

ML 알고리즘을 적용한 Internet Traffic 의 Application-level Classification

December 20, 2007

Jung Kwang Bon
jkbon@postech.ac.kr

DP&NM Lab., Dept. of CSE, POSTECH

Contents

1. 서론
2. 연구 목표
3. 관련 연구
4. 기존 연구의 문제점 & 해결 방안
5. ML 알고리즘 적용 방안
6. ML (J48)과 NGMON의 분류 비교 및 분석
7. 결론 및 향후 과제

서론 (1/2)

❖ Application Traffic Classification

- 트래픽을 발생시킨 애플리케이션 별로 분류하는 과정
- 분류 결과 적용 → Network Design, Network Management, Network Security

❖ Traditional Approach

Approach	Description	Weak Point
Port number-based analysis	<ul style="list-style-type: none">• Well-Known Port Number를 이용하여 Traffic을 분류하는 Traffic Classification Technique• IANA에서 할당한 Port 번호를 사용하는 Traditional Application 분류에 유용함	<ul style="list-style-type: none">• 현재 Application은 Dynamic한 Port 번호를 할당하여 Packet을 발생 → 분류의 정확도 떨어짐
Payload-based analysis	<ul style="list-style-type: none">• Application이 가지는 특정한 Signature를 이용하여 Traffic을 분류하는 Traffic Classification Technique	<ul style="list-style-type: none">• Payload를 암호화한 Packet 분류 불가능• Payload까지 저장해야 함 → 저장공간 증가• Payload 분석 필요 → 분석 수행 시간 증가

서론 (2/2)

❖ ML (Machine Learning) 알고리즘

- 동적으로 변하는 Port 번호의 패턴을 찾아내어 트래픽 분류에 적용
- Payload 내용에 의존하지 않고, 헤더 정보 및 Flow 통계 정보 이용

❖ ML 알고리즘을 적용한 기존 트래픽 분류 연구의 문제점

- 애플리케이션 분류 기준: Port 번호
- 한정된 데이터 안에서의 분류 결과 제시 (Cross validation)
- Flow 기반의 분류 결과만 제시 → Practical 하지 않음 [1]
 - Traffic Shaping, Usage Billing Policy, Network Planning

연구 목표

- ❖ ML 알고리즘을 적용한 Internet Link의 Real-time 애플리케이션 트래픽 분류를 위한 방안 제시
 - 기존 ML 연구 논문의 문제점 제시 및 해결 방안 제안
 - Internet Link의 애플리케이션 트래픽 분류의 Overall Accuracy를 높일 수 있는 최적의 ML 알고리즘과 Feature Set 제시

- ❖ POSTECH Internet Link의 Real-time 트래픽 데이터를 대상으로 ML 알고리즘을 적용한 분류 결과와 NGMON을 적용한 분류 결과 비교 및 분석
 - Real-time 트래픽 데이터의 분류 결과를 분석하기 위해서 NGMON의 분류 결과를 비교 대상으로 선정

관련 연구 (1/4)

❖ ML Approach

- Flow 단위의 **Statistics** 정보를 기반으로 애플리케이션 별 또는 QoS 별 트래픽 분류
 - 통계 정보: Inter-Arrival Time Distribution, Flow Length, Packet Length Distribution 등
- 단점: **Unknown class** (분류하고자 하는 애플리케이션이 아닌 다른 애플리케이션)를 정의하기가 쉽지 않음

❖ Supervised ML 알고리즘

- 알려진 데이터를 이용하여 **Training**한 **Model**을 기반으로 알려지지 않은 **Test Data**를 분류하는 알고리즘
- Neural Network: Multi-layer Perceptron, Radial Basis Function (RBF) Network
- Decision Tree: J48, REPTree
- Bayesian: NaïveBayes, BayesNet

관련 연구 (2/4)

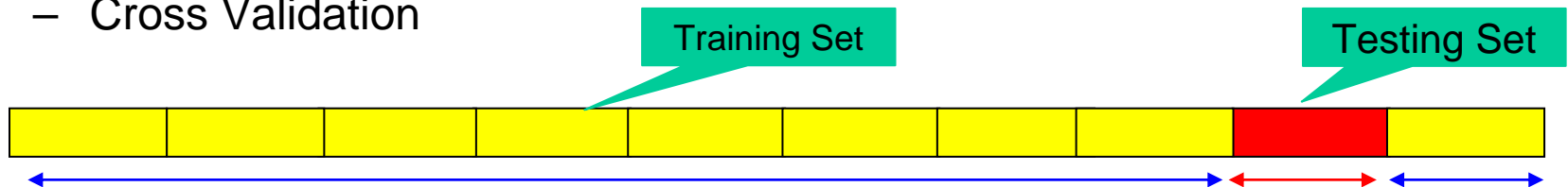
❖ Estimator

- 전체 데이터를 대상으로 분류 결과 평가

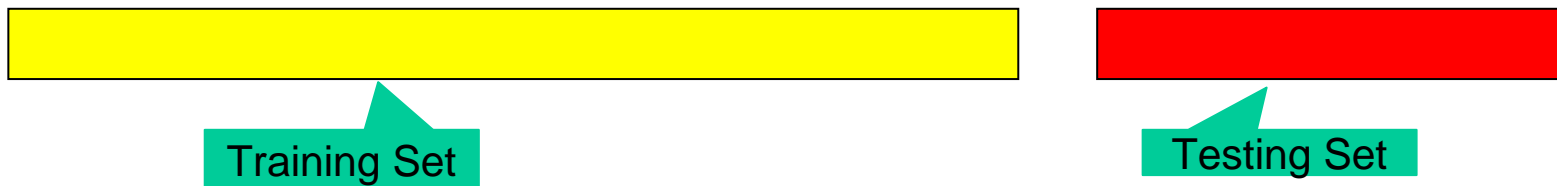
$$\text{Overall Accuracy} = \frac{\sum \text{TP of each Application}}{\text{Total element}}$$

❖ Testing Methodology

- Cross Validation



- Split Validation



관련 연구 (3/4)

