무선 네트워크에서의 효율적 트래픽 분류 기법 연구

이성진† 송종우†† 안수한‡ 원유집† 장재성‡‡
한양대학교 전자컴퓨터통신공학과†, 이화여자대학교 통계학과††
서울시립대학교 통계학과‡, SKT Access 망본부‡‡
{james†, yjwon†}@ece.hanyang.ac.kr, josong@ewha.ac.kr††
sahn@uos.ac.kr‡, jsjang@sktelecom.com‡‡

Efficient Traffic Classifier in Wireless Network

Seongjin Lee Jongwoo Song Soohan Ahn Youjip Won Jae-sung Chang
Dept. of Electronics and Computer Engineering , Hanyang University
Dept. of Statistics, Ewha Womans University
Dept. of Statistics, University of Seoul
Access Network Center, SKT

요 약

무선 인터넷의 구조적 특성상 한 셀에서 대역폭을 공유하고 그 안에서 각기 다른 QoS를 요구하는 서비스들이 한정된 자원을 사용한다. 트래픽의 변화와 패턴을 예측하기 위한 분석은 실제 서비스를 제공하기 전인 기획단계에서 매우 중요한 도구로 사용이 된다. 무선망의 트래픽을 예측하기 위해서는 유선 망의 분석과는 다른 방법이 필요하기 때문에 정확한 분류를 위해서 본 연구에서는 세션의 단위로 분석할 것을 제안한다. 또한 Classification and Regression Tree(CART)와 Support Vector Machine(SVM)의 두 개의 판별 분류 기법을 서로 비교하고 그 성능을 평가한다. 두 개의 판별 기법의 오차는 CART의 경우 0.0094 그리고 SVM의 경우 0.0089로 둘 다 우수한 성능을 보였지만 쉬운 결과 해석이 가능한 CART가 사용하기 용이함을 보인다.

1. 서 론

무선망의 경우 Access Network 를 포함한 네트워크 자원 증설은 큰 비용을 요구하기 때문에, 정교한 Capacity Planning, QoS Management, Traffic Engineering의 방법론 및 지원 인프라를 필요로 하고 있다. 특히, 무선 인터넷은 특성상 셀 내의 가입자들은 하나의 Broadband를 공유할 뿐만 아니라 여러 환경변수를 고려한 스케쥴링 알고리즘에 따라 자원이 배분되므로 가입자의 QoS 및 네트워크의 성능은 트래픽 변화에 민감하다. WiBro 망 기반 서비스와 EV-DO/HSDPA 망을 이용한 무선 모뎀 서비스의 출시와 더불어 무선 망을 이용한 데이터 서비스 형태는 점차 다양한 서비스를 출시하고 있으며 다양한 플랫폼을 지원하여 점차는 유선 인터넷 서비스와 비슷한 환경으로 발전할 것이다.

트래픽의 변화와 패턴예측을 위한 네트워크 상의 서비스 별 트래픽 분석은 네트워크 엔지니어링 및 서비스 기획 단계에서 매우 유용한 도구를 제공한다. 서비스 별 트래픽 분석을 하기 위해서는 서비스 별 분류가 가능해야 한다. 그러나 WiBro와 같은 개방형 망의 경우 서비스의 정확한 분류는 IP/Port 서비스 간의 Relational Database로 구축이 어렵기 때문에 사용자에 의해 발생하는 트래픽에 대한 서비스 분류가 불가능하다.

본 연구에서는 현재 운영 중인 WiBro 망에서 다양한 서비스를 이용할 때 발생하는 트래픽을 2007년 9월 27일부터 11월 30일까지 휴일을 제외하고 약 2달간 트래픽을 수집하였다. 수집한 트래픽 자료를 이용하여 Classification And Regression Tree (Cart) 기법과 Support Vector Machine (SVM) 기법을 이용하여 자료 안의 다양한 서비스들을 CART는 0.0094 SVM은 0.0089 의 낮은 오차율로 분류를 성공적으로 해내는 것을 보인다.

