

PRISM-2026:엔드포인트 기반 자동 라벨링 다계층 네트워크 트래픽 데이터셋

김 지 민*, 유 경 민*, 장 윤 성**, 남 승 우**, 김 주 성**, 백 의 준^{oo}, 김 명 섭^o

PRISM-2026: A Multi-Level Network Traffic Dataset with Endpoint-Based Automatic Labeling

Ji-min Kim*, Gyeong-min Yu*, Yun-Seong Jang**, Seung-Woo Nam**, Ju-Sung Kim**, Ui-Jun Baek^{oo}, Myung-Sup Kim^o

요 약

암호화 트래픽의 확산으로 정확하게 라벨링된 고품질 데이터셋 확보가 트래픽 분류 연구의 핵심 과제로 대두되고 있다. 기존 데이터셋은 단일 응용 격리, 시간 구간 분할 등의 방식에 의존하여 실사용 환경의 동시 트래픽을 반영하지 못하거나 라벨 무결성에 한계가 있다. 본 연구는 트래픽을 발생시킨 엔드포인트로부터 라벨을 직접 획득하는 자동 라벨링 프레임워크 PRISM을 활용하여 다계층 네트워크 트래픽 데이터셋 PRISM-2026을 구축 공개한다. PRISM-2026은 7대의 실사용 호스트에서 3일간 수집한 1,320,366개의 세션으로 구성되며, 각 세션에 5-튜플 중단점 정보와 프로세스/서비스/사이트의 3단계 라벨을 부여한다. 특히 svchost.exe에 대한 서비스 단위 라벨링과 SNI 기반 브라우저 사이트 단위 라벨링을 도입하여, 프로세스 단위 라벨이 구분하지 못하던 공유 호스트 프로세스와 브라우저 트래픽을 세분화하고, 프로세스 단위(108종)와 응용 단위(89종)의 두 라벨 집합을 제공한다. 대표적 분류 모델(ET-BERT, TSHIRT, 2D CNN, XGBoost) 기준 실험에서 XGBoost가 최고 정확도(0.917)를 기록하였다.

Key Words : Network Traffic Classification, Traffic Dataset, Automatic Labeling, Endpoint-based Labeling, Encrypted Traffic

ABSTRACT

As encryption becomes pervasive, the availability of accurately labeled, high-quality datasets is a central challenge in network traffic classification. Existing datasets rely on single-application isolation or time-interval segmentation and thus fail to capture the concurrent traffic of real hosts or suffer from limited label integrity. This study builds and releases PRISM-2026, a multi-level network traffic dataset created with PRISM, an automatic labeling framework that obtains labels directly from the endpoints generating the traffic. PRISM-2026 comprises 1,320,366 sessions collected from seven in-use hosts over three days, with each session annotated by five-tuple endpoint information and three-level labels spanning process, service, and site. In particular, service-level labeling for svchost.exe and SNI-based site-level labeling for browsers refine the shared-host-process and browser traffic that process-level labels alone cannot distinguish, yielding two label sets at the process level (108 classes) and application level (89 classes). In baseline experiments with representative classifiers (ET-BERT, TSHIRT, 2D CNN, XGBoost), XGBoost achieved the highest accuracy (0.917).

* 이 논문은 이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구이며(RS-2025-16067207, 양자 네트워크 라우팅 및 제어/관리 핵심 기술과 표준 개발), 2026년도 정부(산업통상부)의 재원으로 한국산업기술진흥원의 지원을 받아 수행된 연구임 (P0031136, 2026년 중부권 지역혁신클러스터육성).

• First Author : Korea University Department of Computer Science and Software Engineering, illiard1209@korea.ac.kr

^o Corresponding Author : Korea University Department of Computer Science and Software Engineering, tmskim@korea.ac.kr

^{oo} Corresponding Author : Korea Institute of Science and Technology Information (KISTI), pb1069@kisti.re.kr

* Korea University Department of Computer Science and Software Engineering rudals2710@korea.ac.kr

** Korea University Department of Computer Information and Science, {brave1094, nam131119, jsung0514}@korea.ac.kr

논문번호 : KNOM2026-01-007, Received July 25, 2026; Revised August 10, 2026; Accepted August 18, 2026

I. 서론

네트워크 기술이 고도화되면서 이를 안정적으로 관리하고 모니터링하기 위한 기술의 중요성도 함께 커지고 있다. 특히 트래픽 분류는 이상 트래픽 탐지, 사용자 행위 분석 등 다양한 응용으로 확장될 수 있어 네트워크 관리 분야에서 핵심적인 역할을 담당한다.

네트워크 트래픽 분류 기술은 포트 분석[1], DPI 기반 분석[2]을 거쳐 ML/DL 기반 방식[3]으로 발전해왔으며, 포트의 동적 할당과 암호화 트래픽의 증가 등으로 인해 최근에는 ML/DL 기법이 주로 연구되고 있다[4]. 다만 이러한 학습 기반의 모델 성능은 데이터셋의 품질에 의한 성능의 편차가 커서 정확하게 라벨링된 고품질 데이터셋의 확보가 무엇보다 중요하다.

본 연구에서는 네트워크 트래픽 분류 분야에서 사용되는 데이터셋들의 수집 방법 및 품질을 분석하고, 기존 데이터셋들이 지닌 구조적 한계를 극복하기 위한 자동 라벨링 프레임워크 PRISM을 활용한 데이터셋 PRISM-2026을 제안한다. PRISM-2026은 다음과 같은 특성을 갖는다. 첫째, 라벨을 패킷에서 추론하는 대신 트래픽을 발생시킨 엔드포인트로부터 직접 획득하여 무결성을 보장한다. 둘째, 라벨링 단위를 프로세스·서비스·방문 사이트 수준으로 유연하게 확장할 수 있는 확장성을 제공한다. 셋째, 특정 애플리케이션만을 격리해 수집하는 방식이 아니라 실제 호스트에서 동시에 발생하는 트래픽을 그대로 담아 실사용 환경을 반영한다. 넷째, 수집부터 라벨링까지 전 과정을 사람의 개입 없이 완전히 자동화하여 다중 호스트에서 지속적으로 데이터셋을 축적할 수 있다.