❖ 기존 연구 논문에서 사용한 ML 알고리즘

- 기존의 연구 논문에서 제시하는 최적의 알고리즘이 다름 → 각 논문에서 사용하는 데이터와 **feature set**이 달라짐에 따라 최적의 알고리즘은 다름
- 기존 연구는 **well-known port** 번호를 사용하는 애플리케이션을 대상으로 분류함 → 기존 연구의 분류 결과는 다양한 **Application**에서 발생한 데이터들이 많이 있는 네트워크 환경에 적용하기 힘들

ML 알고리즘	Erman et al. [2]	Zander et al. [3]	Nguyen et al. [4]	Park et al. [5]	Andrew et al. [6]	Erman et al. [7]	Williams et al [8]	Park et al. [9]
K-Means	○							
DBSCAN	○							
Autoclass	○							
Expectation Maximization		○				○		
Naïve Bayes			○	○	○	○	○	
NBKE				○	○			○
REP Tree				○				○
J48								○
C4.5							○	
Bayesian Network							○	
Naïve Bayes Tree							○	
FCBF					○			○
GA								○

관련 연구 (4/4)

❖ 기존 ML 연구의 애플리케이션 트래픽 분류 기준

Author	Classification	Description
Andrew et al. [6]	BULK, DATABASE, INTERACTIVE, MAIL, SERVICE, WWW, P2P, ATTACK, GAMES, MULTIMEDIA	단일한 Port 번호를 사용하는 애플리케이션 Category 별로 분류
Park et al. [5]	INTERACTIVE, BULK, REAL TIME, EMAIL, TRANSACTION	
Erman et al. [2]	DNS, FTP, HTTP, IRC, LIMEWIRE, NNTP, POP3, SOCKS.	단일한 Port 번호를 사용하는 특정 애플리케이션 별로 분류
Williams et al. [8]	FTP, TELNET, SMTP, DNS, HTTP, Half-Life	

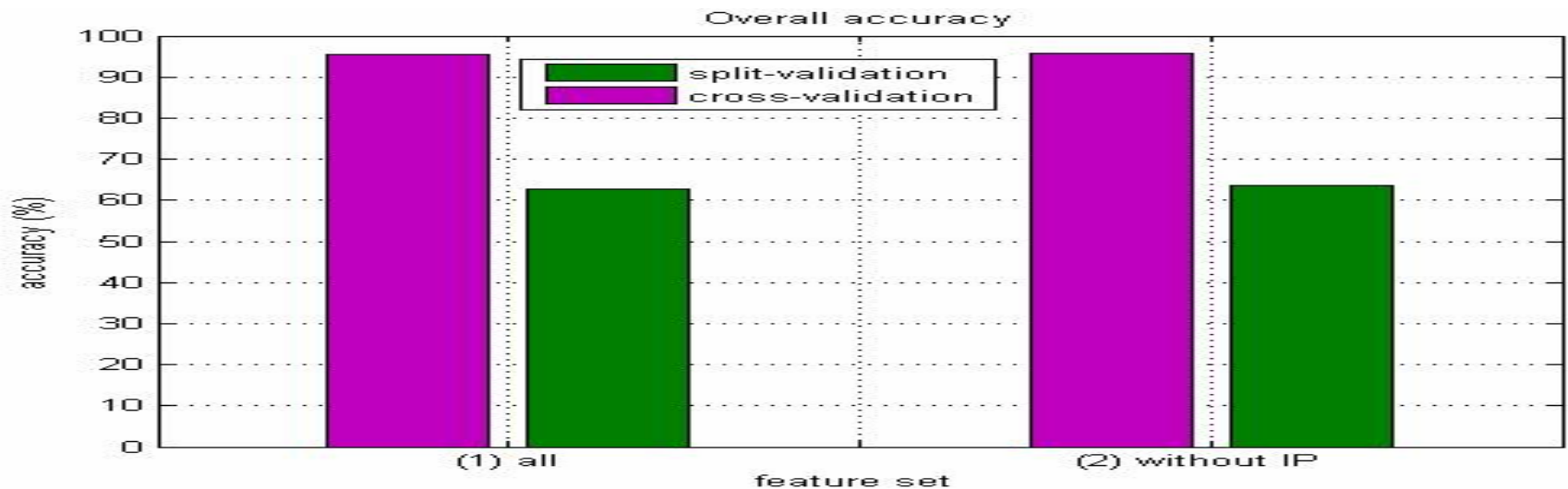
기존 연구의 문제점 & 해결 방안 (1/3)

- ❖ 기존 연구는 애플리케이션 분류 기준으로 Port 번호를 적용함
 - 기존 연구는 Well-known Port 번호를 사용하는 애플리케이션을 대상으로만 트래픽을 분류하여 Training → Training 결과를 적용하여 Testing한 후 애플리케이션 분류 결과를 제시함
 - 기존 연구 분류 결과는 Port Number-based Analysis 분류 결과와 다르지 않음 → 기존 연구의 결과를 동적인 Port 번호를 사용하는 Internet Link 상의 Real-time 애플리케이션 분류에 적용하는데 적합하지 않음
- ❖ Dynamic한 Port 번호를 사용 또는 정체를 숨기기 위해 Well-known Port 번호를 사용하는 애플리케이션 분류
 - NGMON의 애플리케이션 분류 결과를 통해서 분류 대상의 애플리케이션 선정
 - 각 애플리케이션이 생성하는 트래픽을 실제로 capture하여 데이터 수집
 - P2P나 Web Disk 애플리케이션이 발생한 데이터를 수집 → Training Set과 Testing Set을 만듦
 - ML 알고리즘을 적용한 애플리케이션 트래픽 분류

기존 연구의 문제점 & 해결 방안 (2/3)

❖ 한정된 데이터 안에서의 분류 기준 제시 (Cross validation)