본 논문의 구성은 다음과 순서를 따른다. 먼저 관련연구를 2장에서 다루고 3장에서는 데이터 수집 환경과 수집된 데이터의 설명을 한다. 4장에서는 분석을 하기 위한 변수의 생성을 다루고 5장에서는 판별함수에 대한 자세한 설명을 한다. 6장에서는 판별함수를 통해 얻은 결과에 대한 분석과 7장에서는 향후 과제를 다루고 마지막으로 8장에서 결론을 맺는다.

2. 관련 연구

많은 연구들이 클러스터링을 이용하여 분류하는 것을 시도하였다 클러스터링에는 K-Means, DBSCAN, AUTOClass 등의 기법을 이용하여 분류한 연구가 있다[1, 2]. 클러스터링을 이용한 판별이 연산속도와 정확도 면에서 좋은 성능을 보이는 것을 알 수가 있다. 또한, 많은 연구가 Machine Learning을 이용하여 분류하는 것을 시도하였는데, Naïve Bayes Estimator를 사용한 것과 이 판별 함수의 변형을 사용한 연구들이 있다[3-5]. 학습을 하고 검증을 하는 부분으로 나누어서 특성을 스스로 배우고 그에 맞는 분류를 가능하게 하는 것으로 프로파일링이나 시그니쳐를 이용한 분류에 비해 자동화가 잘 되었다고 할 수 있다. 프로파일링 또는 시그니쳐를 이용한 분류 방법들도 큰 주류를 이루고 있다[2, 5-7] 이들의 장점이자 단점은 특정 서비스 어플리케이션에 대한 분석에 좋은 성능을 보인다는 것이다. 이는 많은 서비스 어플리케이션의 시그니쳐, 프로파일, 또는 핑거프린트 같은 정보를 갖고 있어야 한다는 것을 말하고, 만약 기존의 정보와 다르게 되면 무효화되기 쉽다. 특히 많은 어플리케이션들이 버전이 바뀌게 됨에 따라 보안과 개인정보 유지를 위하여 패킷을 암호화하게 되는데 그렇게 되면 이러한 시그니쳐를 얻기가 어려워진다.

분류를 하는데 있어 사용된 서비스 어플리케이션만 분석 해내는 차원을 떠나 포괄적으로 하나의 종단 노드 또는 다수의 종단 노드들의 사화과학적 역할을 규명하고 그 종단 노드의 역할이 생산자인지 소비자인지의 기능적 역할 그리고 어떤 어플리케이션을 사용하는가에 대한 분류까지 한 연구가 있다[8]. 이는 단순히 어플리케이션 분석 뿐만 아니라 사용자의 패턴까지 연구한 것이기 때문에 주목할 만하다. 더불어 많은 분류기법들이 놓치고 있는 것을 지목한 연구가 있는데[9] 이 연구에서는 소수의 세션 또는 플로우가 상당한 량의 바이트를 소모하고 있음을 놓치지 말아야 한다고 강조하였다. 다운로드 또는 VoD와 같은 서비스는 웹을 사용할 때와 달리 적은 수의 플로우가 생성이 되지만 그 량은 웹의 것에 비해 월등히 많다는 것을 말하고 있다.

트랜스포트 계층을 이용한 분류를 시도한 연구에서는 세션이라는 정보를 사용하여 분류를 시도하였지만 하나의 세션의 정의가 불명확하다[9]. 세션의 정의는 본 연구에서 좀더 명확하게 정의를 한다.

많은 종류의 분류 기법이 더 있고 각기 장단이 있는데 이러한 기법들을 분류한 연구가 있어 소개한다[10]. 이 연구는 체계적으로 구분을 하고 특징별로 나누어 놓았다. 특히 Tree를 기반으로 한 판별 함수들을 심도 있게 다루었다.

표 1 시스템 사양 및 어플리케이션

Unit	Description
CPU	XEON Irwindale 3.4 G / 2M x 2FSB 800M / 2 CPU
Memory	DDR2 400 1G x 2 (2Gb)
HDD	73G SCSI Ultra 320 10k
O/S	Linux 2.6
Database	MySQL 5.0.18
Statistics	R Project R-2.6.2

3. 데이터 수집 환경과 데이터 설명

본 연구에서 트래픽을 분류하기 위해서 사용된 시스템의 사양과 사용된 프로그램의 정보는 표 1에 나타나 있다.