II. 관련 연구

네트워크 트래픽 분류 연구에서 데이터셋은 모델의 학습과 평가를 좌우하는 핵심 요소이며, 특히 딥러닝 기반 분류의 성능은 라벨의 정확성과 트래픽의 다양성에 크게 의존한다. 본 절에서는 기존 주요 데이터셋을 라벨링 방식에 따라 세 가지로 분류하고 각 방식이 지닌 구조적 한계를 논의한 뒤, 본 연구가 기반으로 하는 선행 프레임워크와의 관계를 설명한다.

2.1. 단일 응용 격리 수집 방식

ISCX-VPN-NonVPN 데이터셋[5]은 목표 응용 하나만 실행하고 나머지 모든 서비스와 응용을 종료한 상태에서 트래픽을 캡처하여, 해당 시간대의 트래픽 전체를 그 응용으로 간주하는 방식을 사용한다. 이 방식은 단순하고 직관적이지만, 복수의 응용이 동시에 실행되는 실제 호스트 환경을 반영하지 못하고, 목표 응용을 격리하더라도 운영체제의 백그라운드 서비스 트래픽이 혼입될 가능성이 있으며, 매 응용마다 수동으로 캡처 세션을 구성해야 하므로 수집 규모와 현실성이 모두 제한된다.

2.2. 시간 구간 분할 기반 라벨링 방식

CIC-IDS2017 데이터셋[6]은 침입 탐지 시스템 평가를 위해 정상 트래픽과 공격 트래픽을 사전에 정의된 시간대에 따라 구분하여 라벨을 부여한다. USTC-TFC2016[7]은 오늘날까지 널리 사용되는 바이트 이미지 기반 합성곱 처리 파이프라인을 정립하였다. 두 데이터셋 모두 시간 또는 캡처 구간을 기준으로 라벨을 부여하므로, 공격 시간대에 혼재하는 정상 트래픽을 정밀하게 구분하기 어렵고, svchost.exe와 같이 하나의 프로세스가 여러 서비스를 실행하는 공유 호스트 트래픽을 세분화할 수 없다는 한계를 지닌다.

2.3. 호스트 기반 프로세스 태깅 방식

호스트 기반 패킷 캡처 도구인 Microsoft Network Monitor는 대상 호스트에서 직접 트래픽을 캡처하면서 각 패킷에 소유 프로세스 정보를 태깅할 수 있어 라벨링에 유리하다. 그러나 이 방식은 단일 호스트에서만 동작하며, 고트래픽 환경에서는 캡처 도구 자체의 CPU 및 디스크 I/O 부하로 인해 저장 자체가 안 되는 문제가 발생한다. 또한 호스트나 캡처 에이전트에 장애가 발생하면 수집이 함께 중단되어 트래픽 수집의 연속성이 보장되지 않는다.

2.4. 선행 프레임워크와 본 연구의 관계

본 연구는 선행 연구에서 제안한 프로세스 수준 자동 라벨링 프레임워크[8]를 기반으로 한다. [8]은 스위치 포트 미러링을 통한 호스트 독립적 트래픽 수집과 Windows 소켓 폴링 및

표 1. 기존 데이터셋과 PRISM-2026의 비교

Table 1. Comparison of Existing Datasets and PRISM-2026

데이터셋	라벨링 방식	동시 실행 반영	라벨 단위	다중 호스트·연속수집
ISCX-VPN-NonVPN	단일 응용 격리	×	응용	×
CIC-IDS2017	시간 구간 분할	×	정상/공격	×
MS Network Monitor	호스트 태깅	○	응용	×
PRISM-2026(제안)	엔드포인트 매칭	○	프로세스/서비스/사 이트	○

ETW 보조 수집을 결합하고, 소켓 직접 매칭·규칙 저장소·캐시 계층으로 구성된 다단계 매칭 파이프라인과 SNI 기반 후처리를 통해, 7대 호스트 24시간 연속 수집 환경에서 99.8% 매칭률을 달성하였다. 이로써 앞서 언급한 세 방식의 한계·무결성, 확장성, 수집 연속성-를 상당 부분 해소하였으나, 라벨링 단위를 프로세스로 한정하였다는 제약이 있다. 실제로 [8]에서 전체 세션의 절반 이상을 차지한 svchost.exe와 그 뒤를 이은 브라우저 트래픽은 각각 하나의 프로세스로만 표현되어, 공유 호스트 프로세스 내부의 개별 서비스나 브라우저가 접속한 목적지를 구분하지 못한다. 본 연구는 이 프레임워크 위에서 svchost.exe에 대한 서비스 단위 라벨링과 브라우저에 대한 사이트 단위 라벨링을 도입하여 라벨 granularity를 확장하고, 이를 통해 구축한 데이터셋을 공개한다.

2.5. 데이터셋 비교

앞서 논의한 기존 데이터셋과 PRISM-2026의 특성을 표1에 정리하였다. PRISM은 엔드포인트에서 직접 획득한 라벨을 통해 무결성을 확보하고, 실제 호스트의 동시 실행 트래픽을 반영하며, 프로세스·서비스·사이트의 3단계 라벨 granularity를 지원한다는 점에서 기존 데이터셋과 차별화된다.

Ⅲ. 본 론

3.1. 시스템 개요

본 연구의 데이터셋은 선행 연구에서 제안한 프로세스 수준 자동 라벨링 프레임워크 PRISM을 기반으로 생성되었다. PRISM은 스위치 포트 미러링을 통해 대상 호스트에 부하를 주지 않

고 트래픽을 수집하고, 각 Windows 호스트에서 50ms 주기의 소켓 폴링과 ETW(Event Tracing for Windows) 기반 보조 수집을 병행하여 소켓과 프로세스의 대응 관계를 확보한다. 수집된 소켓 정보는 소켓 직접 매칭, 규칙 저장소, 캐시 계층으로 구성된 다단계 매칭 파이프라인과 SNI 기반 후처리를 거쳐 각 세션에 이를 발생시킨 프로세스 라벨을 부여한다. 선행 연구[8]는 7대 호스트 24시간 연속 수집 환경에서 99.8%의 프로세스 매칭률을 보고한 바 있다.