- Cross Validation: Training Set과 Testing Set이 동시간에 얻어진 트래픽 데이터로 구성된 것이기 때문에 Training Set과 Testing Set이 가지는 Pattern이 같음 → 분류의 Overall accuracy가 높음
- Split Validation: Training Set과 Testing Set이 다른 시간에 얻어진 트래픽 데이터로 구성된 것이기 때문에 Training Set과 Testing Set이 가지는 Pattern이 항상 같지 않음 → Internet Link상의 Real-time 트래픽 데이터의 특성상 Split validation을 적용해야 함



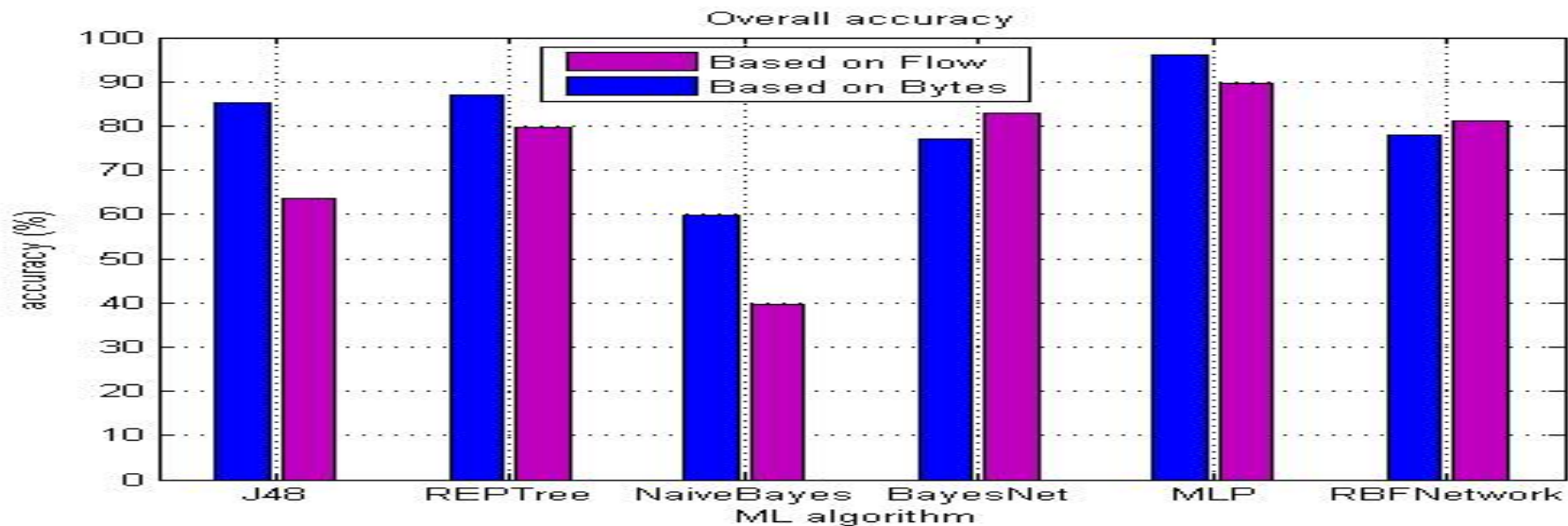
기존 연구의 문제점 & 해결 방안 (3/3)

❖ Flow 기반의 분류 결과만 제시

- Flow-based로 Accuracy를 보는 것은 Flow마다 제각각 다르게 가지는 가중치를 반영하지 못함 → **Class Imbalance Problem** [1] (e.g. 1,000-byte flow vs. 10,000-byte flow)

❖ Byte 기반의 분류 결과가 Practical한 측면에서 유용함 (Erman et al.)