트래픽의 수집은 다양한 서비스의 분류 가능성을 확인하고 또 분석하기 위해서 총 6개의 큰 서비스 군을 형성하였다. 2007년 9월 27일부터 같은 해 11월 30일까지 총 42일간 수집을 하였다. 하루에 수집한 데이터는 서비스 별로 6번에 걸쳐 10분씩 총 한 시간의 데이터를 수집하여서 36개의 트레이스 파일을 생성해냈다. 11월에 수집된 데이터는 단일 서비스를 사용하는 9월과 10월의 데이트들과 달리 수집 파일의 수를 줄이는 6개에서 3개로 줄이는 대신 수집 시간을 20분으로 늘리고 복합적인 서비스를 사용 하는 트레이스를 추가 하였다. 약 한달 동안 수집한 데이터 파일의 개수는 총 900개이다. 표 2 에서는 분류를 위한 총 7개의 서비스를 명시하였다. 각 서비스는 서로 다른 군집을 형성하며 사용자 측면에서 QoS와 사용 패턴, 그리고 Delay requirement 등에 있어서 다른 요구사항을 갖고 있는 서비스들인 동시에 많은 사용자가 무선 인터넷을 이용하여 접하는 주요 서비스들이다. 각 서비스의 이름과 사용된 컨텐츠의 종류의 이름은 편의 상 S1에서 S7로 표기하도록 한다.

수집한 트래픽을 분류를 하기 위해서는 기준이 필요하다. 기준을 정하는데 있어서 세가지 종류의 분류 계층이 있을 수 있는데 우선 패킷 레벨에서의 분류가 있고 두 번째로 플로우 레벨에서의 분류가 있으면

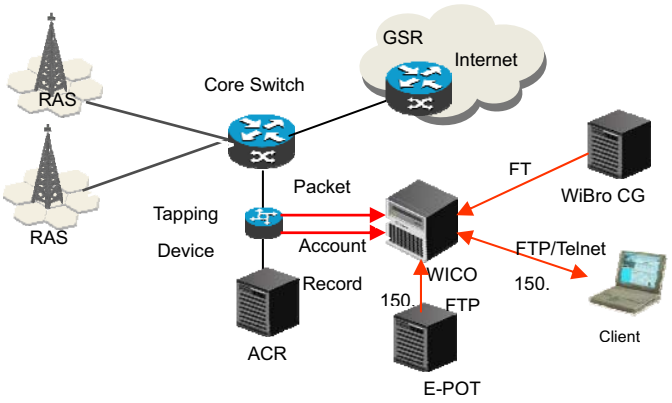

그림 1 WiBro 트래픽 수집 구간

표 2 수집 컨텐츠의 종류

이름	서비스 종류	컨텐츠 종류
S1	Download	Xtoc
S2	Online Game	Koonpa, Maple Story
S3	Streaming	FM Radio Streaming
S4	Uploads	Mail Uploads
S5	VOD	Daum UCC, YouTube
S6	VoIP	SkyPe. NateOn
S7	Web	Naver, Daum, Empas

마지막으로 세션 레벨에서의 분류가 있다. 패킷 레벨은 모든 패킷은 개별적으로 의미가 있다고 가정한다. 이 분석은 세밀하게 특성을 파악하기 위해서 사용할 수 있지만 연산을 하고 분류를 하는데 있어서는 오버헤드가 더 크기 때문에 제외를 한다. 플로우로 수집된 트래픽을 분류하는 기법은 범용적으로 사용되고 있으며 NetFlow와 같은 장비에서도 플로우 단위로 정보를 기록하고 있다. 그림 1에서는 WiBro 트래픽 수집 구간의 구성도를 나타내고 있다.

