논문 19-44-01-15 The Journal of Korean Institute of Communications and Information Sciences '19-01 Vol.44 No.01
https://doi.org/10.7840/kics.2019.44.1.106

SDN 환경의 트래픽 분류를 위한 특징 선택 기법

임 환 희•, 김 경 태*, 이 병 준**, 윤 희 용°

Feature Selection Method for the Classification of Traffic in SDN

Hwan-hee Lim•, Kyung-tae Kim*, Byung-jun Lee**, Hee-yong Youn°

요 약

최근, 수많은 IoT(Intetnet of Things) 기기가 급속히 확산되면서 엄청난 양의 트래픽이 발생하고 있다. 이와 같은 많은 양의 트래픽은 네트워크의 전송 속도를 느리게 할 뿐만 아니라 높은 QoS(Quality of Service)를 보장하기 어렵게 만든다. 이러한 문제를 해결하기 위해 SDN(Software-Defined Networking) 기술이 도입되었는데, SDN은 제어 단과 데이터 처리 단을 분리하여 네트워크를 효율적으로 관리할 수 있으므로 대규모 네트워크 환경에서 효율적으로 사용된다. 본 논문에서는 SDN 환경에서 다양한 트래픽을 적절히 분류할 수 있는 새로운 특징 선택 알고리즘을 제안한다. 기존의 filter 기반 방식에서는 특징을 평가하는 평가 기준이 부족하여 특징의 수가 작게 되면 분류 정확도가 낮아지는 단점이 존재한다. 이러한 문제를 해결하기 위해 가중치 기반 chi2-square test 알고리즘을 도입한 새로운 특징 선택 기법을 제안한다. 세 가지 다양한 데이터 세트에 대한 실험 결과가 제안된 알고리즘이 기존에 널리 사용되는 두 개의 특징 선택 알고리즘에 비해 분류 정확도와 F1 점수가 훨씬 우수한 결과를 보인다.

Key Words : SDN, machine learning, feature selection, network traffic, classification accuracy

ABSTRACT

Nowadays, various Internet of Things (IoT) devices are widely used, generating tremendous amount of traffic. They not only slow the transmission speed of the network, but also make it difficult to guarantee high QoS. The SDN technology had been introduced as a solution to these problems. SDN is used in a large-scale network environment because it can efficiently manage the network by separating the control plane and data plane. This paper proposes a novel feature selection algorithm to efficiently classify various internet traffics in SDN environment. The shortcoming of the existing filter-based feature selection approach is low classification accuracy if the number of features is small. In order to solve such problem, a novel feature selection scheme is proposed which employs the weight-based chi2-square test algorithm. The experimental results with three datasets reveal that the proposed scheme outperforms two popular existing feature selection algorithms in terms of classification accuracy and F1 score.

※ 본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 정보통신·방송연구 개발 사업(No. 2016-0-00133, 초연결 IoT 노드의 군집 지능화를 통한 Edge Computing 핵심 기술 연구), SW중심대학지원사업(2015-0-00914), 한국연구재단 기초연구사업(No.2016R1A6A3A11931385, 실시간 공공안전 서비스를 위한 소프트웨어 정의 무선 센서 네트워크 핵심기술 연구, 2017R1A2B2009095, 실시간 스트림 데이터 처리 및 Multi-connectivity를 지원하는 SDN 기반 WSN 핵심 기술 연구), BK21PLUS 사업의 일환으로 수행되었음.

• First Author : (ORCID:0000-0003-0426-5965)SungKyunkwan University Department of Electrical and Computer Engineering, lhh423@skku.edu, 학생회원

° Corresponding Author : (ORCID:0000-0003-0860-6990)SungKyunkwan University Department of Electrical and Computer Engineering, youn7147@skku.edu, 정회원

* (ORCID:0000-0001-6475-6905)SungKyunkwan University College of Software, kyungtaekim76@gmail.com

** (ORCID:0000-0001-6645-3780)SungKyunkwan University Department of Electrical and Computer Engineering, byungjun@skku.edu, 학생회원

논문번호 : 201811-372-B-RE, Received November 28, 2018; Revised January 2, 2019; Accepted January 3, 2019

Ⅰ. 서 론

최근 인터넷 기술의 발달로 다양한 Internet of Things(IoT) 기기들이 여러 방면으로 활용되고 있다. 예를 들어, 웨어러블 기기, 자율 주행 자동차의 센서, 스마트폰 등이 서로 연결되어 물체를 조작하며 다양한 방식으로 사용되고 있다. 결과적으로, 네트워크 트래픽이 기하급수적으로 증가하고 있으며, 특히 인터넷 콘텐츠를 다루는 대형 데이터 센터의 출현으로 기존 네트워크 환경에서의 가장 큰 문제 중의 하나인 대량 트래픽에 기인한 네트워크 지연이 발생하고 있다[10]. 이러한 문제를 해결하기 위해 SDN(Software-Defined Networking) 기술이 도입되었는데, SDN은 제어 단과 데이터 처리 단을 분리하여 네트워크를 효율적으로 관리할 수 있어서 대규모 네트워크 환경에서 효율적으로 사용된다.

한편, 많은 양의 트래픽을 효율적으로 관리하는 방법이 필요한데, 이를 위해서는 악의적인 공격을 탐지하고 빠른 전송 경로를 보장해야 한다. 네트워크 트래픽 증가에 따른 문제를 해결하기 위한 방법 중의 하나로 네트워크 트래픽에 대한 효율적 관리가 연구되고 있는데, 네트워크 트래픽의 특징을 추출하여 트래픽을 분류하는 방법이 주를 이루고 있다[1,11]. 여기서는 네트워크 트래픽의 특징을 간략화해서 추출함으로써 분류 모델링을 최적화하고 분류의 효율성을 높이는데, 트래픽 관리에 매우 효과적이라고 알려져 있다.