표 2. PRISM-2026 일자별 프로세스 매칭률 및 공개본 구성
Table 2. Daily Process-Matching Rate and Composition of the Released PRISM-2026 Dataset

구분	2026. 07.10	2026. 07.11	2026. 07.12	합계
전체 세션	538,176	458,699	464,257	1,461,132
unknown	5,623	5,507	5,612	16,742
매칭률	98.96%	98.80%	98.79%	98.85%
System	44,022	38,023	40,891	122,936
소수 라벨(<10)	389	363	336	1,088
공개 세션	488,142	414,806	417,418	1,320,366

본 데이터셋 구축 과정에서 측정한 매칭률과 공개본 구성 내역은 표 2와 같다. 3일간 수집된 전체 1,461,132개 세션 가운데 프로세스를 확정하지 못한 세션은 16,742개(1.15%)로, 프로세스 매칭률은 98.85%였다. 매칭에 실패한 세션은 라벨을 임의로 추정하지 않고 unknown으로 분리하였으며, 커널에 귀속되어 특정 응용으로 환원되지 않는 System 트래픽 122,936개(8.41%)와 함께 공개본에서 제외하였다. 여기에 더해 특정 호스트·일자에서 세션 수가 10개 미만으로 관측되어 통계적 의미를 갖기 어려운 라벨의 세션 1,088개(0.07%)를 추가로 제외하였다. 그 결과 공개되는 1,320,366개 세션(전체의 90.37%)은 모두 프로세스 라벨이 확정된 세션으로 구성된다.

매칭 실패 세션은 일자별로 5,507 - 5,623개로 거의 일정하며, 매칭률의 일자 간 차이(98.79 - 98.96%)는 주로 일자별 총 세션 수의 차이에서 기인한다. 본 프레임워크의 규칙 저장소와 캐시 계층은 반복 관측되는 소켓 - 프로세스 대응 패턴을 지속적으로 누적하므로, 수집이 장기화될 수록 소켓 직접 매칭에 실패한 세션을 흡수하여 **unknown** 비율을 낮출 여지가 있다. 다만 3 일이라는 관측 구간에서는 그러한 감소 추세가 뚜렷하게 확인되지 않았는데, 이는 규칙과 캐시가 충분히 축적되기에 기간이 짧았거나, 잔여 **unknown**이 폴링 주기와 ETW 이벤트 모두에서 포착되지 않는 극히 짧은 수명의 세션과 같은 구조적 요인에 기인할 가능성을 시사한다. 수집 기간을 확대하여 규칙 축적에 따른 매칭률의 변화를 관측하는 것은 향후 연구 과제로 남긴다. 한편 여기서의 매칭률은 세션에 라벨이 부여된 비율을 의미하며, 부여된 라벨이 실제와 일치하는 비율, 즉 라벨 정확도와는 구분되는 지표이다. 라벨 정확도에 대한 정량적 검증은 본 연구의 한계로 5.1절에서 논의한다.

그러나 선행 프레임워크는 라벨링 단위를 프로세스로 한정하였다는 제약이 있다. 실제 수집 결과에서 전체 세션의 절반 이상을 차지한 **svchost.exe**는 다수의 Windows 서비스를 대리 실행하는 공유 호스트 프로세스임에도 하나의 라벨로만 표현되며, 그 뒤를 잇는 브라우저 트래픽 역시 **chrome.exe**·**msedge.exe** 등 프로세스 이름으로만 묶여 실제 접속한 목적지를 구분하지 못한다. 이는 동일한 프로세스 라벨 안에 성격이 전혀 다른 트래픽이 혼재함을 의미하며, 세밀한 트래픽 분석을 목표로 하는 데이터셋의 활용도를 제한한다.

본 연구는 이 프레임워크 위에서 라벨 **granularity**를 확장한다. 구체적으로 (i) **svchost.exe**에 대해서는 소켓을 실제로 점유한 서비스를 식별하는 서비스 단위 라벨링을, (ii) 브라우저에 대해서는 접속 목적지를 식별하는 사이트 단위 라벨링을 도입한다. 그 결과 최종 라벨은 프로세스·서비스·사이트의 3단계 **granularity**를 갖게 되며, 수집부터 라벨링·저장까지의 전 과정은 사람의 개입 없이 자동으로 수행되어 다중 호스트에서 지속적으로 데이터셋을 축적한다. 이렇게 구축된 데이터셋을 PRISM-2026로 명명하고 공개한다.

3.2. 프로세스 수준 자동 라벨링 파이프라인

데이터셋 생성의 토대가 되는 파이프라인은 트래픽 수집부, 소켓 수집부, 매칭부, 세션 생성부의 네 단계로 구성되며, 각 단계는 독립 프로세스로 상시 동작한다.

트래픽 수집부는 스위치의 포트 미러링 포트에 연결된 수집 서버에서 미러링된 원본 트래픽을 그대로 받아 1초 주기로 플로우를 생성한다. 대상 호스트에 에이전트 부하나 캡처 손실을 유발하지 않으므로 고트래픽 환경에서도 수집 연속성이 유지된다. 생성된 플로우와 패킷은 5-튜플(프로토콜, 출발지·목적지 IP 및 포트)을 기준으로 세션 단위로 정리된다.

소켓 수집부는 각 Windows 호스트에서 동작하며, 50ms 주기의 소켓 테이블 폴링으로 (5-튜플, PID, 프로세스 경로)의 대응 관계를 지속적으로 수집한다. 짧은 수명의 세션이 폴링 주기 사이에 누락되는 것을 보완하기 위해 ETW 기반 이벤트 수집을 병행하여, 연결 생성·종료 시점의 소켓-프로세스 정보를 함께 확보한다. 수집된 소켓 정보는 수집 서버로 전송되어 매칭부의 입력이 된다.

매칭부는 미러링에서 얻은 세션과 호스트에서 얻은 소켓 정보를 대응시켜 각 세션에 프로세스 라벨을 부여한다. 매칭은 비용이 낮은 순서대로 계층화되어, 먼저 5-튜플과 시간 구간이 정확히 일치하는 소켓을 찾는 소켓 직접 매칭을 시도하고, 실패한 세션에 대해서는 반복적으로 관측되는 패턴을 축적한 규칙 저장소와 최근 매칭 결과를 보관한 캐시 계층을 순차적으로 적용한다. 이 다단계 구조는 폴링 누락이나 시간 오차로 직접 매칭이 되지 않는 세션까지 흡수하여 매칭률을 높인다.

세션 생성부는 라벨이 확정된 세션을 파일 시스템에 물리적으로 정리한다. 이때 세션을 발생시킨 호스트, 부여된 라벨에 따라 디렉토리를 구성하고 각 세션을 개별 pcap 파일로 저장한다. 이 단계의 출력이 곧 PRISM-2026 데이터셋의 원본이며, 이후 3.3 - 3.4에서 설명하는 서비스·사이트 단위 라벨링은 이 파이프라인의 후처리 단계로 통합되어 동작한다.

