Automated Application Signature Generation Using LASER and Cosine Similarity

Byungchul Park, Jae Yoon Jung, John Strassner[*], and James Won-ki Hong[*]

{fates, dejavu94, johns, jwkhong}@postech.ac.kr

Dept. of Computer Science and Engineering, POSTECH, Korea

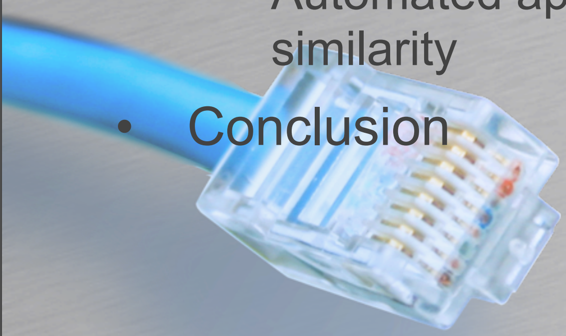
[*]Division of IT Convergence Engineering, POSTECH, Korea

April 24, 2010

The 3rd CAIDA-WIDE-CASFI Joint Measurement Workshop

Contents

- Introduction
- Traffic classification based on flow similarity
  - Research goal
  - Overview of proposed methodology
  - Vector space modeling
  - Measuring packet/flow similarity
  - Evaluation Result
- What is next step?
  - Fine-grained traffic classification
  - Automated application signature generation using LASER and flow similarity
- Conclusion



Introduction

- Internet traffic classification gains continuous attentions
- CAIDA have created a structured taxonomy of traffic classification papers and their data set (68 papers, 2009)
- Various methodologies for traffic classification

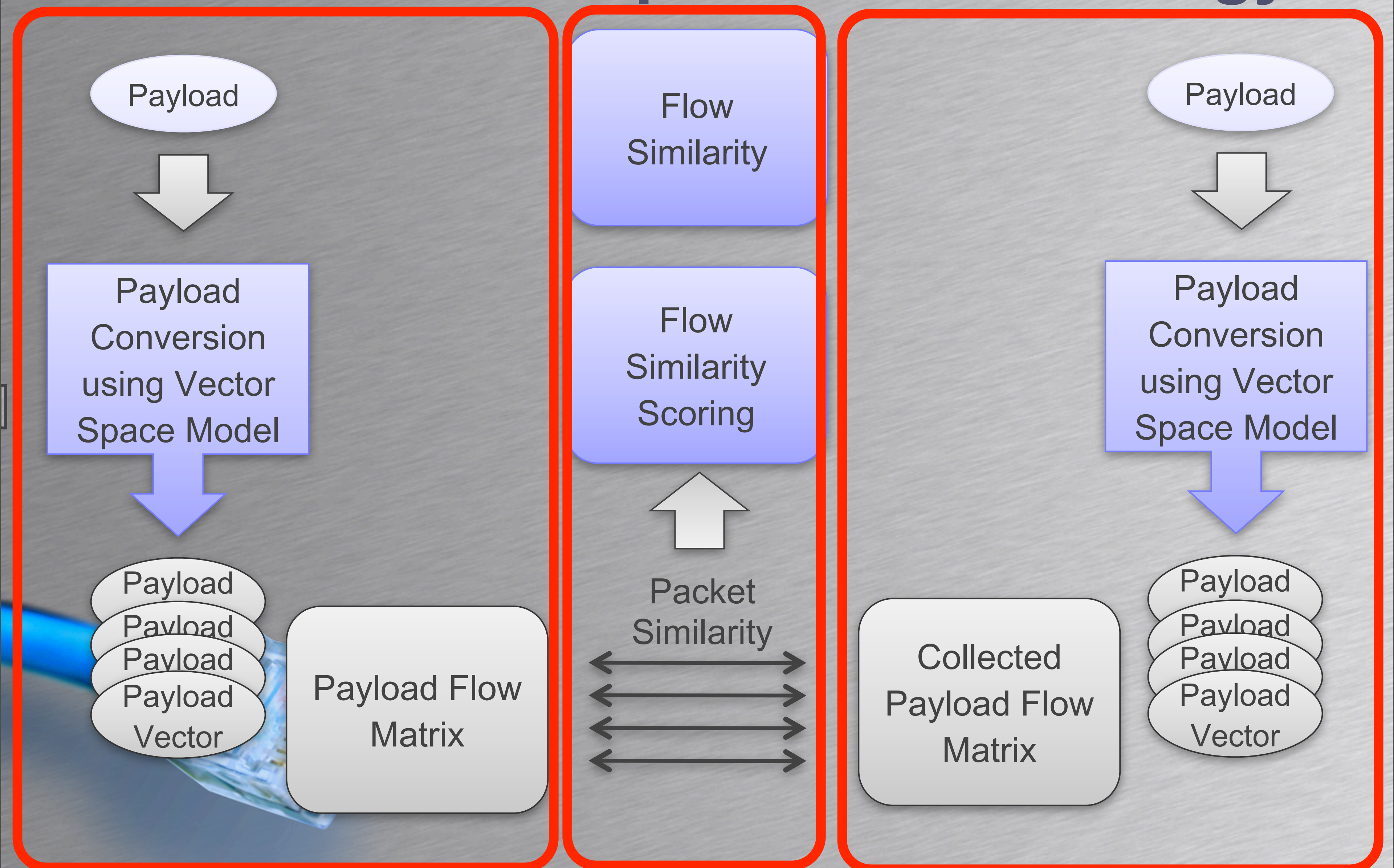
	Accuracy	Strength	Weakness
Port-based	Low	Low computational cost	Low accuracy
Signature-based	High	Most accurate method	Exhaustive signature generation
ML-based	High	Can handle encrypted traffic	High complexity Affected by network condition