네트워크 트래픽 관리를 위해 다양한 방법이 채택될 수 있는데, 이 중 특징 선택을 기반으로 하는 방법론이 매우 효율적이며, 다음과 같은 장점이 있다. 첫째, 테스트 데이터 세트의 정확성을 높이고 데이터 세트를 학습할 때, overfitting 문제를 방지할 수 있다. 두 번째, 분류에 필요한 컴퓨팅 리소스를 줄이고 저장 공간을 최소화할 수 있다. 마지막으로, 예측 모델의 해석 가능성을 향상시킬 수 있다. 기존의 특징 선택 알고리즘은 일반적으로 filter, wrapper, embedded 방법으로 구분된다. filter 방법은 시간 복잡도가 작기 때문에 특징을 선택하는데 짧은 시간이 걸린다. 그러나, 데이터 세트가 작으면 분류와 관련된 특징이 제거될 수 있는 단점이 있다. 다음으로 wrapper 방법은 분류 알고리즘을 사용하여 최상의 특징 집합을 학습하는 방법이다. 정확한 특징 집합을 만들 수 있으나 모든 특징 모델을 학습하기 때문에 느리다는 단점이 있다. 마지막으로, embedded 방법은 wrapper 방법보다는 빠르지만 특정 학습 알고리즘에 의존하기 때문에 정확성이 떨어질 수 있다.

본 논문에서는 SDN 환경에 적합한 새로운 효율적인 특징 선택 방법론을 제안한다. 제안된 특징 선택 알고리즘에서는 우선 SDN 컨트롤러의 Packet-in message를 이용하여 트래픽 데이터를 가져오고 이를 분석하여 특징을 선택하는데, filter 기반 방식에 chi2-square test[8]를 적용하며 새로운 가중치를 부여하는 방식을 사용한다. 이는 기존의 filter 기반 방식의 단점인 데이터 세트가 충분하지 않을 경우, 분류와 관련이 있는 특징이 제거되는 단점을 보완하기 위함이다. 결과적으로, 세 가지 실제 데이터 세트에 대한 시뮬레이션을 통하여 제안된 알고리즘이 두 개의 널리 사용되는 filter 기반 방식들과 비교하여 분류 정확도와 F1 점수가 높다는 것을 보여 준다. 본 논문의 주요 결과는 아래와 같다.

- 네트워크 트래픽의 특징 수가 작아도 분류 정확도가 높은 새로운 특징 선택 기법이 제안되며, 특징 수를 줄임으로써 계산의 복잡도를 감소시킨다.
- 기존의 특징 선택 기법은 상관관계가 높은 특징을 처리하는 것에 한계가 있는데, 특징 선택 과정에서 평가 기준 및 새로운 가중치를 적용함으로써 높은 상관관계를 가지는 특징에 대해보다 더 효율적으로 처리할 수 있다.
- 제안된 특징 선택 기법은 SDN 환경 등과 같은 대규모 네트워크에서의 효율적 트래픽 관리를 지원할 수 있다.

본 논문의 구성은 다음과 같다. 1장 서론에 이어 2장에서는 SDN 기술과 트래픽 특징 선택 기술에 대해 기술한다. 3장에서는 본 논문에서 제안된 특징 선택 기법을 소개하며, 4장에서는 제안된 특징 선택 알고리즘의 성능을 시뮬레이션을 통해 평가하고 기존 방식과 비교한다. 마지막으로, 5장에서는 결론을 맺는다.

Ⅱ. 관련연구

2.1 특징 선택

특징 선택은 원본 데이터에서 최상의 성능을 보여줄 수 있는 특징 집합을 찾는 것으로[2] 세 가지 대표적인 특징 선택 기법이 있는데, filter 방법, wrapper 방법, embedded 방법이다[2-5]. 최근에는 다른 특징 선택 알고리즘들을 조합한 hybrid 방법[19,20]이 연구되고 있다.

Filter 방법은 통계 지표를 사용하여 각 특징의 순위를 매긴다. 일반적으로 전처리 단계에서 사용되며, 기계 학습 알고리즘과는 독립적이다. 일반적으로 결과 변수와의 상관관계를 분석하는데 사용하는데, FCBF

(Fast Correlation-Based Filter), mRMR (Minimum Redundancy Maximum Relevance) 알고리즘, chi2-square test, information gain 등이 있다.

카이 제곱 검정(Chi2-square test)을 이용한 특징 선택[8]은 클래스에 대하여 카이 제곱 통계를 계산하여 트래픽 특징 c_i와 클래스 t 사이의 독립성을 측정하는데, 아래의 카이 제곱 공식을 이용하여 계산된다.

$$\frac{X^2(t,c_i)=\sum[P(t,c_i)P(\bar{t},\bar{c_i})-P(t,\bar{c_i})P(\bar{t},c_i)]^2}{P(t)P(\bar{t})P(c_i)P(\bar{c_i})} \quad (1)$$

정보 이득(Information Gain)을 이용한 특징 선택 기법[7]에서는 X와 Y 사이의 정보가 엔트로피와 조건부 엔트로피에 대한 종속성을 측정하는데 사용된다. 특징 데이터 집합에서 특정 데이터의 출현으로 얻은 비트 수는 특정 데이터가 포함되어야 하는 범주를 예측하는데 사용된다. X와 Y는 함께 공유되는 정보의 양을 측정하고 정보 이득은 다음과 같이 계산된다.

$$\begin{aligned} G(t) =& -\sum_{i=1}^{n} P_r(c_i)\log P(c_i) \\ &+ P(t)\sum_{i=1}^{n} P(c_i|t)\log P(c_t|t) \\ &+ P(\bar{t})\sum_{i=1}^{n} P(c_i|\bar{t})\log P(c_i|\bar{t}) \end{aligned} \quad (2)$$

고속 상관 기반 필터(FCBF) 알고리즘을 이용한 특징 선택 기법[6]은 대칭 불확도를 평가하는데 사용되는데, 선형 상관관계뿐만 아니라 다른 상관관계도 감지할 수 있다. 여기서 대칭 불확실성은 엔트로피 개념을 사용하여 특징 간의 상관관계를 측정하며, 다른 값(x_i)을 가질 수 있는 특징 F가 주어지면, X의 엔트로피는 다음과 같이 계산된다.

$$H(X) = -\sum_i P(x_i)\log_2(P(x_i)) \quad (3)$$

그리고 서로 다른 특징 Y에 대한 X의 엔트로피는 다음과 같이 계산된다.

$$H(X|Y) = -\sum_j P(x_j)\sum_i P(x_i|y_j)\log_2(P(x_i|y_j)) \quad (4)$$

$P(x_i)$는 특징 X에 대한 사전 확률이며, $P(x_i|y_j)$는 특징 Y에 대한 X의 사후 확률이다. 그리고 정보 이득은 다음과 같이 계산된다.

$$IG(X|Y) = H(X) - H(X|Y) \quad (5)$$

마지막으로 X와 Y의 대칭 불확실성은 다음과 같이 계산된다.

$$SU(X,Y) = 2\left[\frac{IG(X|Y)}{H(X)+H(Y)}\right] \quad (6)$$

$SU(X,Y)$가 1이면 X와 Y는 완벽한 상관관계를 나타내며, $SU(X,Y)$가 0이면 X와 Y는 상관관계가 아님을 나타낸다.