3.3. svchost.exe 호스트 단위 라벨링

svchost.exe는 여러 Windows 서비스를 하나의 프로세스로 대리 실행하는 공유 호스트 프로세스

스로, PID만으로는 어떤 서비스가 해당 소켓을 사용했는지 구분할 수 없다. 따라서 프로세스 단위 매칭 결과가 svchost.exe인 세션은 성격이 서로 다른 다수 서비스의 트래픽이 하나의 라벨에 묻혀 있는 상태가 된다.

이를 해소하기 위해 소켓 수집 단계에서 각 소켓을 점유한 서비스를 함께 식별한다. Windows에서 svchost.exe 인스턴스와 그 내부에서 구동되는 서비스의 대응 관계를 조회하여, 소켓의 소유 PID를 프로세스뿐 아니라 개별 서비스까지 해석한다. 그 결과 svchost.exe 라벨은 소켓을 실제로 사용한 서비스 이름으로 세분화되며, 예컨대 원격 데스크톱(TermService), 이름 확인(Dnscache), 백그라운드 전송(BITS), 진단 데이터(DiagTrack), 업데이트 전송 최적화(DoSvc), 인증서 서비스(CryptSvc) 등으로 구분된다.

이 서비스 단위 라벨링을 통해, 기존에 전체 세션의 절반 이상을 차지하며 분석을 저해하던 svchost.exe 트래픽이 서비스별로 분리되어, 각 서비스의 통신 패턴을 독립적으로 관찰할 수 있게 된다.

3.4. 브라우저 사이트 단위 라벨링

브라우저 트래픽은 프로세스 라벨이 chrome.exe·msedge.exe 등으로 동일하더라도 실제로는 서로 다른 다수의 웹 서비스에 대한 접속이 혼재한다. 프로세스 라벨만으로는 이러한 목적지 차이를 구분할 수 없으므로, 브라우저로 매칭된 세션에 대해 접속 사이트를 식별하는 후처리를 추가한다.

사이트 식별은 암호화된 트래픽에서도 평문으로 노출되는 TLS ClientHello의 SNI(Server Name Indication) 필드를 활용한다. 세션의 SNI에서 목적지 도메인을 추출하고, 이를 사전에 정의한 서비스·사이트 매핑에 대응시켜 각 세션을 접속한 사이트 단위로 라벨링한다. 그 결과 동일한 브라우저 프로세스에서 발생한 트래픽이 접속 목적지(예: Google, Microsoft Edge 관련 서비스, KakaoTalk, Notion 등)별로 분리된다. SNI가 존재하지 않거나 매핑되는 목적지를 확인할 수 없는 세션은 별도의 no_visit 그룹으로 분류하여 라벨의 신뢰성을 유지한다.

이 사이트 단위 라벨링은 브라우저라는 단일 프로세스 안에서 사용자가 실제로 소비한 서비스를 구분해내며, 서비스 단위 라벨링과 함께 데이터셋의 라벨 granularity를 프로세스·서비스·사이트의 3단계로 확장한다.

3.5. 디렉토리 구조

PRISM-2026은 동일한 세션 집합을 세 가지 관점으로 제공하며, 라벨 계층을 그대로 파일시스템 경로에 반영하여 별도의 색인 없이 디렉토리 탐색만으로 원하는 단위의 트래픽을 선택할 수 있도록 구성된다. 실제 pcap 파일은 sessions/에만 존재하고, 나머지 두 관점은 이를 가리키는 상대 심볼릭 링크로 제공되어 저장 공간을 추가로 소모하지 않는다.

PRISM-2026/

```
├── sessions/<수집날짜>/<호스트ID>/<프로세스
    라벨>.exe/<세션>.pcap
    │
    └── /svchost.exe/<서비스 이름>/<세션>.pcap
        │
        └── /<브라우저>.exe/<사이트 라벨 |
            no_visit>/<세션>.pcap
            ├── byproduct/<수집날짜>/<응용 라벨>/...
                (응용 단위, 89종)
            └── byproduct_process/<수집날짜>/<프로세스
                라벨>/... (프로세스 단위, 108종)
                └── *.manifest.tsv
                    (관점 ↔ sessions 경로 대응표)
```

sessions/ 는 수집 원본 레이아웃으로 호스트 단위 분석에 사용하고, byproduct/ 와 byproduct_process/ 는 호스트를 통합하여 각각 응용 단위·프로세스 단위 분류 실험에 바로 사용할 수 있다. 세 관점의 세션 수는 모두 1,320,366으로 동일하다. 호스트 식별자는 개인 정보 보호를 위해 익명화된 값이다.

개별 세션 파일은 5-튜플을 그대로 파일명에 담아, 파일명만으로 방향·종단점을 식별할 수 있도록 한다. 명명 규칙은 다음과 같으며, 동일 5-튜플이 재발생한 경우 캡처 시각(연월일_시분초_밀리초)을 접미로 덧붙여 구분한다.

```
TCP_<출발지IP>_<출발지포트>_<목적지IP>_
    <목적지포트>[<캡처시각>].pcap
```

이와 같은 경로 기반 구성은 호스트별·프로세스별·서비스별·사이트별 트래픽을 즉시 분리·선택할 수 있게 하며, 날짜 단위 파티셔닝은 다중

호스트 연속 수집 결과를 지속적으로 누적·관리하는 데 적합하다. 또한 날짜별로 분리된 구성은 학습·시험 집합을 날짜 기준으로 나누는 평가에도 그대로 활용할 수 있다.