- How can we guaranty the classification accuracy with low complexity?
  - Develop a methodology to generate application signature automatically
  - Develop another methodology using packet payload contents

Traffic classification based on flow similarity

- Research goal: a new traffic classification methodology
  - Analyzing payload contents
  - High accuracy and low complexity
- Document classification → Traffic classification
  - Document classification in natural language processing
  - Document ≒ Packet (or traffic)
- Apply a variation of document classification approach to traffic classification
  - Low processing overhead
  - Comparable accuracy to signature-based classification
  - No more exhaustive signature extraction tasks
  - Simple numerical representation of similarity between network traffic

Overview of Proposed Methodology

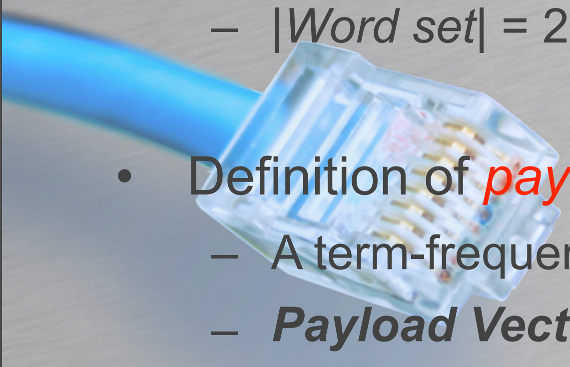


Vector Space Modeling (1/2)

- An algebraic model representing text document as vectors
- Widely used in document classification research

- Payload vector conversion
  - Document classification in natural language processing
  - Document ≒ Packet (or traffic)
  - Document classification utilize occurrence

- Definition of *word* in payload
  - Payload data within an i-bytes sliding window
  - |*Word set*| = $2^{(8*\text{sliding window size})}$



- Definition of *payload vector*
  - A term-frequency vector in NLP
  - ***Payload Vector* = $[w_1\ w_2\ \dots\ w_n]^T$**

Vector Space Modeling (2/2)

Word Word Word

```
HEX
13 42 69 74 54 6f 72 72 65 6e 74 20 70 72 6f 74
6f 63 6f 6c 00 00 00 00 00 10 00 05 fb 95 c0 23
94 92 5e 38 fd 60 57 a1 43 8a e6 96 2b c9 7a c7
4d 36 2d 31 2d 32 2d 2d 6e 34 5f f2 60 1f 2c f7
b1 01 17 e1

ASCII
.BitTorrent prot
ocol..........#
..^8.`W.C...+.z.
M6-1-2--n4_.`.,.
....
```