최소 중복성 최대 연관성(minimum redundancy – maximum relevance; mRMR) 알고리즘을 이용한 특징 선택 기법[14]에서 선택 척도는 가장 유용한 특징의 부분 집합을 선택하기 위해 두 집합의 상호 정보를 측정한다. 서로 다른 특징들 F_a와 F_b 사이의 상호 정보 I를 이용하여 계산하며, 최소 중복성 W는 아래와 같이 계산된다.

$$W = \frac{1}{|s|^2}\sum_{F_aF_bINS} I(F_a,F_b) \quad (7)$$

최대 관련성을 이용하면 서로 다른 특징 와 클래스 X 사이의 상호 정보 I 를 사용하여 관련성이 높은 특징 집합을 추출할 수 있으며 아래와 같은 식으로 계산된다.

$$V = \frac{1}{|s|}\sum_{F_aINS} I(F_a,X) \quad (8)$$

위의 두 조건을 결합하면 상호 정보 차이(MID)와 상호 정보 지수(MIQ)를 나타낼 수 있는데, 이 두 개의 값은 아래와 같이 계산된다.

$$MD = \max(V - W) \quad (9)$$

$$MQ = \max(V/W) \quad (10)$$

Wrapper 방법은 가장 이상적인 특징 조합을 나타내는데 다양한 특징 조합으로 수행되며 학습이 완료된 모델 중 가장 분류 정확도가 높은 조합이 선택된다. Wrapper 방법에는 SVM(Support Vector Machine), Naïve Bayesian, Decision Tree, Genetic Algorithm 등이 있다.

SVM을 이용한 특징 선택 기법[9]은 gradient

descent를 통해 일반화 경계를 최소화하는 것이다. 얼굴 인식, 보행자 검출 및 DNA 마이크로 배열 데이터 분석 등에서 기존의 알고리즘보다 분류 정확도가 높으나, 분류 시간이 많이 걸리는 단점이 있다.

Embedded 방법은 모델의 학습 및 선택 프로세스에서 가장 좋은 특징을 선택한다. Wrapper 방법은 학습 후에 모든 특징을 이용해 가장 분류 정확도가 높은 특징을 선택하지만, embedded 방법은 학습 과정에서 최적화된 특징을 선택한다.

Hybrid 방법으로서 filter 방법과 wrapper 방법을 결합하는 방법이 주로 연구되고 있는데, 지역 검색을 통해 모집단 기반의 최적화를 진행하여 최적의 특징을 선택한다[21]. 예를 들어, 두 가지 방법을 결합한 특징 선택 기법[19]에서는 중복되거나 관련성이 없는 특징을 제거하기 위해 filter 방법(F1 점수와 정보 이득)과 wrapper 방법 (Sequential Floating Search Method)을 이용하여 특징을 선택할 수 있다. 그러나 wrapper 방법과 결합함으로써 모든 특징 조합을 평가하기 때문에 분류 시간이 많이 걸리는 단점이 있다.

2.2 Software-Defined Networking

최근 클라우드 컴퓨팅 서비스 및 소셜 네트워크 서비스가 등장하고 있다. 이러한 서비스는 기존의 IP 네트워크를 주로 사용하며, 그 결과 트래픽이 폭발적으로 증가한다. 기존의 네트워크는 복잡한 구조와 다양한 트래픽 패턴에 대한 비효율성 등 여러 문제를 가지고 있는데, 이를 해결하기 위해 SDN이 개발되었다. SDN은 네트워크 모니터링 및 관리를 용이하게 함으로써 네트워크 성능을 향상시키기 위한 목적으로 널리 보급되었다. 또한, SDN 기술은 하드웨어 중심 네트워크 환경을 소프트웨어 중심 네트워크로 전환했으며, 네트워크 데이터 평면과 제어 평면을 분리하고 다양한 상황에 따라 다르게 제어한다. SDN의 구조는 그림 1에 표시된 Application 계층, Control 계층, Infrastructure 계층으로 구분된다[17]. Application 계층과 Control 계층에는 상위 계층에서 동작하는 네트워크 컨트롤러와 응용 프로그램이 있는데, 본 논문에서는 Control 계층에서 네트워크 트래픽을 가져오고 Application 계층에서 네트워크 트래픽의 특징들을 선택하고 분류하는 방안을 연구한다[23]. Infrastructure 계층에서 일부 기기는 네트워크 통신을 담당한다.

SDN을 채택하여 많은 성과를 이루고 있지만, 트래픽 분류와 같은 네트워킹 분야는 아직 초기 단계이다[24]. 최근에 SDN 환경에서 트래픽 분류를 위한 많은 연구가 이루어지고 있으며, 주로 트래픽 분류를 위한

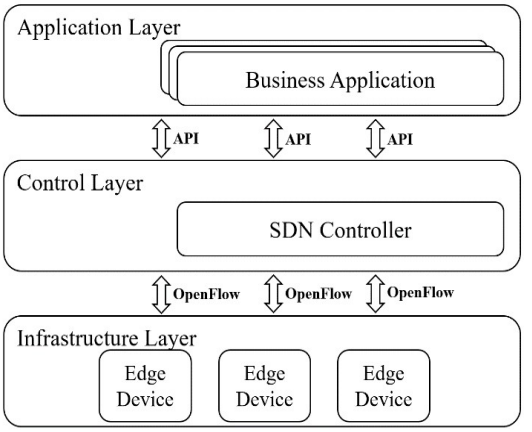

그림 1. SDN의 구조
Fig. 1. Structure of SDN.

아키텍처 연구가 진행되고 있다.