- Traffic Shaping, Usage Billing Policy, Network Planning

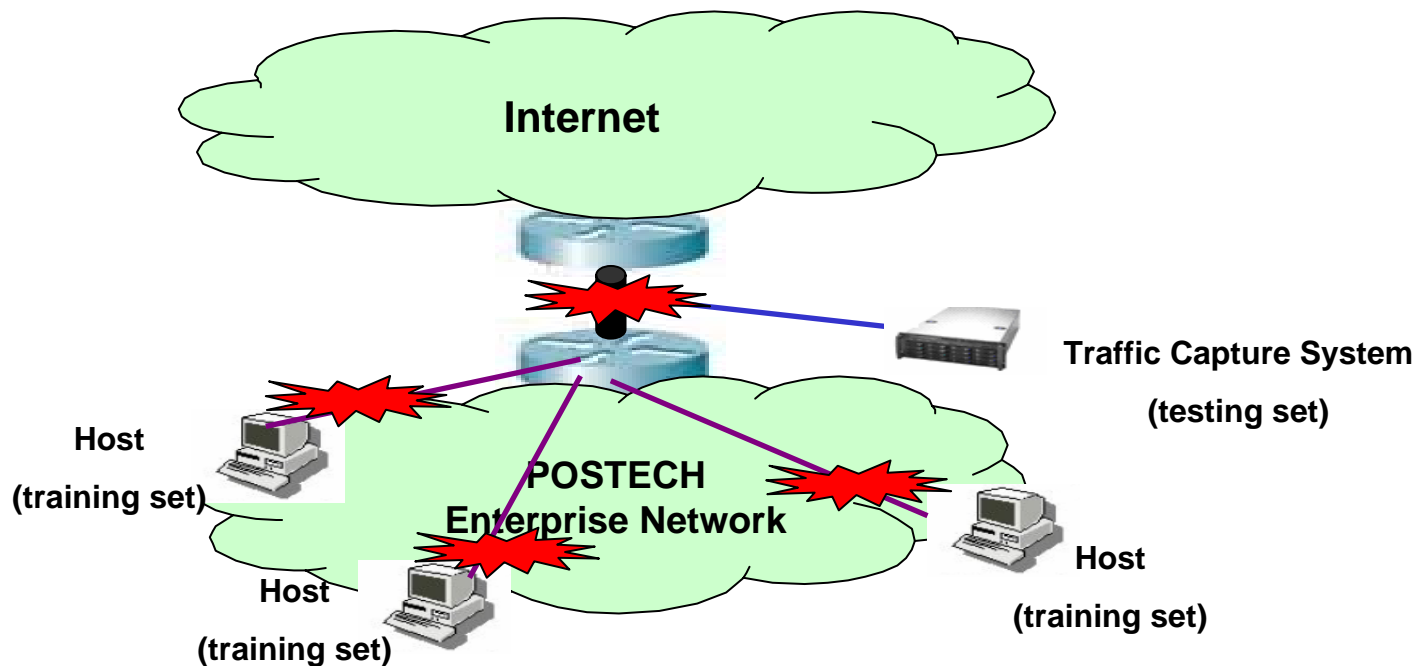


ML 알고리즘 적용 방안

1. Training과 Testing을 위한 트래픽 데이터 수집 방법
2. 분류 애플리케이션 선정
3. Split Validation 기법 적용
4. ML 알고리즘과 Feature Set 선정
5. 최적의 ML 알고리즘과 Feature Set

데이터 수집 방법

- ❖ Host에서 수집한 데이터로 만든 Training Set을 Modeling
→ 인터넷 링크에서 수집한 데이터로 만들어진 Testing Set
을 분류하는 방법 이용



분류 애플리케이션 선정

- ❖ POSTECH의 Traffic을 발생시킨 애플리케이션을 조사
 - NGMON의 애플리케이션 트래픽 분류 결과를 이용

Category	Application
Web Disk	CLUBBOX, PARANDISK, TOTODISK, ENDISK
Web Traffic	HTTP, HANGAME, SAYCLUB
FTP	ALFTP
P2P	FILEGURI, SORIBADA, GAMPLE, BITTORRENT
Game Traffic	STARCRAFT
Remote Controller	MSTC

	BITTORRENT		FTP		CLUBBOX		DESKTOP		FILEGURI		GAMPLE		ENDISK	
	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)
Training	11000	708	518	1011	231	165	86	28	571	506	10943	989.9	246	447
Testing	4000	256	231	342	165	102	34	12.4	408	407	546.9	4032		

	HANGAME		HTTP		PARANDISK		SAYCLUB		SORIBADA		STARCRAFT		TOTODISK	
	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)	Flow(개수)	Byte(MB)
Training	998	23.5	4263	26.8	223	278.45	2870	10.07	8803	339	15	0.244	526	367
Testing	47	7	2045	15.3	148	145	85	3	5960	273	6	0.126	247	192

Split Validation 기법 적용

- ❖ Split Validation으로 애플리케이션 트래픽 분류를 하기 위해서 충분한 데이터를 기반으로 Training Set을 만듦
- ❖ Nguyen et al.은 적합한 Training Set을 만들기 위해서는 각 애플리케이션마다 25개의 flow가 필요하다고 함 → Cross Validation
- ❖ 본 논문에서 실험한 결과 Split Validation을 이용할 때에는 각 애플리케이션마다 200개의 flow가 필요함

ML 알고리즘과 Feature Set 선정

❖ Weka 를 이용한 ML 알고리즘 적용

- 기존 ML 연구 논문이 사용한 알고리즘: J48, REPTree, Bayesian-Network
- 다른 영역에서 많이 사용하는 알고리즘: MLP

❖ Feature Set

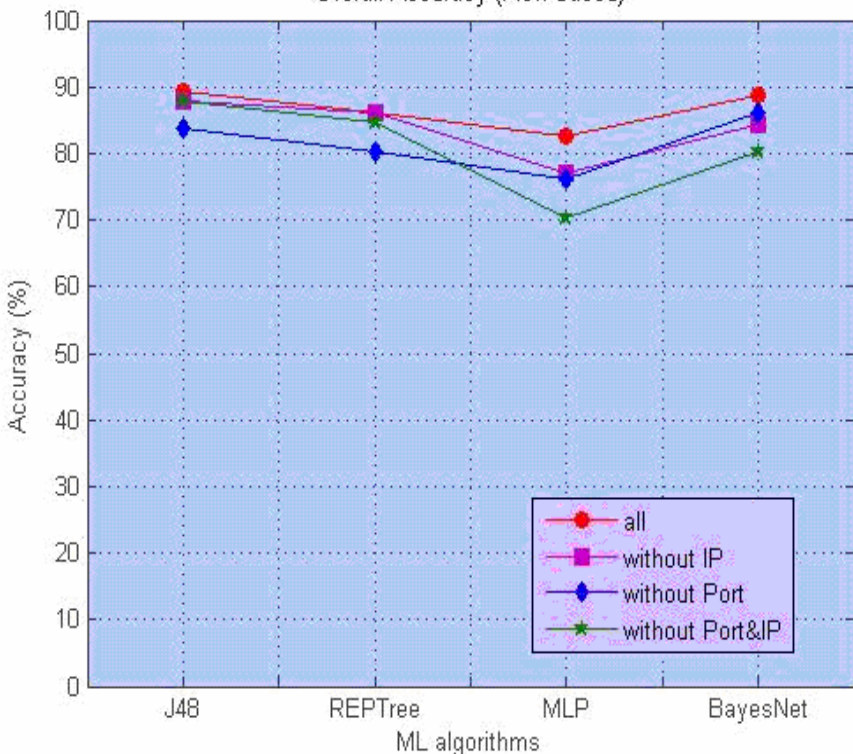
- 1) All: 모든 Feature가 선택된 경우
- 2) Without IP: Source와 Destination의 IP Address를 제외한 경우
- 3) Without port: Source와 Destination의 Port 번호를 제외한 경우
- 4) Without IP&Port: Source와 Destination의 IP Address와 Port 번호를 제외한 경우

IP address (source, destination)
Port number (source, destination)
Byte counts
Connection duration
Packet size statistics (minimum, maximum, mean, standard deviation)
Inter - packet arrival time statistics (minimum, maximum, mean, standard deviation)