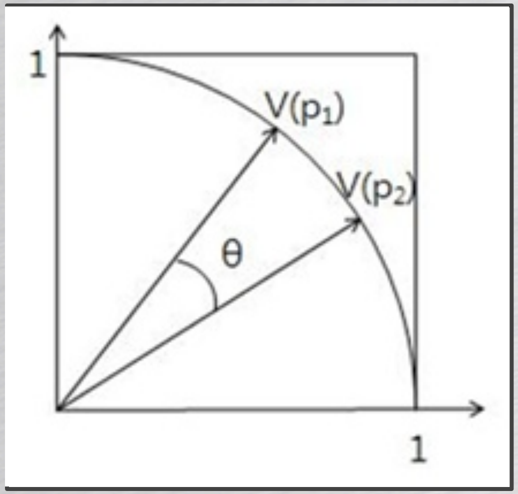
```
Payload Vector
p[0x0000] = 4;
...
p[0x1342] = 1;
...
p[0x4269] = 1;
...
p[0x6974] = 1;
...
p[0x0117] = 1;
...
p[0x17e1] = 1;
...
```

- The word size is 2 and the word set size is 2^{16}
- Larger word size → dimension of payload vector is increased exponentially

Measuring Packet Similarity

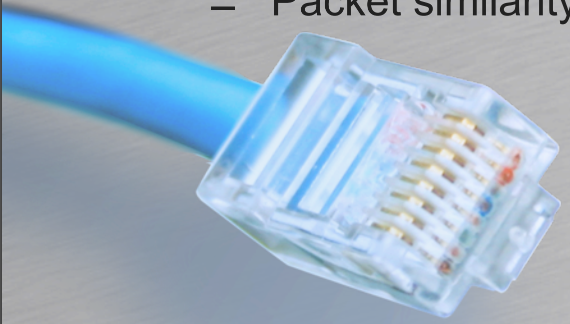
- Cosine Similarity
  - The most common similarity metric in NLP

$$Similarity\ (p_1, p_2) = \frac{V(p_1) \cdot V(p_2)}{|V(p_1)|\,|V(p_2)|}$$



{ 0: Independent
  1: Exactly same

- Packet Comparison
  - Packet similarity = Cosine Similarity (payload_vector$_1$, payload_vector$_2$)



{ 0: Payloads are different
  1: Payloads are similar

Measuring Flow Similarity

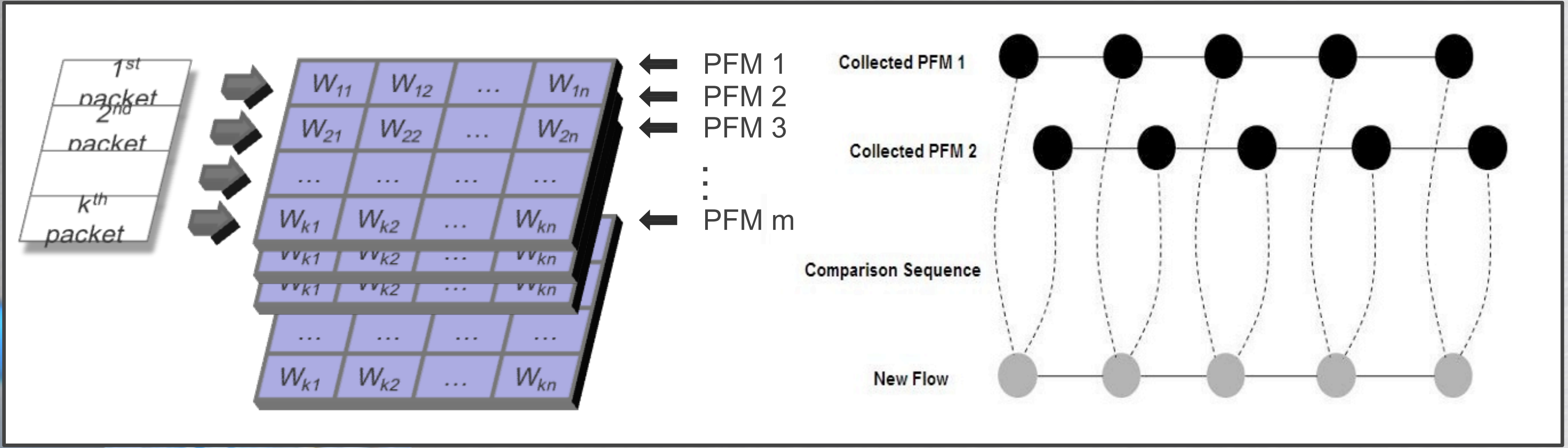
- Payload Flow Matrix (PFM)
  - *k* payload vectors in a flow
  - Represent a traffic flow