본 연구에서는 세션이라는 단위로 분석을 하며 세션은 그림 2로 나타낸다. 하나의 세션은 여러 개의 플로우로 구성이 되어 있다. 이 플로우는 일반적으로 Src/Dst IP, Src/Dst Port, 그리고 프로토콜이 같고 기준 시간 동안 해당 트래픽이 발생하지 않으면 하나의 플로우는 끝이 나는 것으로 정의 한다. 세션은 두 호스트가 서로 메시지를 교환할 때 생기는 트래픽으로써 식 1의 정의를 따른다.

이렇게 정의된 세션은 양방향에서 발생되는 플로우를 관리해주는 역할을 하기 때문에 해당 서비스가 발생시키는 ingress와 egress 트래픽에 대해서 효과적인 분석이 가능하게 한다. 일반적으로 ingress와 egress는 특성이 다른 것으로 알려져 있다. 특히 패킷 사이즈 분포와 패킷의 빈도 분포 그리고 패킷들의 도착 간격 프로세스 등이 다르다.

$$\{\beta.SrcIP=\gamma.DstIP \ \wedge \ \beta.SrcPort=\gamma.DstPort\} \vee\{\beta.SrcIP=\gamma.SrcIP \wedge \beta.SrcPort=\gamma.SrcPort\} \wedge\{\beta.Protocol=\gamma.Protocol\} \quad \text{식 1}$$

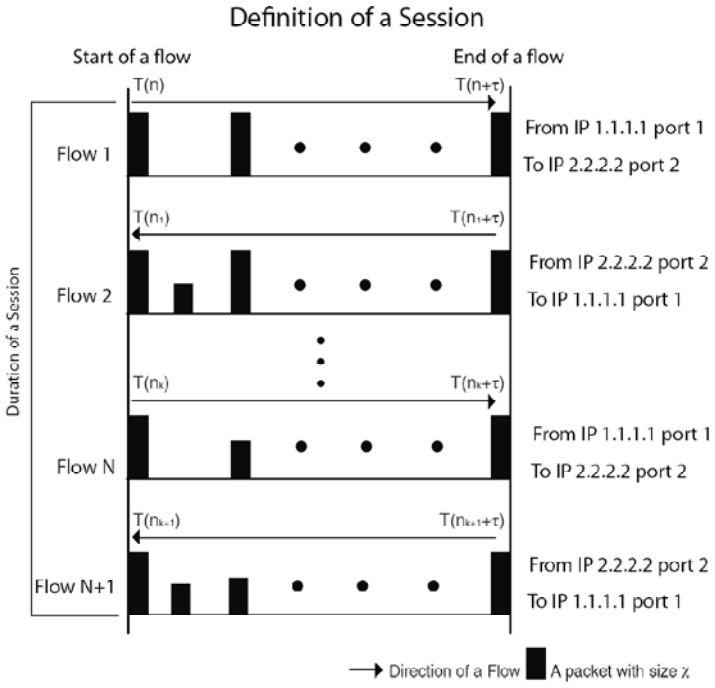

그림 2 세션의 정의

4. 분석 변수의 생성

본 연구의 목적인 서비스를 분류 해 낼 수 있는 판별함수(Classifier)를 생성하기 위해서 분석의 단위는 무선망에서 발생되는 세션으로 선택을 하였다. 분석을 하기 위해서 패킷 데이터를 이용해 세션 구성하고, 구성된 세션으로 다음과 같은 통계량을 얻어 낸다. 하나의 세션에 포함된 패킷의 수, 세션의 지속 시간, 패킷 크기들의 평균과 표준 편차, 패킷 간 도착 간격 평균과 표준 편차, 패킷 간 통계량 비율의 변수를 생성하였으며 명기한 변수들을 활용하여 판별 분석에 이용하였다. 또한, 분류를 위해서 세션을 이루는 uplink와 downlink를 표기 하였고, 해당 날짜와 서비스의 종류에 대한 내용도 표기 했다.