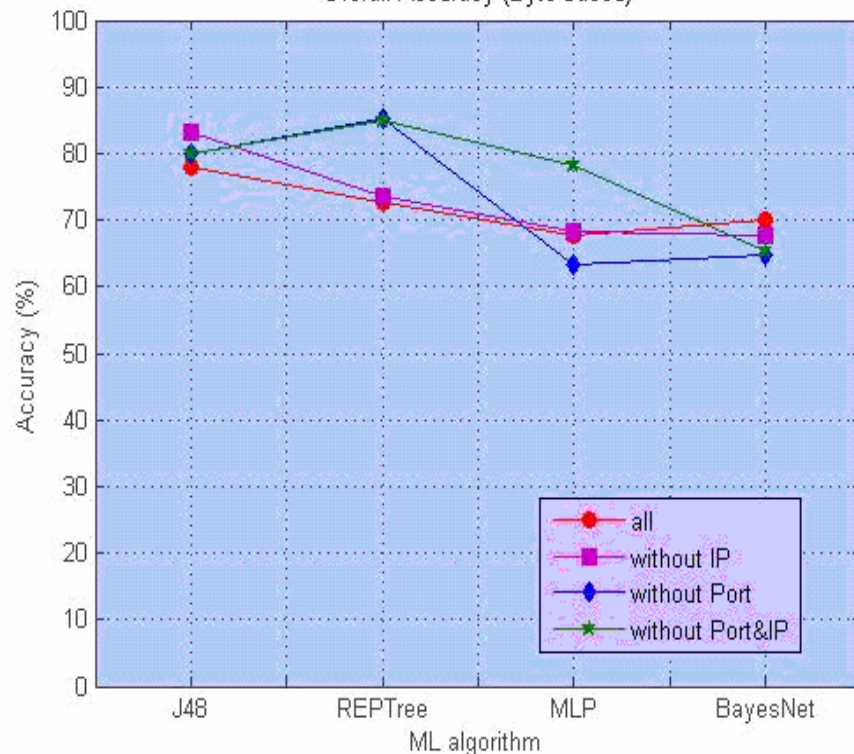
최적의 알고리즘과 Feature Set (1/2)

❖ ML 알고리즘을 적용한 분류 결과 (Flow-based & Byte-based)

Overall Accuracy (Flow-based)



Overall Accuracy (Byte-based)



최적의 알고리즘과 Feature Set (2/2)