[1]에서는 SDN 환경에서 트래픽 분류를 위해 패킷을 Control 계층을 통해 Application 계층에 전송하면 Control 계층의 컴퓨팅 리소스가 소모되어 네트워크 성능이 저하되는 단점을 해결하는 아키텍처를 제안하고 있다. [23]에서는 SDN 환경에서 정확한 트래픽 분류를 위해 Principal Component Analysis(PCA) 알고리즘과 Genetic Algorithm(GA) 알고리즘을 이용하여, 네트워크 flow 특징들을 식별하고 선택하는 기법을 제안하고 있다. [25]에서는 네트워크 유해 트래픽 탐지를 위해 유해 트래픽 탐지 기술을 개발하고, 이를 기반으로 네트워크 특징을 식별 및 선택을 하는 기법을 제안하고 있다.

위의 많은 연구에도 불구하고, SDN은 트래픽 분류에 적합한 특징을 선택하기 위한 충분한 툴을 제공하지 않는다. 따라서, 본 논문에서는 SDN 환경에서 효율적인 트래픽 관리를 위한 특징 선택 기법을 제안한다.

Ⅲ. 본 론

기존의 특징 선택 기법은 특징의 수가 작으면 분류 정확도가 낮으며, 특징의 수가 많으면 알고리즘의 동작 시간이 길어지는 단점이 있다. 특히, 기존의 filter 방법에서는 특징을 평가하는 평가 기준이 부족하여 특징의 수가 작게 되면 분류 정확도가 낮아지는 단점이 존재한다. 이러한 문제를 해결하기 위해 기존의 filter 방법인 mRMR 알고리즘[14]에 가중치 기반 chi2-square test 알고리즘[8]을 도입한 새로운 특징 선택 기법을 제안한다. 다양한 환경에서의 시뮬레이션이

본 논문에서 제안된 기법이 기존의 특징 선택 알고리즘들보다 분류 정확도와 F1 점수가 높은 것을 보여준다.

3.1 제안된 특징 선택 알고리즘

제안된 기법의 첫 번째 단계에서 SDN 컨트롤러에서 packet-in message를 이용하여 인터넷 트래픽에 대한 정보를 얻는다[18]. 이 단계에서는 control 계층의 SDN컨트롤러를 이용한다. 두 번째 단계에서는 추출한 인터넷 트래픽 정보의 특징을 선택한다. 이 단계는 SDN 컨트롤러의 API를 이용하여 application 계층에서 수행된다. 세번째로 네트워크 트래픽의 특징 데이터를 mRMR알고리즘과 chi2-square test 알고리즘을 실행시켜 얻은 통계 지표를 이용하여 정렬하는데, 각각의 알고리즘에서 나온 특징들에 대해 가중치를 부여한 후 더해서 값이 높은 순서대로 정렬하여 선택한다. 제안된 알고리즘은 기존의 mRMR 알고리즘의 단점인 특징을 평가하는 평가 기준을 chi2-square test 알고리즘으로 보완하고 마지막으로 분류 알고리즘을 사용하여 분류 정확도와 F1 점수를 제고한다. 본 논문에서는 Naive Bayesian 알고리즘[22]을 사용하는데, 이 알고리즘은 클래스의 특정 변수가 다른 변수와 독립적이라고 가정한 베이즈 정리를 기반으로 조건부 확률을 계산하고 이를 기반으로 분류를 하며 분류 속도 및 계산 속도가 매우 빠른 장점이 있다. 그리고 Naive Bayesian 알고리즘은 특징 간의 관계를 무시하기 때문에 상관관계가 높은 특징이 존재하면 에러율이 높아지지만 본 논문에서 제안된 특징 선택 알고리즘을 이용해 상관관계가 높은 특징들을 제거하여 효과적으로 분류를 할 수 있다. 그림 2는 전체적인 흐름도를 보여준다.

mRMR 알고리즘은 위의 식 (9), (10)을 통해 특징들의 점수를 매겨, 순위를 정렬한다. 그러나 mRMR 알고리즘은 중복성과 관련성만을 가지고 특징들을 평가하기 때문에 최적의 특징을 선택하기가 어렵다. 따라서, 부족한 평가 기준 보완하기 위해 chi2-square test 알고리즘과 가중치를 추가한다. 아래의 수식은 mRMR 알고리즘에서 추출된 특징을 점수별로 정리된 결과 특징들에 대해 가중치를 부여하는 계산식이다. f_i는 추출된 특징이며, w_i는 전체 특징의 개수이며, i는 순위별로 정렬된 특징의 점수이다.

$$S_i = f_i + (w_i - i) \tag{11}$$

Chi2-square test 알고리즘은 각 클래스와 각 특징

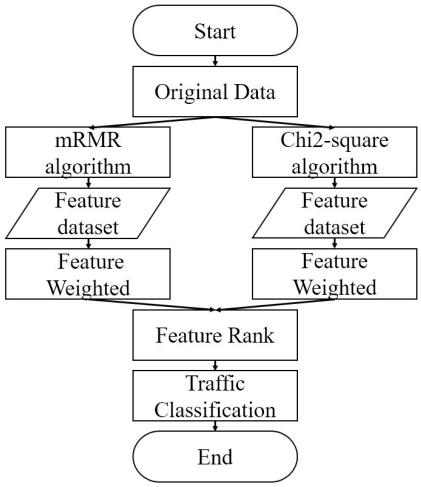

그림 2. 제안된 알고리즘의 흐름도
Fig. 2. Flow chart of proposed algorithm.