3.6. 라벨 체계 및 세션 필드

PRISM-2026의 라벨은 앞 절에서 설명한 다단계 매칭 및 후처리 파이프라인의 출력으로 확정되며, 프로세스·서비스·사이트의 3단계 **granularity**를 갖는다. 가장 기본이 되는 프로세스 단위 라벨은 세션을 발생시킨 실행 파일 이름(예: `chrome.exe`, `spotify.exe`)으로 부여되고, `svchost.exe`에 대해서는 소켓을 실제로 점유한 서비스 이름으로, 브라우저에 대해서는 SNI로 식별한 접속 사이트 이름으로 세분화된다. 본 데이터셋은 이러한 라벨 계층을 활용하여 두 가지 관점의 라벨 집합을 함께 제공한다. 하나는 실행 파일 단위를 그대로 반영한 프로세스 단위 라벨(108종)이며, 다른 하나는 동일 응용을 서비스·사이트 관점으로 통합·정규화한 응용 단위 라벨(89종)이다. 후자는 `chrome.exe·msedge.exe` 등으로 흩어진 브라우저 트래픽을 접속 목적지(예: Google Chrome, ChatGPT, Claude, Notion) 기준으로 재정렬한 결과로, 프로세스 이름만으로는 드러나지 않는 사용자 수준의 서비스 소비 양상을 반영한다.

각 세션은 하나의 `pcap` 파일로 저장되며, 세션을 식별하는 핵심 필드는 파일명에 그대로 인코딩된 5-튜플(전송 프로토콜, 출발지 IP·포트, 목적지 IP·포트)이다. 여기에 수집 날짜, 호스트 IP, 라벨(프로세스/서비스/사이트) 정보가 디렉토리 경로로 부여되며, 동일 5-튜플이 재발생한 세션은 캡처 시각(연월일_시분초_밀리초) 접미로 구분된다. SNI가 존재하지 않거나 매핑 가능한 목적지를 확정할 수 없는 브라우저 세션은 `no_visit` 그룹으로 분리하여 사이트 라벨의 신뢰성을 유지한다. 이로써 하나의 세션은 프로토콜·종단점 정보(5-튜플)와 다단계 라벨을 동시에 갖추게 되어, 분석 목적에 따라 원하는 **granularity**의 부분 집합을 손쉽게 구성할 수 있다. 공개본에서는 이러한 라벨 정의를 별도 문서 없이 확인할 수 있도록, 응용 단위·프로세스 단위 라벨의 전체 목록과 각 라벨의 세션 수를 일자·호스트별 통계 파일(`_exp_summary.json`)로 함께 제공하며, 관점 트리와 원본 세션 경로의

대응 관계는 `manifest` 파일에 기록하여 라벨 부여 결과를 세션 단위로 역추적할 수 있도록 하였다.

3.7. 트래픽 분포

PRISM-2026은 7대 호스트의 3일간(2026.07.10 - 07.12) 연속 수집을 통해 총 1,320,366개의 세션으로 구성된다. 전송 계층 기준으로는 UDP가 55.4%, TCP가 44.6%를 차지하여, QUIC 및 DNS로 대표되는 UDP 트래픽의 비중이 TCP를 상회한다. 이는 최근 웹·응용 트래픽이 HTTP/3(QUIC)로 이전하는 경향을 반영한다.

표 3. PRISM-2026 상위 10개 라벨 분포 (프로세스 단위)
Table 3. Top-10 Label Distribution of PRISM-2026 (Process-level)

순위	라벨	세션 수	비율
1	svchost.exe	518,750	39.29%
2	chrome.exe	244,876	18.55%
3	servicemapcollector.exe	58,320	4.42%
4	claude.exe	49,188	3.73%
5	notion.exe	43,601	3.30%
6	code.exe	38,072	2.88%
7	remoting_host.exe	34,483	2.61%
8	codex.exe	33,041	2.50%
9	nexonplug.exe	29,287	2.22%
10	msedgewebview2.exe	28,076	2.13%

라벨 분포는 실제 호스트 환경의 특성을 그대로 드러내며 강한 불균형을 보인다. 표 3에서 보듯 다수의 Windows 서비스를 대리 실행하는 `svchost`가 전체 세션의 39.3%로 가장 큰 비중을 차지하고, 그 뒤를 브라우저(Google Chrome, 18.6%)가 잇는다. 상위 2개 라벨이 전체의 57.8%, 상위 5개가 69.3%, 상위 10개가 81.6%를 차지하는 반면, 하위 다수의 라벨은 수십 개 이하의 세션만을 가진다. 실제로 가장 큰 라벨(518,750 세션)과 가장 작은 라벨(2 세션)의 차이는 약 26만 대 1에 이르며, 프로세스 단위 108종 가운데 6종은 세션 수가 10개 미만이다. 이러한 심한 클래스 불균형 분포는 특정 응용만을 격리 수집하는 기존 데이터셋과 달리, 실사용 호스트에서 동시에 발생하는 트래픽을 그대로 담아낸 결과이며, 소수 클래스 학습이라는 현실적 난제를 함께 제공한다는 점에서 의의가 있다.

3.8. 암호화 비율

PRISM-2026이 실제 응용 트래픽을 반영하는지 확인하기 위해, 세션의 5-튜플에 나타난 서비스

표 4. PRISM-2026 서비스 포트 기반 암호화 트래픽 분류

Table 4. Service-Port-based Classification of Encrypted Traffic in PRISM-2026

구분	대표 포트	세션 수	전체 대비	DNS 제외 대비
암호화 (TLS/QUIC/SSH)	443, 22, 8443, 993...	471,927	35.7%	55.8%
평문(DNS 포함)	53, 80, 137-139 ...	513,672	38.9%	-
DNS(제어)	53	473,976	35.9%	-
비DNS 평문	80 등	39,696	3.0%	4.7%
기타/독자 프로토콜	-	334,767	25.4%	39.6%
전체	-	1,320,366	100%	-

포트를 기준으로 암호화 트래픽의 비중을 분석하였다. 표 4과 같이 TLS/QUIC(443), SSH(22) 등 암호화 프로토콜로 식별되는 세션이 전체의 35.7%를 차지하며, 이 중 단일 포트로는 HTTPS/QUIC(포트 443)가 35.0%로 콘텐츠 트래픽의 대부분을 구성한다. 평균으로 분류되는 세션은 38.9%이나, 그 대부분(전체의 35.9%)은 이름 확인을 위한 DNS(포트 53) 제어 평면 트래픽이며 실제 평문 콘텐츠 트래픽(HTTP 등)은 5% 미만에 불과하다. 나머지 25.4%는 게임·실시간 통신·내부 수집기 등이 사용하는 비표준 고포트 트래픽으로, 표준 포트만으로는 암호화 여부를 단정하기 어려운 독자 프로토콜에 해당한다.