❖ Flow-based Accuracy

- J48 or Bayesian-Network & 1) Feature Set, 약 89%

❖ Byte-based Accuracy

- REPTree & (2) Feature Set, 약 86%

❖ J48이 최적의 알고리즘

- J48이 Flow-based와 Byte-based에서 좋은 정확도를 보였으며 또한 Feature Set에 따른 편차도 작음

❖ Byte-based Accuracy가 Flow-based Accuracy보다 낮은 이유

- FTP와 같이 큰 byte를 가지는 flow의 분류 정확도가 낮기 때문

ML (J48) vs. NGMON

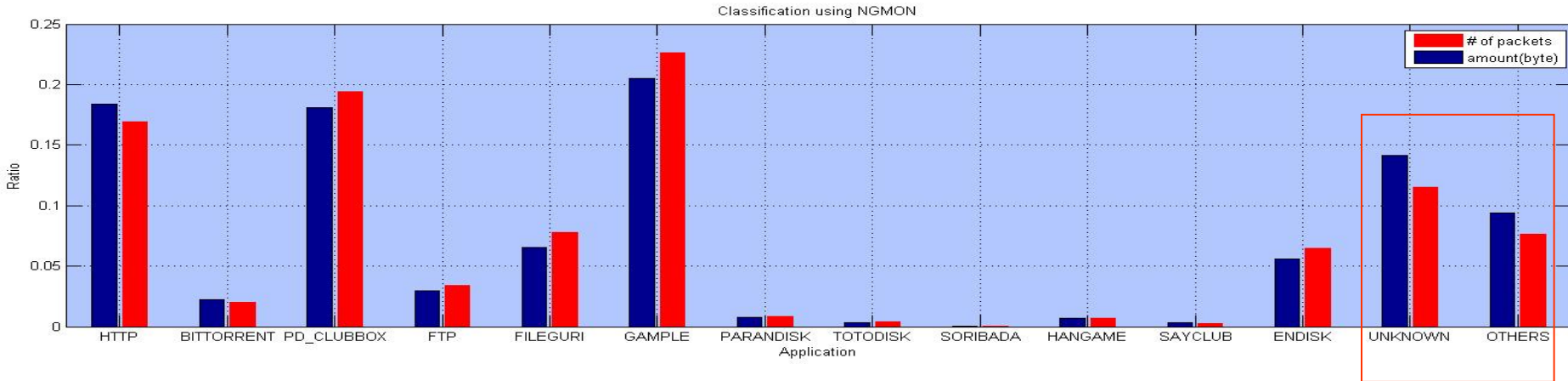
분류 결과 비교

Unknown 데이터 분류

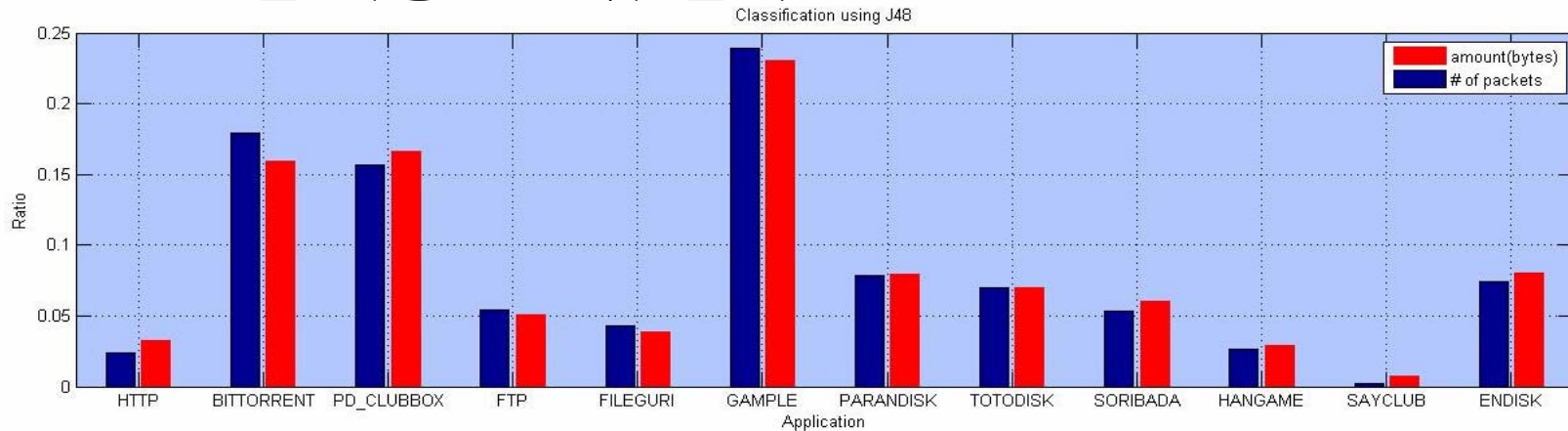
- ❖ Host에서 수집한 데이터로 만든 Training Set을 Modeling 하고 Internet Link에서 수집한 데이터로 만들어진 Testing Set을 분류함
 1. 인터넷 링크에서 약 3시간 동안 데이터를 수집 →알려지지 않은 데이터
 2. 처음에 한 실험에서 사용한 Training set을 Modeling하고 1에서 얻은 데이터를 이용한 Testing Set을 J48을 적용하여 분류
 3. 1에서 얻은 데이터를 NGMON으로 분류한 결과를 2에서 얻은 분류 결과와 비교
 - 알려지지 않은 데이터의 ML 알고리즘을 적용한 분류 결과를 비교하기 위해서 NGMON의 분류 결과를 참조

분류 결과 (1/2)

❖ NGMON을 적용한 분류 결과



❖ J48을 적용한 분류 결과



분류 결과 (2/2)

- ❖ ML과 NGMON의 분류 결과가 다른 이유
 - 각 방법에서 정의하는 애플리케이션의 의미가 다름
 - E.g. NGMON의 FTP vs. ML의 FTP
 - NGMON이 최근의 애플리케이션의 변화를 반영하지 못함
 - 각 방법의 Port 번호를 80번 쓰는 애플리케이션들의 분류 차이
 - 각 방법의 BITTORRENT를 분류 하는 차이

특정 애플리케이션 분류 (1/3)

❖ Port 번호로 80번을 사용하는 애플리케이션 분류

Port 번호 80을 사용하는 Application	NGMON	J48 (ML)
HTTP	16.9%	3.2%
GAMPLE	22%	23%
TOTODISK	0.4%	6.7%
PARANDISK	0.9%	7.9%
SORIBADA	0%	6%
HANGAME	1%	2.9%
SAYCLUB	0.3%	0.7%

단일한 IP 사용 또는 한 서브넷에 속하는 다수개의 IP 사용
 다수개의 서브넷에 속하는 다수개의 IP 사용

$$16.9+22+0.4+0.9+0+1+0.3 = 41.5$$

$$16.9+0.4+0.9+0 = 18.2$$

$$3.2+23+6.7+7.9+6+2.9+0.7 = 50.4$$

$$3.2+6.7+7.9+6 = 23.8$$

알고리즘	설명
NGMON	Port 번호로 80번을 사용하고 단일한 IP, 또는 한 서브넷의 다수개의 IP를 사용하면 Application은 분류가 가능하지만 다수개의 IP를 사용하는 Application은 HTTP로 분류됨 (e.g. SORIBADA, TOTODISK)
J48 (ML)	Port 번호로 80번을 사용하고 다수개의 IP를 사용하는 Application도 분류 가능

특정 애플리케이션 분류 (2/3)