간의 독립성을 측정한다. 독립성은 수식 (1)과 같이 계산되며 수식 (1)에서 나온 결과 특징들에 대해서 가중치를 부여하며, 아래의 수식은 Chi2-square test 알고리즘에서 추출된 특징을 점수별로 정리된 결과 특징들에 대해 가중치를 부여하는 계산식이다. f_j는 추출된 특징이며, w_j는 전체 특징의 개수이며, i는 순위별로 정렬된 특징의 점수이다.

$$S_j = f_j + (w_j - i) \tag{12}$$

각 특징들을 mRMR 알고리즘과 chi2-square test 알고리즘을 통해 가중치를 부여한 점수를 매긴 뒤, 점수가 높은 순으로 정렬을 한다. 정렬된 특징들을 1순위부터 마지막 순위까지 가중치를 넣어 합산을 하는데, 계산식은 아래와 같다.

$$\widehat{S_a} = S_i + S_j \tag{13}$$

그리고 가중치가 더해진 특징별로 다른 알고리즘의 매칭되는 특징들과 더해 다시 순위별로 정렬을 한다. 마지막으로 다시 정렬된 특징을 선택하여 분류 알고리즘을 수행한다. 예를 들어, KDDCup99 데이터 세트의 특징인 duration, protocol_type가 mRMR 알고리즘에서 1순위, 3순위로 나왔을 때 chi2-square test 알고리즘에서 2, 3순위로 나왔다고 가정하자. mRMR 알고리즘에서 식 (11)을 이용하여 duration과 protocol_type에 결과 값 가중치인 41과 39를 더한다. 이때 chi2-square test 알고리즘은 식 (12)를 이용하여

Algorithm 1. The Proposed algorithm

```
Start episode
input number of features
Extract traffic feature sets F_a, F_b
Extract traffic feature set classes X
for 1, feature F_a , ++
        I_a = mutual_Information (F_a, X)
        relevance = (1/number of X)×sum(I_a)
        for 1, feature F_b, ++
                redundancy = [1/(number of X)^2]×sum(I_a)
        end for
        results_1[F_a] = relevance − redundancy
end
for 1, number of features, ++
        feature sort(results_1[F_a].score)
        add results_1[F_a] weighted(number of feature − i)
end
for 1, feature F_a, ++
        C_a = Chi2-square_test (F_a, X)
        results_2[F_a] += C_a
end
for 1, number of features, ++
        feature sort(result_2[F_a].score)
        add results_2[F_a] weighted(number of feature − i)
end
for 1, number of features, ++
        results_f[F_a].score = result_1[F_a] + result_2[F_a]
end
if results_f[F_a].score == results_f[F_b].score
        c1 = results_1[F_a].score − results_2[F_a].score
        c2 = results_1[F_b].score − results_2[F_b].score
        select smaller c1, c2
features sort
```

결과 값 가중치인 40과 39를 더한다. 식 (13)을 이용하여 앞서 계산된 두 개의 가중치를 합산한 뒤 최종적인 순위를 정렬한다. 아래는 제안된 알고리즘의 pseudo code이다.

Ⅳ. 실험결과

4.1 데이터 세트

이 절에서는 제안된 특징 선택 알고리즘의 성능 평가를 위한 데이터 세트를 소개한다. 성능평가에는 세 가지 데이터 세트가 사용되며 각 데이터 세트의 특징은 아래와 같다.

첫째, KDDCup99[15] 데이터 세트는 침입 탐지 연구 평가를 위해 9주간 시뮬레이션을 통하여 MIT 링컨 연구소의 1998 DARPA 침입 탐지 평가 프로그램에서 생성되었다. 이 데이터 세트는 41개의 특징으로 구성되어 있으며, 약 4,370,000개의 샘플로 구성된다. 이 데이터 집합의 목적은 컴퓨터의 공격 또는 비정상적인 동작에 해당하는 각 flow를 식별하는 것이다. 본 논문에서는 성능평가를 위해 KDDCup99 데이터 세트의 10% 버전을 사용했다. 이 데이터 세트는 트래픽의 탐지 및 분류 기법에 활발하게 사용되고 있다. 최근 인터넷 침입 탐지에 대한 관심이 높아졌기 때문에 이 데이터 세트를 사용하는 것이 실용성이 높다. 아래의 표 1은 KDDCup99 데이터 세트의 특징들을 나타낸다[26].

둘째, NSL-KDD[16] 데이터 세트는 KDDCup99 데이터 세트를 보완하기 위해 2009년 Tavallaee가 제안한 데이터 세트이다. KDDCup99 데이터 세트에는 많은 중복된 데이터 (약 75%)가 포함되어 있다. 따라서, NSL-KDD 데이터 세트는 중복 데이터를 삭제하고 추가 데이터를 생성하여 제안된 데이터 세트이다. 중복된 데이터가 많으면 자주 나오는 flow에 의해 학습될 가능성이 높으며, 테스트 프로세서의 평가 결과에도 영향을 미칠 수 있다. NSL-KDD 데이터 세트의 특징은 KDDCup99 데이터 세트와 같다.

셋째, Moustafa와 Slay가 UNSW-NB15 데이터 세트를 제안했다[12,13]. UNSW-NB15 데이터 세트는 기존의 침입 탐지 데이터 세트인 KDDCup99와 NSL-KDD 데이터 세트의 문제점을 보완한 데이터 세트이다. 기존의 데이터 세트는 생성된 지 오랜 시간이 지나 정상적인 동작과 유사하게 동작하는 현대적인 네트워크 flow가 없는 문제가 있다. UNSW-NB15 데이터 세트는 이러한 문제를 해결하고 비정상적인 flow와 정상적인 flow의 혼합되어 있다. 그리고, 최신의 공격 유형 9가지를 추가하고 네트워크 트래픽을 위한 49개의 호스트 흐름 기반 특징과 패킷 헤더를 포함하고 있다. 아래의 표 2는 UNSW-NB15 데이터 세트의 특징이며, 표 3은 이 세 가지 데이터 세트의 요약이다.

4.2 시뮬레이션 결과

그림 3부터 그림 8은 제안된 알고리즘을 기존의 널리 사용되고 있는 특징 선택 알고리즘인 mRMR 알고리즘과 L1 regularization의 성능평가 결과를 비교한

표 1. KDDCup99 데이터세트의 특징
Table 1. Features of KDDCup99.