5. 분류 기법 연구: CART 와 SVM

분류를 위해서 사용된 기법은 CART와 SVM이다. 데이터 마이닝 기법으로서 CART는 간단하고 SVM은

표 3 CART를 이용한 판별 분석 결과
(70% Data for Training, 30% Data for Testing).
Misclassification rate for test set = 0.01

Appl. Pred	S1	S2	S3	S4	S5	S6	S7
S1	67	0	0	16	0	1	1
S2	0	213	0	0	0	1	0
S3	0	0	8	0	0	0	0
S4	0	0	0	45	1	0	0
S5	0	0	11	0	33	0	5
S6	0	20	0	0	1	131	1
S7	1	16	27	1	5	5	11484
Tot.	68	249	46	62	40	138	11491

표 4 SVM을 이용한 판별 분석 결과
(70% Data for Training, 30% Data for Testing).
Misclassification rate for test set = 0.01

Appl. Pred	S1	S2	S3	S4	S5	S6	S7
S1	48	0	0	0	0	0	0
S2	0	215	0	0	0	1	0
S3	0	0	12	0	0	0	0
S4	0	0	0	44	0	0	0
S5	0	0	11	0	33	0	0
S6	0	13	0	0	1	127	1
S7	20	21	23	18	6	10	11490
Tot.	68	249	46	62	40	138	11491

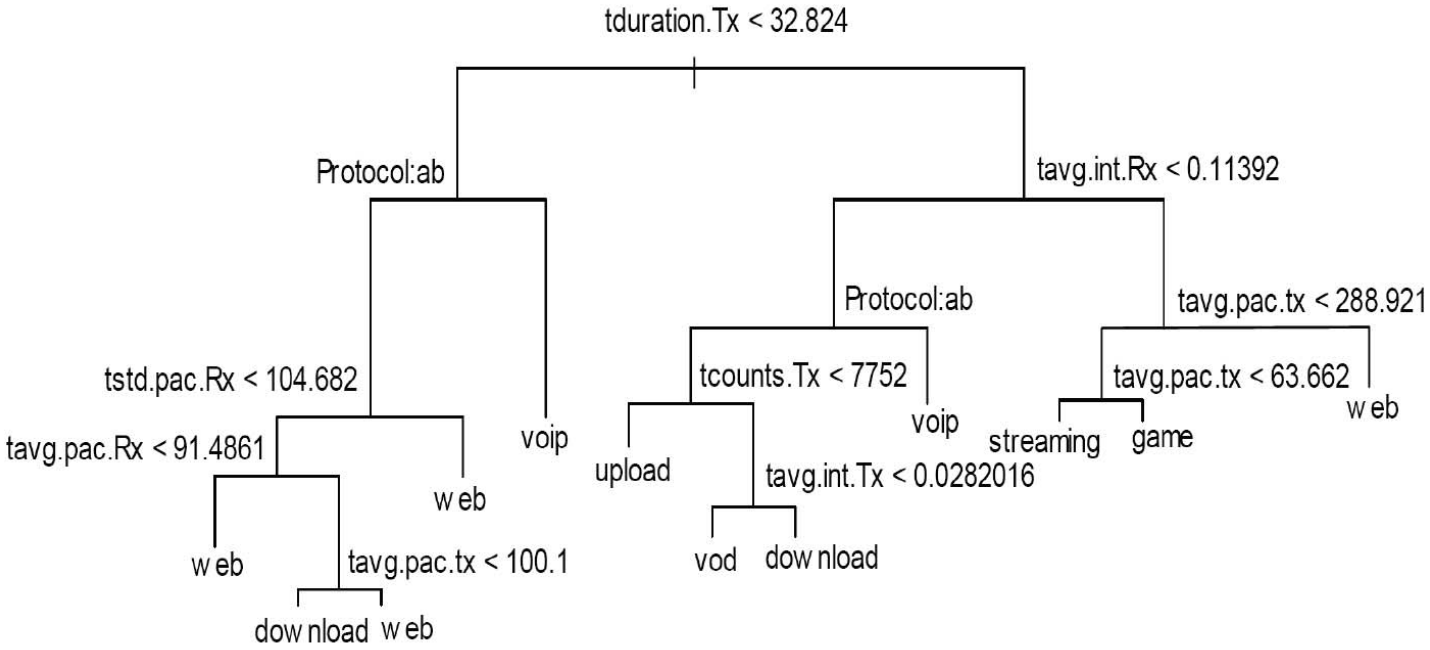

그림 3 CART를 사용한 분류의 예시

정확도가 높은 것으로 알려져 있다. 먼저 CART에 대해서 살펴보고 난 후에 SVM에 대해서 살펴 보도록 한다.