❖ BITTORRENT의 분류 비교

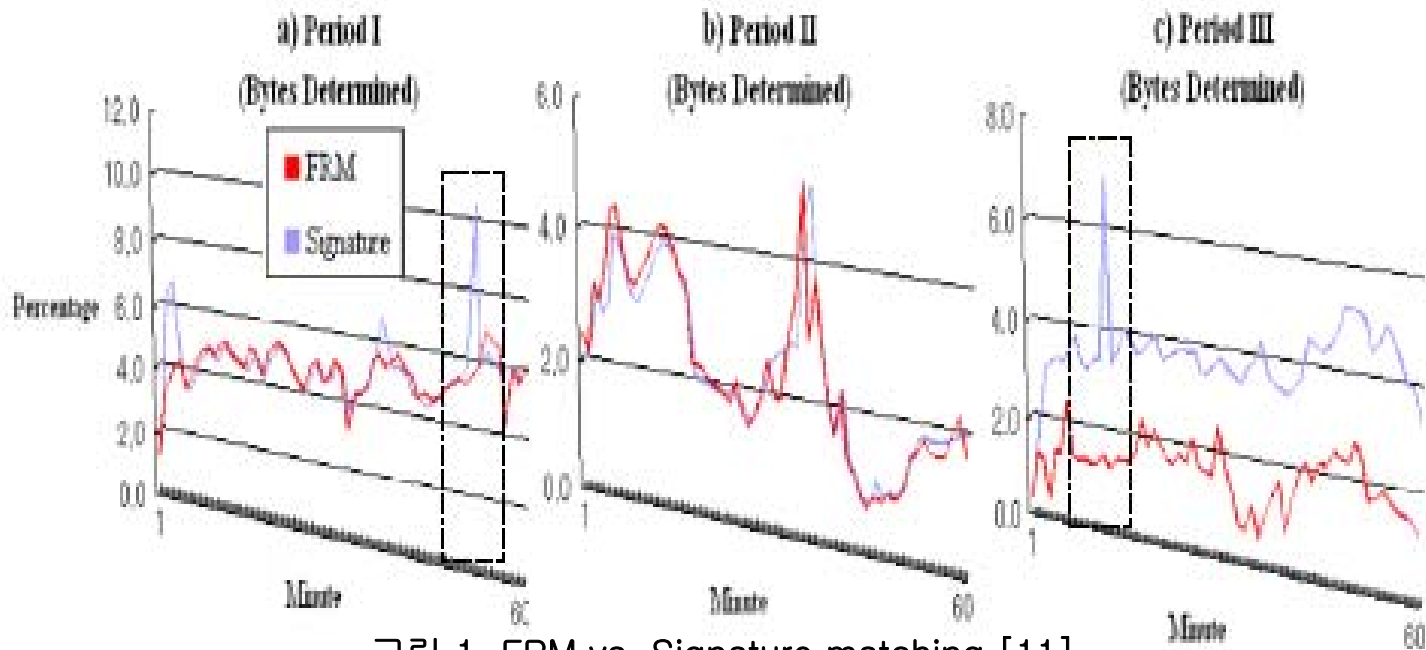


그림 1. FRM vs. Signature matching [11]

	BITTORRENT
J48 (ML)	15.9%
NGMON	2%

표 1. NGMON vs. J48 (ML)

특정 애플리케이션 분류 (3/3)

- ❖ 그림 1의 a)와 c)에서 나타나 듯 NGMON이 측정한 점유율보다 약 2~3배 가량 더 많은 점유율을 Signature를 이용하여 측정할 수 있음
 - J48을 이용하여 측정한 결과와 비슷함
- ❖ 그림 1에서 보여준 점유율의 차이보다 표 1에서 더 많은 점유율의 차이를 보이는 까닭
 - NGMON의 UNKNOWN 또는 OTHERS에 속하는 Packet들 중 BITTORRENT 또는 BITTORRENT와 비슷한 애플리케이션에서 발생된 데이터들이 많기 때문

결론 (1/2)

❖ ML 알고리즘을 적용한 인터넷 트래픽 분류의 기존 연구의 문제점 해결

- 애플리케이션 기준이 아닌 **Port** 번호 기준의 분류 결과 제시 → 애플리케이션별 데이터 수집 방법을 사용하여 다양한 애플리케이션을 대상으로 트래픽 분류 결과 제시
- 한정된 데이터 안에서의 분류 결과 제시 (**Cross validation**) → 충분한 데이터로 만든 **Training Set**을 기반으로 **Split validation**을 이용한 분류 결과 제시
- **Flow** 기반의 분류 결과만 제시 → **Byte** 기반의 분류 결과 제시

❖ 최적의 알고리즘과 **Feature Set** 제시

- Flow-based Accuracy: J48 or Bayesian-Network & 1) Feature Set
- Byte-based Accuracy: REPTree & 2) Feature Set
- J48이 Flow-based와 Byte-based에서 좋은 정확도를 보였으며 또한 Feature Set에 따른 편차도 작음 → **J48이 최적의 알고리즘**

결론 (2/2)

- ❖ ML 알고리즘 적용 분류 결과와 NGMON 적용 분류 결과 비교 및 분석
 - HTTP와 BITTORRENT을 분류한 결과를 비교해 보았을 때, ML 알고리즘을 적용한 분류 결과가 더 정확한 결과를 가짐
 - ML을 적용한 트래픽 분류 방법을 기존 트래픽 분류 방법(Port-based analysis, NGMON)이 가지는 한계의 대안으로 제시 가능함

- ❖ 어떤 Internet Link의 데이터를 분류 하려면 그 Internet Link의 데이터가 어떤 애플리케이션에서 발생된 데이터인지 사전 조사한 후 그것을 기반으로 Training Set을 만들어야 함
 - 존재하는 모든 Application들의 데이터를 기반으로 Training Set을 만들면 사전 조사를 하지 않아도 됨

향후 연구

- ❖ 정확한 애플리케이션 데이터를 사용하여 ML 알고리즘과 NGMON의 분류 결과를 비교
 - POSTECH, 고려대학교의 Traffic Data를 활용
- ❖ ML 알고리즘을 적용하여 모든 애플리케이션 트래픽 분류가 가능하도록 Training Set을 만듦
 - 한 애플리케이션에 200개의 Flow 정보를 수집
 - 가능한 많은 애플리케이션의 데이터를 기반으로 Training Set을 만듦
- ❖ 최소한의 Payload를 분석하는 Signature Approach와 ML 알고리즘을 같이 적용하는 Hybrid Approach를 이용하여 애플리케이션 트래픽 분류 제안

Q & A

감사합니다!



References

- [1] Jeffrey Erman, Anirban Mahanti, Martin Arlitt, “*Byte Me: A Case for Byte Accuracy in Traffic Classification*”, MineNet’07, J San Diego, California, USA, Jun. 2007, pp. 35~37.
- [2] Jeffrey Erman, Martin Arlitt, Anirban Mahanti, “*Traffic Classification Using Clustering Algorithms*”, SIGCOMM’06 Workshops, Pisa, Italy, Sep. 2006, pp. 281~286.
- [3] Sebastian Zander, Thuy Nguyen, Grenville Armitage, “*Automated Traffic Classification and Application Identification using Machine Learning*”, Proceedings of the IEEE Conference on Local Computer Networks, Sydney, Australia, Nov. 2005, pp. 250-257.
- [4] Thuy T. T. Nguyen, Grenville Armitage, “*Training on multiple sub - flows to optimize the use of Machine Learning classifiers in real - world IP networks*”, IEEE Conference on Local Computer Networks, Tampa, Florida, USA, Nov. 2006, pp. 369~376.
- [5] Junghun Park, Hsiao - Rong Tyan, and C. C. Jay Kuo, “*Inetnet Traffic Classification For Scalable QoS Provision*”, IEEE International Conference on Multimedia and Expo, Jul. 2006, pp. 1221~1224.