$$PFM = [p_1\ p_2\ \dots\ p_k]^T$$

where p_i is payload

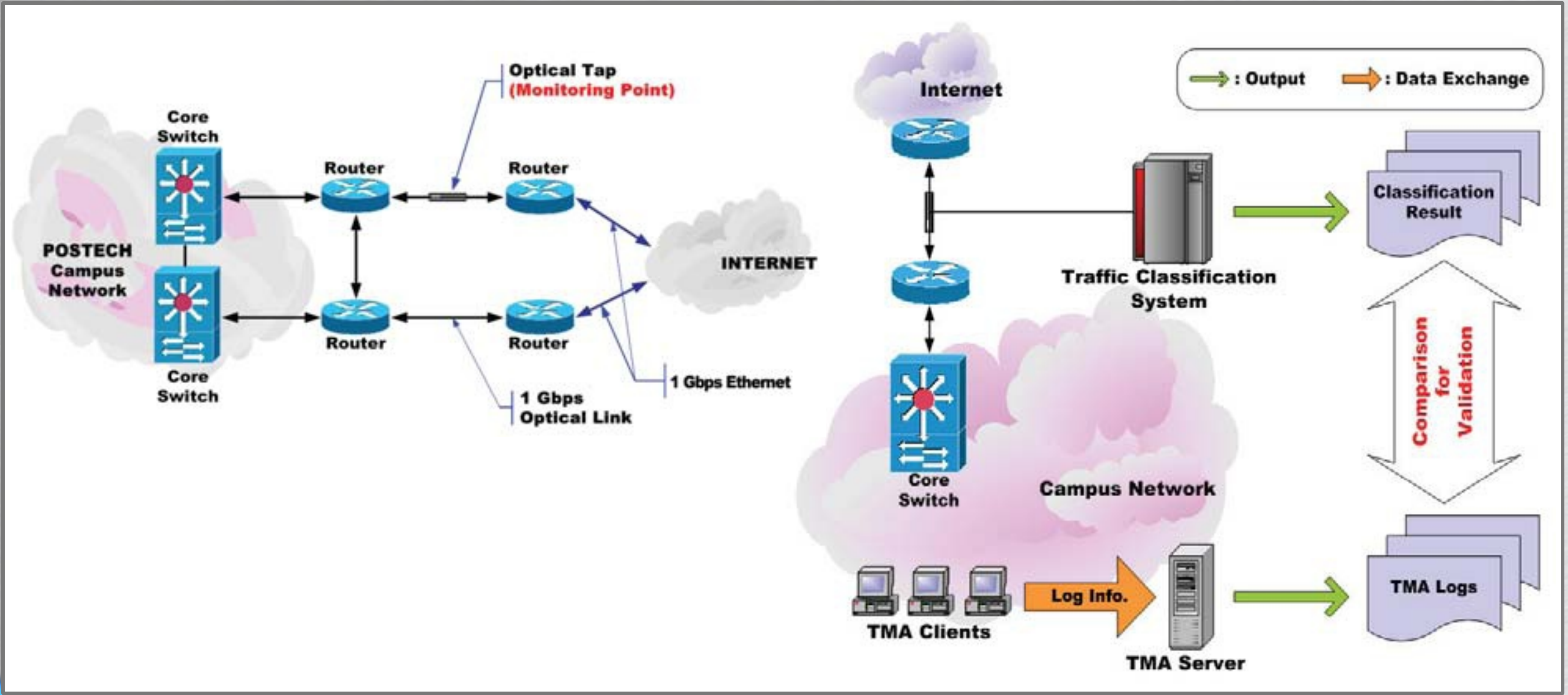
- Collected PFM
  - Information about target flows
  - Alternative signatures
  - Accumulated empirically to enhance signature word