5.1. Classification and Regression Tree

CART는 Classifcation and Regression Tree의 약자로 분류가 간단하고, 그 결과를 해석하기가 쉬우며 좋은 예측 성능을 보인다[11]. 이름에서 알 수 있듯이 Tree 구조로 분할하며 판별식에 분석 변수들을 통해서 binary split을 한다. 데이터의 설명 변수들을 이용하여 오차가 최소가 되도록 분할을 정해 놓은 최소 문턱 값이 될 때까지 반복한다. 그러면 각 터미널 노드에 놓인 변수들이 적합하게 분류가 되었다고 한다. 이 방법의 목적은 분할을 한 결과들이 서로 동차가 되도록 만드는 것이다.

CART 분석 기법은 4단계로 이루어져 있다. 첫 단계 먼저 Tree를 만드는 것이다. Tree는 학습자료에 의해 생성된 예측된 클라스와 의사결정비용 매트릭스에 의해 나눠지는 노드들로 구성이 되어 있다. 두 번째 단계는 Tree가 더 자라는 것을 멈추는 것이다. 현재 클라스의

자료로 자식 노드를 구성할 수 없을 때까지 반복이 되기 때문에 멈추는 과정은 중요하다. 멈추는 시점은 각 노드에 값이 하나씩인 경우, 모든 자식 노드가 같은 확률 분포를 갖고 있어서 더 이상 나눠지지 않는 경우, 그리고 처음 설정할 때 Tree의 깊이를 설정한 경우 등으로 경우를 나눌 수 있다. 세 번째 단계는 나무의 가지치기를 통해서 Tree를 간단화 한다. 터미널 노드에서부터 시작하여 Tree의 복잡도가 그 노드를 잘랐을 때 어떤 값 δ 이하로 되지 않을 때 잘라낸다. 이 δ에 의해서 Tree는 간단화가 될 수가 있다. 마지막으로 최적화 된 Tree를 찾는 것이다. δ 값이 작아져서 Tree가 더 세분화 될 때는 과도하게 분류를 하여 오히려 정확도가 떨어질 수가 있다. 학습 데이터를 갖고 위의 과정들을 거칠 때 검증을 하기 위해서 데이터를 N 등분을 한다. N등분 중 한 섹션으로 학습을 하고 N-1등분의 데이터를 통해 검증을 하는 과정을 N번을 시도하여 정확도를 높이게 된다.

5.2. Support Vector Machine

SVM은 Support Vector Machine의 약자로 자주 사용되는 분류 방법 중에 하나인 이유는 이 방법이 갖는 우수한 성능 때문이다[12]. SVM은 판별을 하기 위해서 최적의 분리 경계면을 찾는다. 이 경계면은 각 데이터 값들은 최대로 멀리 떨어져 있게 되는 지점을 알려준다. 예를 들어 데이터 D가 두 클라스로 구성이 되어 있다고 하자. $D = \{(x_1, y_1), \dots, (x_n, y_n)\}, x \in \mathbb{R}^n, y \in \{-1, 1\}$ 이 데이터들은 어떤 하이퍼플래인 $\langle w, x_i \rangle + b = 0$에 존재한다. 변수 w, x가 $\min_i |\langle w, x_i \rangle + b| = 1$의 제한 조건이 있을 때 두 클래스를 나누는 하이퍼플래인은 $y_i[\langle w, x_i \rangle + b] \geq 1, i = 1, \dots, n$ 의 조건을 만족 해야 한다.

6. 성능 평가

주어진 자료를 CART와 SVM을 사용하여 분류할 때 정확도를 높이기 위해서 학습과 검증 부분을 70:30으로 나누어 한다. 이 때 전체 세션의 개수는 40131개 이었다. 이 과정을 10번을 진행하여 오판율의 평균을 내어 성능을 분석한다.