References

- [6] Andrew W. Moore, Denis Zuev, “*Internet Traffic Classification Using Bayesian Analysis Techniques*”, SIGMETRICS’05, Banff, Alberta, Canada, Jun. 2005, pp. 50~60.
- [7] Jeffrey Erman, Anirban Mahanti, Martin Arlitt, “*Internet Traffic Identification using Machine Learning*”, IEEE Global Telecommunications Conference, California, USA. Nov.~Dec. 2006, pp. 1~6.
- [8] N. Williams, S. Zander, G. Armitage, “*A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification*”, SIGCOMM Computer Communication Review, Oct. 2006, pp. 7~15.
- [9] Junghun Park, Hsiao - Rong Tyan, C. - C. Jay Kuo, “*GA - Based Internet Traffic Classification Technique for QoS Provisioning*”, International Conference on Intelligent Information Hiding and Multimedia, Pasadena, California, USA, Dec. 2006, pp. 251~254.
- [10] Andrew Moore, Denis Zuev and Michael Crogan, “*Discriminators for use in flow - based classification*”, Technical Report, Intel Research Cambridge, 2005.
- [11] Young J. Won, “*A Hybrid Approach for Accurate Application Traffic Identification*”, MS Thesis, Dept. of Computer Science and Engineering,

Appendix

최적의 알고리즘과 Feature Set (2/2)

Application	Precision	Recall
Bittorrent	0.978	0.757
Alftp	0.177	0.061
Clubbox	0.615	0.508
Desktop	1	1
Fileguri	0.713	0.559
Gample	0.962	0.983
Hangame	0.169	0.511
Http	0.917	0.956
Parandisk	0.185	0.432
Sayclub	0.126	0.376
Soribada	0.891	0.983
Starcraft	1	0.231
Totodisk	0.784	0.514

Bayesian-Network & all

Application	Precision	Recall
Bittorrent	0.981	0.637
Alftp	0.0045	0.0003
Clubbox	0.768	0.232
Desktop	1	1
Fileguri	0.35	0.202
Gample	0.953	0.981
Hangame	0.43	0.102
Http	0.403	0.858
Parandisk	0.133	0.311
Sayclub	0.003	0.02
Soribada	0.968	1
Starcraft	1	0.145
Totodisk	0.97	0.09

REPTree & without Port

- Starcraft: 해당 애플리케이션의 Training이 제대로 이루어 지지 않음 (적은 게임 traffic 발생 → training data 부족)
- Hangame & Sayclub: 해당 애플리케이션에 속한 많은 트래픽이 HTTP로 분류됨
- Alftp & Parandisk: Alftp에 속하는 트래픽이 Parandisk에 또는 Parandisk에 속하는 트래픽이 Alftp로 서로 엇갈려서 분류됨

관련 연구 (1/4)

❖ Machine Learning Approach

- Flow 단위의 Statistics 정보를 기반으로 애플리케이션 별 또는 QoS 별 트래픽 분류
 - 통계 정보: Inter-Arrival Time Distribution, Flow Length, Packet Length Distribution
- 단점: Unknown을 정의할 수 없음

❖ Supervised ML Algorithm

- 알려진 데이터를 이용하여 training한 Model을 기반으로 알려지지 않은 test data를 분류하는 알고리즘
- Neural Network: Multi-layer Perceptron, Radial Basis Function (RBF) Network
- Decision Tree: J48, REPTree
- Bayesian: NaïveBayes, BayesNet

❖ Unsupervised ML Algorithm

- 데이터들의 유사성을 기반으로 그룹핑함으로써 분류하는 알고리즘
- 알려진 그룹 외에 전혀 다른 그룹으로도 분류가 될 수 있음 → 새로운 그룹이 정확히 무엇인지 알 수 없음
- Expectation Maximization, K-Nearest Neighbor, DBSCAN, etc

각 approach의 분류 비교 및 분석 (5/5)

❖ Discussion

	장점	단점	앞으로 나아가야 할 방향
ML (J48)	<ul style="list-style-type: none"> 다른 Approach에 비해 Intelligent한 기법을 적용하여 최근 애플리케이션 변화에 대응 패킷의 Header의 값만을 이용하여서 높은 정확도를 제공 	<ul style="list-style-type: none"> 단지 12개의 애플리케이션에 대해서만 적용 	<ul style="list-style-type: none"> 좀 더 많은 데이터를 이용하여 현재보다 나은 Training Set을 얻어야 함 좀 더 많은 애플리케이션에 적용할 수 있도록 개선해야 함 많은 애플리케이션에 대한 데이터 수집
NGMON (FRM)	<ul style="list-style-type: none"> 많은 애플리케이션에 적합하게 대응 현재 real time에 적합한 시스템을 구축 	<ul style="list-style-type: none"> 구현하고 난 이후 최근 애플리케이션 변화를 적용하지 않음 	<ul style="list-style-type: none"> 최근 애플리케이션 변화를 적용하여야 함

관련 연구 (4/4)

❖ 기존 연구 논문이 사용한 feature

	Erman et al. [2]	Zander et al. [3]	Nguyen et al. [4]	Park et al. [5]	Andrew et al. [6]	Erman et al. [7]	Williams et al. [8]
Total number of packet	<input type="radio"/>			<input type="radio"/>		<input type="radio"/>	
Packet size	<input type="radio"/>		<input type="radio"/>	<input type="radio"/>		<input type="radio"/>	<input type="radio"/>
Payload size	<input type="radio"/>				<input type="radio"/>		
Number of transferred bytes	<input type="radio"/>						
Inter packet arrival time	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
IP address		<input type="radio"/>	<input type="radio"/>				
Port number		<input type="radio"/>	<input type="radio"/>		<input type="radio"/>		
Flow size		<input type="radio"/>					<input type="radio"/>
Flow duration		<input type="radio"/>		<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Protocol			<input type="radio"/>				<input type="radio"/>
Inter-packet length variation			<input type="radio"/>				
Initial advertised-window bytes				<input type="radio"/>			
Number of packet with 'PUSH' option				<input type="radio"/>			
Advertised-windows bytes				<input type="radio"/>			
Effective bandwidth based upon entropy					<input type="radio"/>		
Fourier Transform of packet					<input type="radio"/>		