DNS 제어 트래픽을 제외한 응용·콘텐츠 세션만을 기준으로 보면, TLS/QUIC 암호화 트래픽이 55.8%로 과반을 차지하고 평문 트래픽은 4.7%에 그친다. 즉 PRISM-2026의 콘텐츠 트래픽은 사실상 암호화되어 있으며, 이는 페이로드 기반 분석이 어려운 현대 네트워크 환경을 충실히 반영한다. 이러한 특성은 본 데이터셋이 암호화 트래픽 분류·행위 분석 연구의 벤치마크로 적합함을 뒷받침한다.

IV. 실험 및 결과

본 장에서는 PRISM-2026이 트래픽 분류 모델 학습에 실효적으로 활용될 수 있는지를 검증하기 위해, 라벨 granularity가 서로 다른 두 라벨 집합(프로세스 단위 108종, 응용 단위 89종)에 대하여 대표적 트래픽 분류 모델인 ET-BERT, TSHIRT, 2D CNN, XGBoost 네 가지를 각각 학습·평가하였다. 이들은 각각 사전학습 트랜스포머(ET-BERT), 프로토콜 구조 인지 계층 표현 모델(TSHIRT), 바이트 이미지 기반 CNN(2D CNN), 트리 기반 앙상블(XGBoost)로서 서로 다른 입력 표현과 학습 방식

을 대표하며, 동일 데이터셋에서 표현 방식에 따른 성능 차이를 비교하기 위해 선정하였다. 성능은 정확도(Accuracy)를 기준으로 평가하였다.

4.1. 실험 환경

모델별 입력 표현과 학습 설정은 다음과 같다. ET-BERT[9]는 세션 페이로드 선두 784바이트를 토큰화한 시퀀스(seq_length 128)를 입력으로 하며, 사전학습된 암호화 트래픽 표현 모델을 미세조정하는 방식으로 배치 크기 32, 3 epoch, 학습률 기본값으로 학습하고 데이터를 7:1:2(train:valid:test)로 분할하였다. 2D CNN은 Ethernet·IP·전송 계층 헤더를 제거한 784바이트를 28×28 그레이스케일 이미지로 변환·정규화(0 - 1)하여 입력하며, 두 개의 합성곱 계층(32·64 채널, 5×5 커널)과 두 개의 완전연결 계층으로 구성된다. 배치 크기 3,000, 300 epoch, Adam(학습률 1e-3)으로 학습하고 데이터를 8:2로 분할하였다. TSHIRT[10]는 패킷을 프로토콜 디섹터(Wireshark)로 파싱하여 세션 - 패킷 - 계층 - 필드의 계층 구조로 토큰화하는 구조 인지 표현 학습 모델이다. 각 필드를 하나의 토큰으로 두고, 필드 내부 바이트를 순서 불변 통계(mean·max 풀링)로 요약한 콘텐츠 임베딩에 위치 임베딩과 필드 길이 임베딩을 결합한다. 세션당 최대 256개 토큰, 필드당 최대 1,400바이트를 사용하며, 6개 계층·8개 어텐션 헤드·은닉 차원 528의 트랜스포머 인코더를 백본으로 한다. 계층 서플 사전학습(필드·계층·패킷 순서 복원)으로 구조적 의존성을 학습한 뒤, 계층적 어텐션 풀링과 레벨 간 게이팅으로 미세조정하며, AdamW(학습률 1e-5), 배치 크기 32로 학습하고 데이터를 8:2로 분할하였다. 전처리 단계에서 TCP 3-way handshake 패킷과 DNS·NTP·SSDP 등 시스템 수준 트래픽을 제거하고 응용 계층(L7) 정보만을 입력으로 사용한다. XGBoost는 원시 바이트가 아닌 플로우 단위 통계 피처를 입력으로 사용한다. 각 세션(플로우)에서 (i) 패킷 크기·패킷 간 도착시간(IAT)·페이로드 길이에 대해 방향(양방향·정방향·역

방향)과 관측 패킷 수(전체 및 상위 5·10·15·20·25·30개)별로 15종 통계량을 산출한 945개 피쳐, (ii) 패킷 크기 시퀀스와 IAT 시퀀스를 각각 앞 100개까지 취한 200개 시계열 피쳐, (iii) 첫 패킷의 IP/TCP/UDP 헤더 필드 30개, (iv) 초당 바이트·패킷 수, 방향별 비율, 지속시간 등 플로우 요약 21개, (v) TCP 플래그 유형별 카운트 9개, 그리고 전송 프로토콜 구분 1개를 결합하여 총 1,206개의 수치 피쳐를 구성하였다. 이때 세션의 출발지·목적지 IP와 포트, 파일명에서 파생된 식별 정보는 모델이 중단점 신원을 암기하여 성능이 비현실적으로 과대평가되는 것(정보 누출)을 방지하기 위해 피쳐에서 명시적으로 제외하였다. 모델은 트리 수 600, 최대 깊이 8, 학습률 0.1, subsample 0.8, colsample_bytree 0.8, L2 정규화 계수 1.0의 히스토그램 기반 GBDT[11]로 학습하였으며, 데이터를 8:2로 계층적 분할(stratified)하였다. 네 모델은 표본이 10개 미만인 클래스를 제외한 세션 집합을 사용하되, 데이터 분할은 ET-BERT의 경우 7:1:2(train:valid:test), TSHIRT, 2D CNN 및 XGBoost의 경우 8:2(train:test) 계층적 분할을 적용하였으며, 입력 표현은 모델 특성에 따라 상이하다.

4.2. 기준 성능 평가

표 5. PRISM-2026에 대한 기준 모델 분류 성능
Table 5. Baseline Model Classification Performance on PRISM-2026

모델	라벨 단위	클래스 수	Test Accuracy
ET-BERT	프로세스	108	0.7730
ET-BERT	응용	89	0.7715
TSHIRT	프로세스	108	0.7942
TSHIRT	응용	89	0.7915
2D CNN	프로세스	108	0.8291
2D CNN	응용	89	0.8288
XGBoost	프로세스	108	0.9173
XGBoost	응용	89	0.9164

표 5는 네 모델의 두 라벨 집합에 대한 분류 정확도를 정리한 것이다. 네 모델 모두 프로세스 단위(108종)와 응용 단위(89종)에서 거의 동일한 정확도를 보여, 라벨 정의 방식보다 모델의 입력 표현과 특징 설계가 성능을 좌우함을 확인할 수 있다. 정확도 기준으로는 플로우 통계 피쳐를 사용한 XGBoost가 두 라벨 집합에서 각각 0.9173, 0.9164로 가장 높았고, 2D CNN(약 0.829), TSHIRT(약 0.79), ET-BERT(약 0.77)가 그 뒤를 이었다. 이는 세밀하게 설계된 플로우 수준 통계 특징이 본 데이터셋의 프로세스·응용 구분에 효과적이었음을 시사한다.