Index	features	description
1	duration	Length of connection
2	protocol_type	type of protocol
3	service	destination of connection
4	flag	status of connection
5	src_bytes	data bytes from src to dst
6	dst_bytes	data bytes from dst to src
7	land	If the connection is made on the same host/port 1, otherwise 0
8	wrong_fragment	number of wrong fragments
9	urgent	number of urgent packets
10	hot	nunmber of hot indicator
11	num_failed_logins	number of failed login attempts
12	logged_in	1 if successfully logged in, otherwise 0
13	num_compromised	number of compromised states
14	root_shell	1 if root shell is obtained, otherwise 0
15	su_attempted	If the 'su root' command is attempted, it is 1; otherwise it is 0
16	num_root	number of root accesses
17	num_file_creations	Number of operation that create new files
18	num_shells	number of shell prompts
19	num_access_files	number of operations on access control files
20	num_outbound_cmds	number of outbound commands in an ftp session
21	is_hot_login	if the login is part of the hot list, it is 1; otherwise it is 0.
22	is_guest_login	if the login is guest login, then 1; otherwise, 0
23	count	number of connections to same host as the current connection at the specified interval
24	srv_count	number of connections to same service as current connection at the specified interval
25	serror_rate	% of connections with SYN errors
26	srv_error_rate	% of connections with SYN errors
27	rerror_rate	% of connections with REJ errors
28	srv_rerror_rate	% of connections with REJ errors
29	same_srv_rate	% of connections to the same service
30	diff_srv_rate	% of connections to different services
31	srv_diff_host_rate	% of connections to different hosts
32	dst_host_count	number of connections to the same destination
33	dst_host_srv_count	number of connections to the same destination that use the same service
34	dst_host_same_src_rate	% of connections to the same destination that use the same service
35	dst_host_srv_rate	% of connections to different hosts on the same system
36	dst_host_same_srv_port_rate	% of connections to a system with the same source port
37	dst_host_srv_diff_host_rate	% of connections to the same service coming from different hosts
38	dst_host_serror_rate	% of connections to a host with an S0 error
39	dst_host_srv_serror_rate	% of connections to a host and specified service with an S0 error
40	dst_host_serror_rate	% of connections to a host with an RST error
41	dst_host_srv_serror_rate	% of connections to a host and specified service with an RST error

표 2. UNSW-NB15 데이터세트 특징
Table 2. Features of UNSW-NB15.

Index	features	Index	features	Index	features	Index	features	Index	features
1	dur	10	sttl	19	djit	28	dmean	37	is_ftp_login
2	proto	11	dttl	20	swin	29	trans_depth	38	ct_ftp_cmd
3	service	12	sload	21	stcpb	30	response_body_len	39	ct_flw_http_mthd
4	state	13	dload	22	dtrcpb	31	ct_srv_src	40	ct_src_ltm
5	spkts	14	sloss	23	dwin	32	ct_state_ttl	41	ct_srv_dst
6	dpkts	15	dloss	24	tcprtt	33	ct_dst_ltm	42	is_sm_ips_ports
7	sbytes	16	sinpkt	25	synack	34	ct_src_dport_ltm	43	attack_cat
8	dbytes	17	dinpkt	26	ackdat	35	ct_dst_sport_ltm		
9	rate	18	sjit	27	smean	36	ct_dst_src_ltm		

표 3. 세 가지 데이터 세트 정보
Table 3. Information of three data sets.

Data set	Feature	Lable	Number of sample
KDDCup99	41	23	494,022
NSL-KDD	41	2	148,518
UNSW-NB15	43	2	257,674

것이다. 실험 환경은 Intel(R) Core (TM) i5-4440 CPU @ 3.1GHz, 램은 16GB이며, Windows 10 Pro 환경에서 성능을 측정하였다. 성능평가의 측정 기준은 분류 정확도와 F1 점수이다. 분류 정확도는 데이터가 분류되는 정확도이며, F1 점수는 정확도만 아니라 리콜 비율을 포함하는 정보이다.

그림 3과 4는 KDDCup99 데이터 세트를 사용하여 본 논문에서 제안된 알고리즘과 기존의 특징 선택 알고리즘인 mRMR 알고리즘과 L1 regularization의 성능평가 결과를 비교한 것이다. KDDCup99 데이터 세트는 41개의 특징을 가지고 있기 때문에 5개의 단위로 35개의 특징을 선택하여 분류 정확도와 F1 점수를 측정했다. 분류 클래스는 U2R(buffer_overflow, loadmodule 등), R2L(ftp_write, guesspasswd 등), DoS(back, land 등), Probing(ipsweep, nmap 등), 정상 트래픽으로 구분되며, 제안된 기법이 기존 특징 선택 알고리즘들보다 분류 정확도와 F1 점수가 전체적으로 높은 것을 확인할 수 있다.

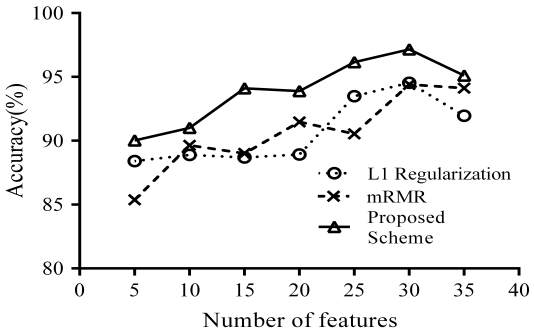

그림 3. KDDCup99 데이터 세트에 대한 분류 정확도 비교
Fig. 3. Classification accuracy of KDDCup99.

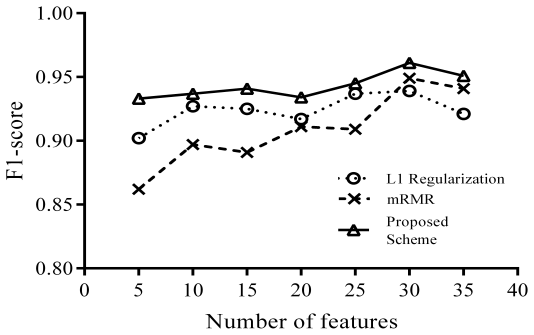

그림 4. KDDCup99 데이터 세트에 대한 F1-score 비교
Fig. 4. F1-score of KDDCup99.

그림 5와 그림 6은 NSL-KDD 데이터 세트에 대한 성능평가 결과 비교이다. NSL-KDD 데이터 세트는 41개의 특징을 가지고 있기 때문에 5개의 단위로 35개의 특징을 선택하여 분류 정확도와 F1 점수를 측정했다. 분류 클래스는 정상적인 트래픽과 비정상적인 트래픽으로 구분되며, 제안된 기법은 기존의 특징 선택 알고리즘보다 나은 결과를 보인다. L1 regularization은 특징의 수가 5와 10일 때 분류 정확도가 매우 낮은 것을 확인할 수 있다. 제안된 기법은 특징의 수가 적을 때에도 분류 정확도가 높으며, F1 점수에 관점에서 mRMR 알고리즘과 비슷한 점수를 보이지만 전반적으로 점수가 높은 것을 확인할 수 있다.