Collected PFMs =
a * new PFM + (1 - a) * Collected PFMs



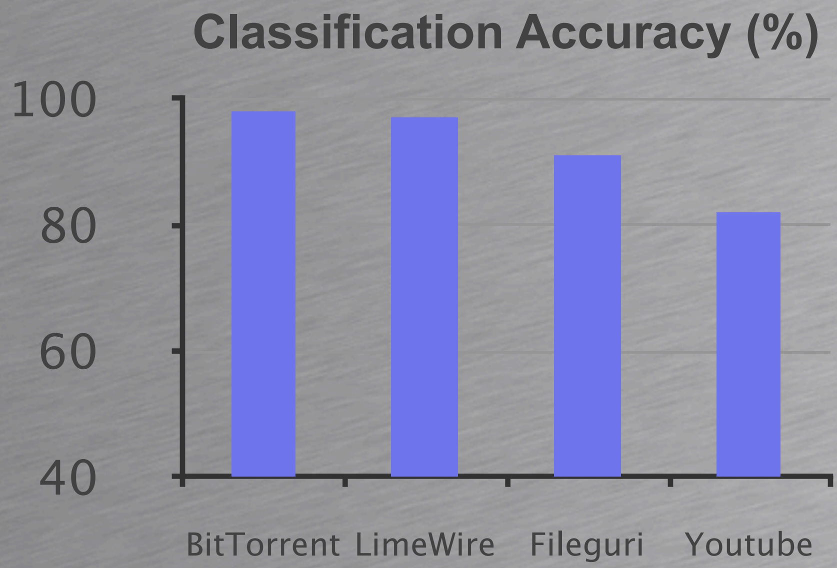
- Packets are compared sequentially with only the corresponding packet in the other flow
- Flow similarity score = ∑ packet similarity

Measuring Packet Similarity



- Dataset: traffic trace on one of two Internet junction at POSTECH
- Traffic Measurement Agent (TMA)
  - Monitoring the network interface of the host
  - Recording log data (5-tuple flow info., process name, packet count, etc)
  - Generating ground-truth to validate traffic classification results

Classification Results



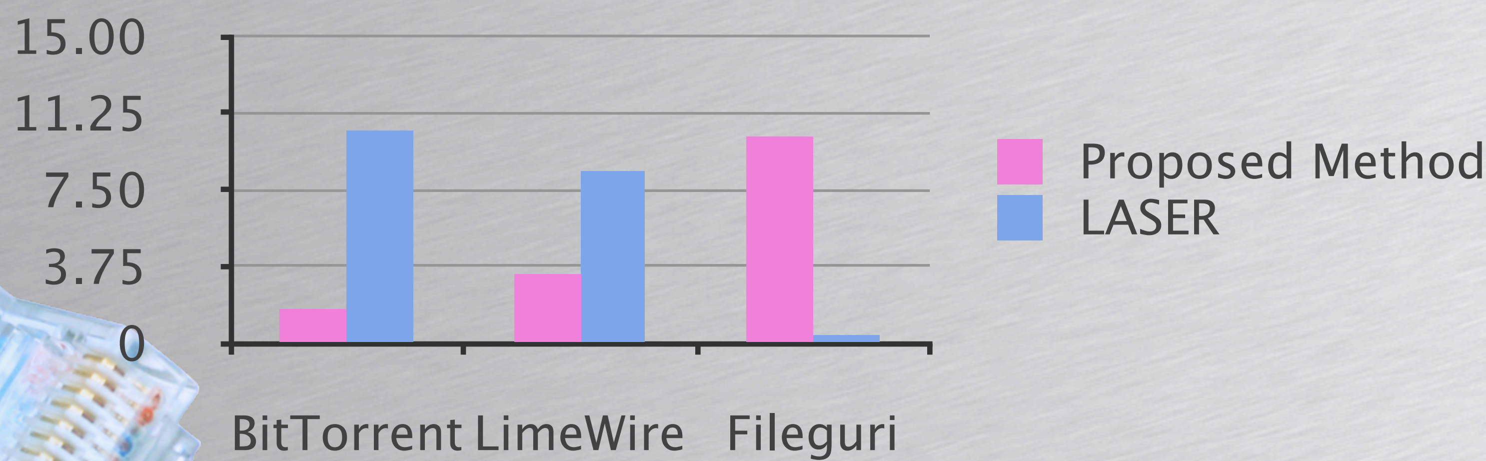
Application	Classified Traffic (kB)	False Negative (kB)	False Positive (kB)
BitTorrent	202,018	3,361	0
LimeWire	87,678	2,951	0
FileGuri	95,804	9,691	0
YouTube	16,061	0	3,775
TMA Log Traffic		421,339 kB	