다만 PRISM-2026은 소수의 라벨에 세션이 크게 편중된 심한 클래스 불균형 특성을 가지므로, 정확도만으로는 표본이 적은 클래스의 분류 성능을 온전히 반영하기 어렵다. 따라서 클래스 불균형을 고려한 Macro-F1 등 보조 지표를 병행하여, 표본이 적은 클래스에 대한 성능까지 함께 평가할 필요가 있다.

이러한 결과는 PRISM-2026이 트랜스포머·구조 인지 모델·CNN·트리 기반 앙상블 등 서로 다른 계열의 분류 모델 모두에서 유의미한 학습이 가능한 라벨 품질을 갖추었으며, 프로세스·응용 두 라벨 granularity가 서로 다른 분석 관점을 제공하는 다단계 벤치마크로 활용될 수 있음을 뒷받침한다.

V. 결 론

본 연구에서는 패킷에서 라벨을 추론하는 기존 방식의 한계를 극복하기 위해, 트래픽을 발생시킨 엔드포인트로부터 라벨을 직접 수집하는 자동 라벨링 프레임워크 PRISM을 활용하여 다계층 네트워크 트래픽 데이터셋 PRISM-2026을 구축하였다. PRISM-2026은 7대의 실사용 호스트에서 3일간 수집한 총 1,320,366개의 세션으로 구성되며, 각 세션에 프로세스·서비스·사이트의 다계층 라벨과 5-튜플 기반 중단점 정보를 함께 부여함으로써 분석 목적에 따라 원하는 granularity의 라벨 집합을 유연하게 구성할 수 있도록 하였다.

5.1. 논의 및 한계

데이터 분석 결과 PRISM-2026은 실사용 환경의 특성을 그대로 반영한다. 전체 트래픽의 상당수가 TLS/QUIC 등으로 암호화되어 있어 페이로드 기반 분석이 어려운 현대 네트워크 환경을 잘 담고 있으며, 소수 프로세스가 트래픽의 대부분을 차지하는 심한 클래스 불균형을 보여 소수 클래스 학습이라는 현실적 난제를 함께 제공한다. 다만 본 연구에는 몇 가지 한계가 있다. 첫째, 원본 패킷 페이로드가 외부 스토리지에 보관되어 있어 암호화 여부 분석을 5-튜플(서비스 포트) 기반 추정으로 수행하였으므로, TLS/QUIC 핸드셰이크에 대한 심층 패킷 검사(DPI)를 통한 정밀화가 필요하다. 둘째, 수집 기간과 호스트 수가 제한적이어서 라벨 분포가 특정 사용 환경에 편중될 수 있다. 셋째, 기준 모델 실험은 대표적인 분류기에 한정하였고 평

가 지표 또한 정확도(Accuracy) 중심으로 제시하였다. 본 데이터셋은 상위 2개 라벨이 전체의 57.8%를 차지하고 프로세스 단위 108종 중 6종은 세션 수가 10개 미만인 심한 불균형 구조를 가지므로, 정확도만으로는 소수 클래스의 분류 성능이 충분히 드러나지 않는다. 따라서 Macro-F1·Weighted-F1 및 클래스별 precision/recall을 병행한 평가가 요구된다. 또한 현재의 계층적 무작위 분할은 동일 호스트·인접 시간대의 유사 세션이 학습 집합과 시험 집합에 함께 포함될 수 있어 성능이 낙관적으로 추정될 여지가 있으므로, 날짜별 또는 호스트별로 분리한 분할 평가를 병행할 필요가 있다. 넷째, 본 연구는 세션에 라벨이 부여된 비율인 매칭률(98.85%)을 보고하였을 뿐, 부여된 라벨이 실제와 일치하는지를 나타내는 라벨 정확도는 정량적으로 검증하지 못하였다. 특히 본 연구에서 새로 도입한 서비스 단위 라벨과 SNI 기반 사이트 단위 라벨은 표본에 대한 수작업 대조를 통해 정확도와 오류 유형을 제시할 필요가 있다. 이는 데이터셋의 신뢰성과 직결되는 사안으로 후속 연구에서 우선적으로 다룰 예정이다.

5.2. 결론 및 향후 연구

기준 모델 실험에서 ET-BERT, TSHIRT, 2D CNN, XGBoost를 프로세스(108종)·응용(89종) 두 라벨 집합에 대해 평가한 결과, 트리 기반의 XGBoost가 각각 0.9173, 0.9164의 정확도로 가장 우수하였고, 2D CNN(약 0.829), TSHIRT(약 0.79), ET-BERT(약 0.77)가 그 뒤를 이었다. 이는 라벨 정의 방식보다 모델의 표현력과 특징 설계가 성능을 좌우함을 보여주며, PRISM-2026이 다양한 분류 모델의 벤치마크로 활용될 수 있음을 뒷받침한다. 향후에는 (1) TLS/QUIC 핸드셰이크 기반 DPI를 통한 암호화 트래픽 정밀 분석, (2) 수집 호스트·기간의 확대를 통한 라벨 다양성 및 대표성 강화, (3) 클래스 불균형을 고려한 소수 클래스 학습 기법의 적용을 통해 데이터셋과 분류 성능을 함께 발전시키고자 한다.

구축한 PRISM-2026 데이터셋은 개인정보 보호를 위한 익명화 처리를 거쳐 GitHub 저장소(<https://github.com/illiard1209/PRISM-2026-Dataset>)에서 공개한다. 공개본은 동일한 세션 집합을 세 가지 관점(sessions, byproduct, byproduct_process)으로 구성하여 제공하며, 일자·호스트별 라벨 통계(_exp_summary.json)와 관점별 디렉토리 구조와 원

본 세션 경로의 대응 관계를 기록한 manifest 파일을 함께 포함한다. 또한 배포 파일의 무결성을 확인할 수 있도록 SHA-256 체크섬을 제공한다. 개인 정보 보호를 위해 수집 기관 대역에 속하는 IPv4 주소는 1:1 전단사 방식으로 익명화하되 /24 서브넷 구조를 보존하였으며, 해당 처리는 디렉토리명, 파일명 및 패킷 IPv4 헤더에 일관되게 적용하였다. 이를 통해 후속 연구자가 PRISM-2026의 다계층 라벨 구조와 실제 네트워크 트래픽을 활용하여 다양한 트래픽 분류 및 분석 연구를 수행할 수 있도록 한다.