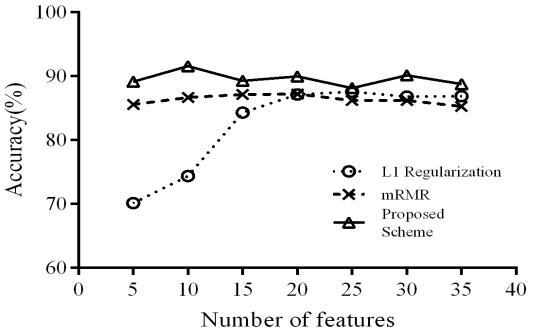

그림 5. NSL-KDD 데이터 세트에 대한 분류 정확도 비교
Fig. 5. Classification accuracy of NSL-KDD.

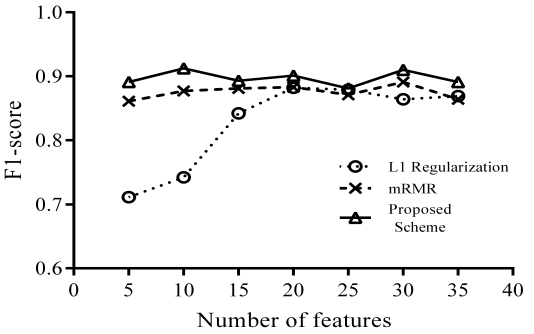

그림 6. NSL-KDD 데이터 세트에 대한 F1-score 비교
Fig. 6. F1-score of NSL-KDD.

그림 7과 그림 8은 UNSW-NB15 데이터 세트에 대한 비교결과이다. 분류 클래스는 정상적인 트래픽과 비정상적인 트래픽으로 구분되며, UNSW-NB15 데이터 세트는 43개의 특징을 가지고 있기때문에 5개의 단위로 35개의 특징을 선택하여 분류 정확도와 F1 점수를 측정했다. 앞의 두 가지 데이터 세트와 마찬가지로 제안된 기법이 기존 방식들에 비해 나은 성능을 보여주는 것을 확인할 수 있다.

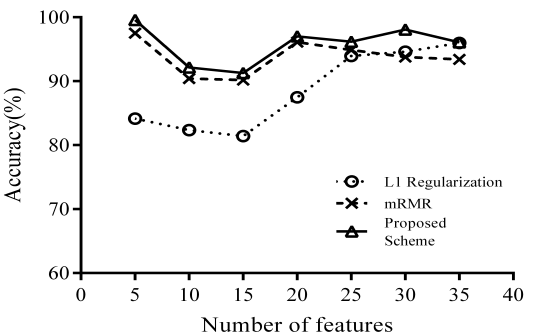

그림 7. UNSW-NB15 데이터 세트에 대한 분류 정확도 비교
Fig. 7. Classification accuracy of UNSW-NB15.

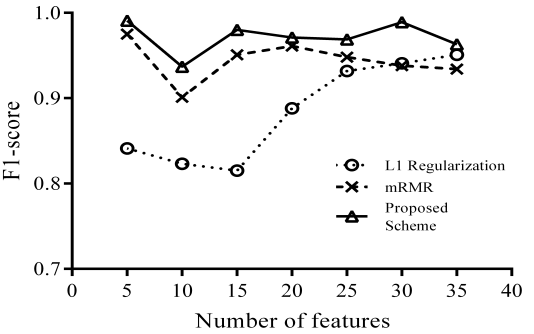

그림 8. UNSW-NB15 데이터 세트에 대한 F1-score 비교
Fig. 8. F1-score of UNSW-NB15.

V. 결 론

최근 네트워크가 점점 더 복잡해지고 관리하기가 어려워졌다. 그러나, 인공지능 및 네트워크 관리기술이 진화함에 따라 많은 트래픽 관리 기술이 등장했다. 그 중 하나는 트래픽을 효율적으로 분류하기 위해 특징 선택 알고리즘을 사용하여 인터넷 트래픽을 분류하는 기술이다. 본 논문에서는 SDN 환경에서 트래픽 관리를 용이하게 하는 특징 선택 알고리즘을 제안하였다. 제안된 기법은 정확한 분류를 보장하기 위해 의존성을 고려한 새로운 특징 선택 기법을 도입하고 분류와 관련 없는 특징을 제거한다. 성능을 증명하기 위해 세 가지 대표적인 데이터 세트를 사용하여 제안된 기법에 대해 폭 넓은 범위의 실험이 수행되었다. 실험 결과는 분류 정확도와 F1 점수가 기존의 두 가지 널리 사용되는 특징 선택 알고리즘보다 우수함을 보여준다.

향후 연구로는 실제 SDN 환경에서 Raspberry-Pi를 사용하여 트래픽 특징을 추출하고 테스트하는 테스트 베드를 구축할 것이다. 실제 인터넷 트래픽을 분류하기 위해서는 실시간 처리가 필요한데, 이와 관련해 OpenFlow 기능을 조사하고 실시간 인터넷 트래픽 분류 체계를 연구할 계획이다.

References

[1] M. Hayes, B. Ng, A. Pekar, and W. K. G. Seah, "Scalable architecture for SDN traffic classification," *IEEE System J.*, vol. 12, no. 99, pp. 1-12, Dec. 2018.

[2] I. Guyon and A Elisseeff, "An Introduction to Variable and Feature Selection," *J. Machine Learning Res.*, vol. 3, pp. 1157-1182, Mar.

2003.

[3] Y. Saeys, I. Inza, and P. Larranaga, “A review of feature selection techniques in bioinformatics,” *Bioinformatics*, vol. 33, no. 19, pp. 2507-2517, Aug. 2007.

[4] G. Chandrashekar and F. Sahin, “A survey on feature selection methods,” *Comput. & Electrical Eng.*, vol. 40, no. 1, pp. 16-28, Jan. 2014.

[5] M. Dash and H. Liu, “Feature selection for classification,” *Intell. Data Anal.*, vol. 1, no. 3, pp. 131-156, 1997.