표 3에서는 CART를 이용하여 분석한 결과를 나타낸다. 가로축의 S1-S7은 서비스의 종류를, 세로축의 S1-S7은 판별을 정확도를 나타낸다. 서비스 S1을 CART를 이용했을 때 정확히 판별을 했는가를 보면, 총 68개의 세션들 중에서 1번을 제외하고는 모두 S1으로 판별해 낸 것을 확인 할 수가 있다. 표 3에서 대각선 방향의 값이 높을수록 판별함수의 성능이 좋다. 여기서 판별을 잘 하지 못한 경우는 S3의 경우 인데, Streaming 서비스를 Web 서비스로 잘못 분류를 한 경우이다. 나머지의 경우에서도 Web 트래픽으로 잘못 판별한 경우들이 있지만 S3을 제외한 경우는 그 수치가

표 5 CART 분류기법에서 사용된 변수의 의미

변수 명	의미
tduration	세션의 지속 시간
protocol:ab	세션의 프로토콜이 a:HTTP, HTTP > b:UDP, Etc. 의 구분
tstd.pac	한 세션에 속해 있는 패킷들의 표준편차
tavg.pac	한 세션에 속해 잇는 패킷들의 평균
tavg.int	한 세션에 속해 있는 패킷들의 평균 도착 간격 시간
tcounts	한 세션에 속해 있는 총 패킷의 개수
Rx, Tx	한 세션에 속해 있는 플로우들의 방향, 사용자측에서 서버측으로 전송되는 플로우를 Tx로 하고 그 반대를 Rx로 한다.

큰 의미를 갖지 않는다. 표 4은 SVM을 이용한 판별 분석의 결과이다. 여기서도 마찬가지로 Web 트래픽은 다른 서비스 분류들에서 나타나고 있다. Web 트래픽은 CART에서와 같이 다른 서비스들 보다 더 많은 세션 개수를 나타내고 있고 나머지 모든 서비스를 합한 것보다 더 많은 수의 세션을 보이고 있다. 그럼에도 불구하고 다른 서비스에서 Web 서비스로 오판을 하고 있는 것은 여러 서비스들이 Web서비스와 유사한 성격을 갖고 있기 때문이라 할 수 있다.

판별 분석의 오차는 CART의 경우 0.0094이고 SVM의 경우 0.0089이다. 수치상으로는 SVM이 매우 근소하게 우월한 것을 알 수가 있다. 분류 성능이 두 방법에서 비슷하다면 실용적인 면을 다루지 않을 수가 없다. 실용적인 면은 크게 세가지로 둔다면 먼저는 간단한 결과 해석, 두 번째는 처리속도, 그리고 마지막은 서비스 추가와 같은 환경 변화에 쉽게 적용 될 수 있는가에 대한 질문을 해야 한다. 이 세가지 질문에 하나씩 대답을 해보면 CART를 선택할 수가 있다. CART는 분류 항목의 크고 작음에 대해서 Binary Split을 통해 파티션을 한다. 그리고 최종적으로 파티션이 끝나게 되면 정확하게 분류된 결과를 나타내게 된다. 이것은 매우 큰 장점으로 인지적으로 이해하기가 쉽다. 두 번째로는 처리속도 측면에서 SVM은 정교한 판별식으로 높은 복잡도의 연산을 통해서 정확한 판별을 할 수 있고, 복잡도가 증가 할수록 더 높은 정확도를 얻을 수 있지만, CART의 경우 각 서비스의 특성에 대해서 크기 비교만을 통해서 간단히 분류하기 때문에 그 복잡도가 매우 낮은 것과 그 처리속도가 SVM에 비해 더 빠름을 알 수가 있다. 마지막으로

변화에 대한 적응 능력인데, 이것 역시 CART가 더 간단하게 대응할 수가 있다. 종합적으로 보면 비록 성능상 SVM이 우수한 것은 사실이지만 지속적인 업데이트와 복잡도 그리고 결과 해석 면에서 뛰어난 CART가 더 좋은 판별기의 기능을 하고 있음을 알 수가 있다.

