and countermeasures in intelligent networks: Status quo and prospects,” Digital Communications and Networks, Volume 8, Issue 2, Pages 225-234, ISSN 2352-8648, https://doi.org/10.1016/j.dcan.2021.07.009.

[14] V. Shejwalkar and A. Houmansadr(2021), “Manipulating the byzantine: optimizing model poisoning attacks and defenses for federated learning,” NDSS, The Internet Society.

[15] Mohammadreza MontazeriShatoori, Logan Davidson, Gurdip Kaur, and Arash Habibi Lashkari(2020), “Detection of DoH Tunnels using Time-series Classification of Encrypted Traffic,” The 5th IEEE Cyber Science and Technology Congress, Calgary, Canada, August 2020.

[16] Xun Liu, Junling You, Yulei Wu, Tong Li, Liangxiong Li, Zheyuan Zhang, Jingguo Ge(2020), “Attention-based bidirectional GRU networks for efficient HTTPS traffic classification,” Information Sciences, Volume 541, Pages 297-315, ISSN 0020-0255, https://doi.org/10.1016/j.ins.2020.05.035.

[17] M. Behnke et al.(2021), “Feature Engineering and Machine Learning Model Comparison for Malicious Activity Detection in the DNS-Over-HTTPS Protocol,” in IEEE Access, vol. 9, pp. 129902-129916, doi: 10.1109/ACCESS.2021.3113294.

[18] Li, Kunlin, and Baojiang Cui(2021), “Malicious Encrypted Traffic Identification Based on Four-Tuple Feature and Deep Learning,” Innovative Mobile and Internet Services in Ubiquitous Computing, June, 199 – 208. https://doi.org/10.1007/978-3-030-79728-7_20.

[19] Zhang, Xueqin, Min Zhao, Jiyuan Wang, Shuang Li, Yue Zhou, and Shinan Zhu(2022), “Deep-Forest-Based Encrypted Malicious Traffic Detection,” Electronics 11 (7): 977. https://doi.org/10.3390/electronics11070977.

[20] Onur Barut, Rebecca Zhu, Yan Luo, and Tong Zhang(2020), “TLS Encrypted Application Classification Using Machine Learning with Flow Feature Engineering,” In 2020 the 10th International Conference on Communication and Network Security (ICCNS 2020). Association for Computing Machinery, New York, NY, USA, 32 – 41. https://doi.org/10.1145/3442520.3442529

[21] Shafiq, Muhammad, Zhihong Tian, Ali Kashif Bashir, Xiaojiang Du, and Mohsen Guizani(2020), “IoT Malicious Traffic Identification Using Wrapper-Based Feature Selection Mechanisms,” Computers & Security 94 (July): 101863. https://doi.org/10.1016/j.cose.2020.101863.

[22] Michal Piskozub, Fabio De Gaspari, Freddie Barr-Smith, Luigi Mancini, and Ivan Martinovic(2021), “MalPhase: Fine-Grained Malware Detection Using Network Flow Data,” Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security, (ASIA CCS '21), Association for Computing Machinery, New York, NY, USA, 774 – 786. https://doi.org/10.1145/3433210.3453101

[23] Chuanpu Fu, Qi Li, Meng Shen, and Ke Xu(2021) “Realtime Robust

Malicious Traffic Detection via Frequency Domain Analysis," In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21), Association for Computing Machinery, New York, NY, USA, 3431 – 3446, https://doi.org/10.1145/3460120.3484585

[24] Jinchi Huang, Lie Qu, Rongfei Jia, Binqiang Zhao(2019), "O2U-Net: A Simple Noisy Label Detection Approach for Deep Neural Networks," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 3326-3334

[25] C. Manzano, C. Meneses, P. Leger, H. Fukuda(2022), "An Empirical Evaluation of Supervised Learning Methods for Network Malware Identification Based on Feature Selection," Complexity, vol. 2022, Article ID 6760920, 18 pages, https://doi.org/10.1155/2022/6760920

[26] Owen, Harry, Javad Zarrin, and Shahrzad M. Pour(2022), "A Survey on Botnets, Issues, Threats, Methods, Detection and Prevention," Journal of Cybersecurity and Privacy 2, no. 1: 74-88. https://doi.org/10.3390/jcp2010006

[27] Wang, Xusheng, Linlin Zhang, Kai Zhao, Xuhui Ding, and Mingming Yu(2022), "MFDroid: A Stacking Ensemble Learning Framework for Android Malware Detection," Sensors 22, no. 7: 2597. https://doi.org/10.3390/s22072597

[28] C. Manzano, C. Meneses, P. Leger, H. Fukuda(2022), "An Empirical Evaluation of Supervised Learning Methods for Network Malware Identification Based on Feature Selection," Complexity, vol. Article ID 6760920, 18 pages, 2022. https://doi.org/10.1155/2022/6760920

[29] Afeez Ajani Afuwape, Ying Xu, Joseph Henry Anajemba, Gautam Srivastava(2021), “Performance evaluation of secured network traffic classification using a machine learning approach, Computer Standards & Interfaces,” Volume 78, 103545, ISSN 0920-5489, https://doi.org/10.1016/j.csi.2021.103545.

ABSTRACT

Principal Component Analysis based Adaptive Feature Selection for Encrypted Malware Traffic Classification

Yurim Lee
Department of Future Convergence Technology Engineering
Graduate School of
Sungshin Women's University

As data security has become increasingly important in all industries, encryption technology is critical for secure data transmission and reception, and the proportion of encrypted traffic is rapidly increasing. However, advanced attacks based on malicious encrypted traffic have emerged, exploiting the inability of Network Intrusion Detection Systems (NIDS) to classify malware due to the inability to verify packet contents. Although response research using Deep Packet Inspection (DPI) and artificial intelligence has been conducted to solve this problem, it is still difficult to use it in the real industry due to privacy concerns and limitations that cannot be detected in real-time. Furthermore, the difficulty of training and classifying network traffic with poisoning attack that intentionally injects noise to disrupt training models is increasing. In this study, a feature selection technique based on dimension reduction is proposed to efficiently

classify malicious encrypted traffic using principal component analysis (PCA). After reducing the dimension of data from 1D to 27D in an environment where noise levels vary, the change in detection rate is checked. Then, an adaptive Feature Selection based F1-score Gradient (adaptive FSFG) is proposed, which utilizes high-explained variance ratio features at points with steep gradient or selectively adds features with reduced dimensions. As a result, the detection rate improved by 24.74%, the training time improved by 35% when only PCA was used, the detection rate improved by 32.82%, and the training time improved by 48% when no feature selection was used. Furthermore, by contrasting the static and adaptive applications of the proposed method, the effectiveness of the proposed adaptive FSFG was demonstrated. The adaptive FSFG can be used to classify encrypted malware traffic quickly and efficiently, and can be applied when other data protection techniques were used.