[6] L. Yu and H. Liu, “Feature selection for high-dimensional data: A fast correlation-based filter solution,” *Int. Conf. Machine Learning*, pp. 856-863, Washington DC, US, Aug. 2003.

[7] L. E. Raileanu and K. Stoffel, “Theoretical comparison between the gini index and information gain criteria,” *Annals of Math. and Artificial Intell.*, vol. 41, no. 1, pp. 77-93, May 2004.

[8] X. Jin, A. Xu, R. Bie, and P. Guo, “Machine learning techniques and chi-square feature selection for cancer classification using SAGE gene expression profiles,” *Int. Workshop on Data Mining for Biomed. Appl.*, pp. 106-115, Singapore, Apr. 2006.

[9] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, “Feature selection for SVMs,” *Int. Conf. Neural Inf. Process. Syst.*, pp. 467-653, Denver, CO, US, Nov. 2000.

[10] S. Subhabrata, S. Oliver, and W. Dongmei, “Accurate, scalable in-network identification of P2P traffic using application signatures,” *Int. Conf. World Wide Web*, pp. 512-521, New York, NY, US, May 2004.

[11] J. Erman, M. Arlitt, and A. Mahanti, “Traffic classification using clustering algorithms,” *SIGCOMM workshop on Mining network data*, pp. 281-286, Pisa, Italy, Sept. 2006.

[12] N. Moustafa and J. Slay, “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” *MilCIS*, pp. 1-6, Canberra, Australia, Nov. 2015.

[13] N. Moustafa and J. Slay, “The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set,” *Inf. Secur. J.: A Global Perspective*, vol. 25, no. 1-3, pp. 18-31, Apr. 2016.

[14] H. Peng, F. Long, and C. Ding, “Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy,” *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 27, no. 8, pp. 1226-1238, Jun. 2005.

[15] *KDDCup99 Dataset*, Retrieved Aug. 2018, from http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[16] *NSL-KDD Dataset*, Retrieved Aug. 2018, from https://www.unb.ca/cic/datasets/nsl.html

[17] T. J. Park, S. S. Lee, and S. W. Shin, “A reflectornet based on software defined network,” *J. KICS*, vol. 39, no. 6, pp. 397-405, Jun. 2014.

[18] H. Xue, K. T. Kim, and H. Y. Youn, “Packet scheduling for multiple-switch software-defined networking in edge computing environment,” *Wireless Commun. and Mob. Comput.*, vol. 2018, pp. 1-11, Nov. 2018.

[19] H. Hsu, C. Hsieh, and M Lu, “Hybrid feature selection by combining filter and wrappers,” *Expert System with Appl.*, vol. 38, no. 7, pp. 8144-8150, Jul. 2011.

[20] M. Naseriparsa, A. Bidgoli, and T. Varaee, “A hybrid feature selection method to improve performance of a group of classification algorithm,” *Int. J. Comput. Appl.*, vol. 67, no. 17, pp. 28-35, May 2013.

[21] X. Liu, Y. Liang, S. Wang, Z. Yang, and H. Ye, “A hybrid genetic algorithm with wrapper-embedded approaches for feature selection,” *IEEE Access*, vol. 6, no. pp. 22863-22874, Mar. 2018.

[22] G. H. John and P. Langley, “Estimating continuous distribution in bayesian classifiers,” in *Proc. 11th Conf. Uncertainty in Artificial*

Intell., pp. 338-345, Morgan Kaufmann, San Mateo, 1995.

[23] A. S. da Silva and C. C. Machado, "Identification and selection of flow features for accurate traffic classification in SDN," *IEEE 14th Int. Symp. Netw. Comput. and Appl.*, pp. 134-141, Cambridge, MA, US, Sept. 2015.

[24] B. Ng, M. Hayes, and W. K. G. Seah, "Developing a traffic classification platform for enterprise networks with SDN: Experiences & lessons learned," *IFIP Netw. Conf.*, pp. 1-9, Toulouse, France, May 2015.

[25] D. Jankowski and M. Amanowicz, "A study on flow features selection for malicious activities detection in software defined networks," *Int. Conf. Military Commun. and Inf. Syst.*, pp. 1-9, Warsaw, Poland, May 2018.

[26] D. D. Protic, "Review of KDD Cup'99, NSL-KDD and Kyoto 2006+ datasets," *Military Technical Courier / Vojnotehnicki Glasnik*, vol. 66, no. 3, pp. 580-596, Jul. 2018.

임 환 희 (Hwan-hee Lim)

2017년 8월 : 경상대학교 기계항공정보융합공학부 항공우주 및 소프트웨어공학전공 졸업

2017년 9월~현재 : 성균관대학교 전자전기컴퓨터공학과 석사과정

<관심분야> IoT, SDN, 머신러닝

김 경 태 (Kyung-tae Kim)

2003년 2월 : 수원대학교 컴퓨터과학과 졸업

2005년 8월 : 성균관대학교 정보통신공학부 컴퓨터공학과 석사

2013년 2월 : 성균관대학교 정보통신공학부 컴퓨터공학과 박사

2013년 9월~현재 : 성균관대학교 소프트웨어대학 연구교수

<관심분야> IoT, WSN, Edge Computing, 머신러닝, SDN

이 병 준 (Byung-jun Lee)

2009년 2월 : 한라대학교 컴퓨터공학과 졸업

2012년 2월 : 성균관대학교 전자전기컴퓨터공학과 석사

2014년 3월~현재 : 성균관대학교 전자전기컴퓨터공학과 박사과정

<관심분야> IoT, SDN, 머신러닝

윤 희 용 (Hee-yong Youn)

1977년 : 서울대학교 전기공학과 졸업

1979년 : 서울대학교 전기공학과 석사

1988년 : Univ. of Messachusetts at Amherst 컴퓨터공학과 박사

1988년 9월~1991년 5월 : Univ. of North Texas 조교수

1991년 6월~2000년 8월 : Univ. of Texas at Arlington 부교수

2000년 9월~현재 : 성균관대학교 컴퓨터공학과 교수 및 유비쿼터스 컴퓨팅기술연구소 소장

<관심분야> WSN, Edge Computing, IoT, SDN, 머신러닝