그림 3에서는 CART를 사용한 분류가 얼마나 간단하게 진행되는지 그 단계를 도식화 하였다. 분류 기법에 사용된 모든 변수 명을 하나씩 설명하지는 않았지만 그림 3에서 사용된 변수들의 이름들은 표 5에 나와 있다. CART의 가장 끝은 노드까지 내려오면 해당 세션이 어느 서비스 군에 속하는지 자명하게 알 수 있게 해준다.

7. 결론

본 연구에서는 서비스 분류를 위하여 세션 단위의 트래픽 분류를 시도하였고, CART와 SVM을 이용한 트래픽 판별 함수를 개발하여 성능 검증을 하였다. 개발된 두 판별 함수의 오판별율은 1%에 불과하였고 이를 통해 두 방법 모두 매우 우수한 성능으로 분류를 할 수 있음을 보였다. 두 방법 모두 우수하기는 하나 연산량이나, 판별함수의 결과에 대한 이해, 그리고 사후 변화 적응력에 비교한다면 CART를 이용하는 것이 더 용이함을 보였다. 본 연구의 향후 지능형 트래픽 분석 솔루션 시스템의 구축을 위해서는 실제 네트워크에서 추출한 데이터를 통해 판별 함수를 적용하여 성능을 분석하여야 하고, 실제 트래픽에 나타난 서비스들의 패턴을 분석하는 과정을 거쳐야 한다. 또한, 전체 데이터를 판별함수에 적용하는 것은 많은 자원을 사용하기 때문에 표본 추출을 통한 판별 함수 연구가 필요하다.

8. Acknowledgements

본 연구는 MOST/KOSEF (Grant No. R11-2000-073-00000)의 SRC/ERC 프로그램의 지원을 받아 수행하였습니다.

Reference

[1] E. Jeffrey, M. Anirban, and A. Martin, "Byte me: a case for byte accuracy in traffic classification," in *Proceedings of the 3rd annual ACM workshop on Mining network data* San Diego, California, USA: ACM, 2007.

[2] E. Jeffrey, A. Martin, and M. Anirban, "Traffic classification using clustering algorithms," in *Proceedings of the 2006 SIGCOMM workshop on Mining network data* Pisa, Italy: ACM, 2006.

[3] W. M. Andrew and Z. Denis, "Internet traffic classification using bayesian analysis techniques," in *Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems* Banff, Alberta, Canada: ACM, 2005.

[4] W. Nigel, Z. Sebastian, and A. Grenville, "A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification," *SIGCOMM Comput. Commun. Rev.,* vol. 36, pp. 5-16, 2006.

[5] H. Patrick, S. Subhabrata, S. Oliver, and W. Dongmei, "ACAS: automated construction of application signatures," in *Proceeding of the 2005 ACM SIGCOMM workshop on Mining network data* Philadelphia, Pennsylvania, USA: ACM, 2005.

[6] X. Kuai, Z. Zhi-Li, and B. Supratik, "Profiling internet backbone traffic: behavior models and applications," in *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications* Philadelphia, Pennsylvania, USA: ACM, 2005.

[7] M. Roughan, S. Sen, O. Spatscheck, and N. Duffield, "Class of Service Mapping for QoS: A Statistical Signature based Approach to IP Traffic classification," in *IMC'04* Taormina, Sicily, Italy, 2004.

[8] K. Thomas, P. Konstantina, and F. Michalis, "BLINC: multilevel traffic classification in the dark," *SIGCOMM Comput. Commun. Rev.,* vol. 35, pp. 229-240, 2005.

[9] D. Hamza, V. Sandrine, and R. David, "A markovian signature-based approach to IP traffic classification," in *Proceedings of the 3rd annual ACM workshop on Mining network data* San Diego, California, USA: ACM, 2007.

[10] E. T. David, "Survey and taxonomy of packet classification techniques," *ACM Comput. Surv.,* vol. 37, pp. 238-275, 2005.

[11] L. Breiman, *Classification and Regression Trees*: Chapman & Hall/CRC, 1998.

[12] V. N. Vapnik, *The Nature of Statistical Learning Theory*: Springer, 2000.