References

- [1] Guang Cheng and Song Wang, "Traffic classification based on port connection pattern," 2011 International Conference on Computer Science and Service System (CSSS), Nanjing, 2011, pp. 914-917, doi: 10.1109/CSSS.2011.5974374.
- [2] M. S. Raza, S. B. A. Kazmi, R. Ali, M. M. Naqvi, H. Fiaz and A. Akram, "High Performance DPI Engine Design for Network Traffic Classification, Metadata Extraction and Data Visualization," 2024 5th International Conference on Advancements in Computational Sciences (ICACS), Lahore, Pakistan, 2024, pp. 1-6, doi: 10.1109/ICACS60934.2024.10473274.
- [3] Moore, Andrew & Zuev, Denis. (2005). Internet traffic classification using Bayesian analysis techniques. Sigmetrics Performance Evaluation Review - SIGMETRICS. 33. 50-60. doi:10.1145/1064212.1064220.
- [4] Lotfollahi, Mohammad & Shirali hossein zade, Ramin & Jafari Siavoshani, Mahdi & Saberian, Mohammadsadegh. (2020). Deep Packet: A Novel Approach For Encrypted Traffic Classification Using Deep Learning. Soft Computing. 24. doi:10.1007/s00500-019-04030-2.
- [5] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun and A. A. Ghorbani, "Characterization of

Encrypted and VPN Traffic Using Time-Related Features," Proc. 2nd International Conference on Information Systems Security and Privacy (ICISSP), Rome, Italy, 2016, pp. 407-414, doi: 10.5220/0005740704070414.

- [6] I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," Proc. 4th International Conference on Information Systems Security and Privacy (ICISSP), Funchal, Portugal, 2018, pp. 108-116, doi: 10.5220/0006639801080116.
- [7] Wei Wang, Ming Zhu, Xuewen Zeng, Xiaozhou Ye and Yiqiang Sheng, "Malware traffic classification using convolutional neural network for representation learning," 2017 International Conference on Information Networking (ICOIN), Da Nang, Vietnam, 2017, pp. 712-717, doi: 10.1109/ICOIN.2017.7899588.
- [8] J.-M. Kim, Y.-S. Jang, G.-M. Yu, S.-W. Nam, J.-S. Kim, and M.-S. Kim, "Socket-based automatic labeling framework for process-level network traffic classification," in Proc. KNOM Conf., 2026.
- [9] Lin, Xinjie & Xiong, Gang & Gou, Gaopeng & Li, Zhen & Shi, Junzheng & Yu, Jing. (2022). ET-BERT: A Contextualized Datagram Representation with Pre-training Transformers for Encrypted Traffic Classification. 633-642. 10.1145/3485447.3512217.
- [10] G.-M. Yu, Y.-S. Jang, J.-S. Kim, S.-W. Nam, J.-M. Kim, Y.-S. Choi and M.-S. Kim, "Beyond Byte-Level Modeling: Structure-Aware and Adaptive Traffic Classification for Encrypted Networks," Electronics, vol. 15, no. 9, 1828, 2026, doi: 10.3390/electronics15091828.
- [11] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Proc. 22nd ACM SIGKDD International Conference on Knowledge

Discovery and Data Mining, San Francisco, CA, USA, 2016, pp. 785-794, doi: 10.1145/2939672.2939785.

김 지 민 (Ji-Min Kim)



2026 : 고려대학교 컴퓨터융합소프트웨어학과 학사
2026 ~ 현재 : 고려대학교 컴퓨터소프트웨어학과 석박사통합과정
<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

유 경 민 (Gyeong-Min Yu)



2023 : 고려대학교 컴퓨터융합소프트웨어학과 학사
2026 : 고려대학교 컴퓨터융합소프트웨어학과 석사
2026 ~ 현재 : 고려대학교 컴퓨터소프트웨어학과 박사과정
<관심분야> 네트워크 트래픽 분류, 암호화 트래픽 분석, 딥러닝 기반 네트워크 트래픽 탐지

장 윤 성 (Yoon-Seong Jang)



공격 트래픽 탐지

2023 : 고려대학교 컴퓨터융합소프트웨어학과 학사
2023 ~ 현재 : 고려대학교 컴퓨터소프트웨어학과 석박사통합과정
<관심분야> 네트워크 트래픽 분류, 암호화 트래픽 분석,

남 승 우 (Seung-Woo Nam)



2023 : 고려대학교 컴퓨터융합소프트웨어학과 학사
2023 ~ 현재 : 고려대학교 컴퓨터소프트웨어학과 석박사통합과정
<관심분야> 트래픽 분석, 네트워크 보안, ZTA

김 주 성 (Ju-Sung Kim)



2023 : 고려대학교 컴퓨터융
합소프트웨어학과 학사

2023 ~ 현재 : 고려대학교
컴퓨터소프트웨어학과 석박
사통합과정

<관심분야> 네트워크 트래픽
분류, 암호화 트래픽 분석

백 의 준 (Ui-Jun Baek)



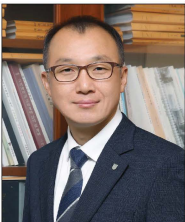
2018년 : 고려대학교 컴퓨터정
보학과 학사

2025년 : 고려대학교 컴퓨터정
보학과 박사

2025년 ~ 현재 : 한국과학기술
정보연구원(KISTI) 선임연구
원

<관심분야> 블록체인 거래 모니터링, 네트워크 관
리 및 보안, 트래픽 모니터링 및 분석

김 명 섭(Myung-Sup Kim)



1998년 : 포항공과대학교 전자
계산학과 학사

2000년 : 포항공과대학교 전자
계산학과 석사

2004년 : 포항공과대학교 전자
계산학과 박사

2024년~2006년 : Dept. of

ECS, Univ of Toronto Canada

2006년 ~ 현재 : 고려대학교 컴퓨터정보학과 교수

<관심분야> 네트워크 관리 및 보안, 트래픽 모니
터링 및 분석, 단일미디어 네트워크