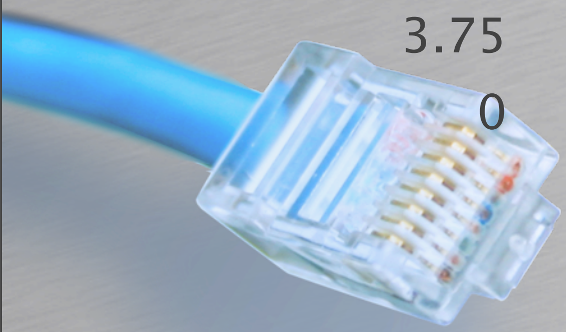
HTTP packet contents	YouTube signal packet contents
GET / HTTP/1.1 User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US) … … Connection: Keep-Alive	GET/videoplayback?sparams=id%2Cexprie %2Cip%2ipbits% … HTTP/1.1 User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US) … Connection: Keep-Alive

Proposed Method vs. LASER

- Accuracy comparison with our earlier work (LASER, automated signature generation system)

	Proposed Method	LASER
Overall Accuracy	96.01%	97.93%



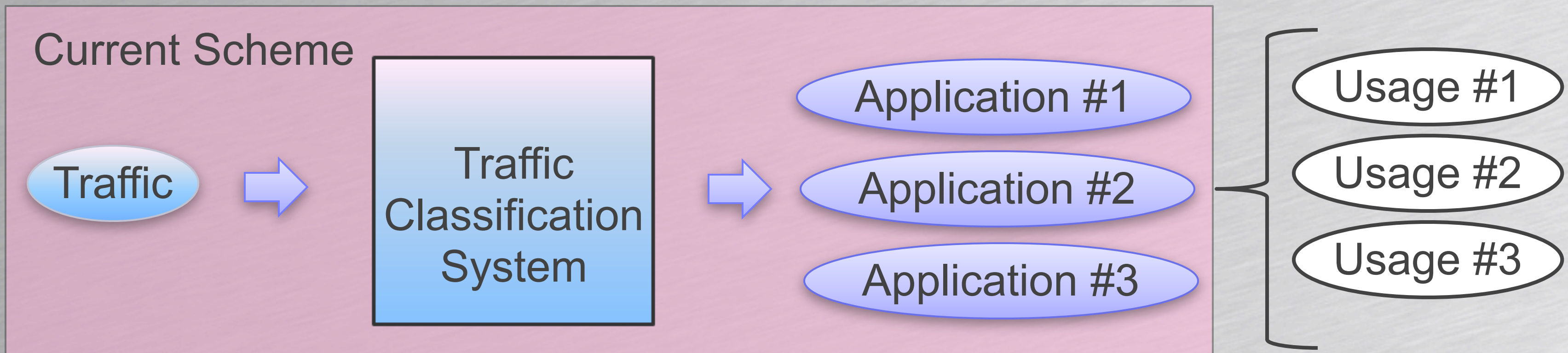


Summary

- New traffic classification approach
  - Converting payloads into vector representations
  - Document classification approach to traffic classification
  - Accuracy analysis on representative target applications in the real traffic
- Contribution
  - No more exhaustive search for payload signatures
  - Achieving simplicity – simple numerical representation of similarity in traffic classification
- Strength
  - Accuracy of classification result was almost same with signature-based classification result (overall accuracy: 96%)
  - Similar to unsupervised ML (clustering) with low complexity
- Weakness
  - Manual parameter adjustment
  - Scalability problem (efficient for small number of target application)
  - Vector and matrix conversion are required

What is Next Step?

- Fine-grained traffic classification
  - Current traffic classification schemes are only able to discriminate broad application classes or application names



  - One application generates different types of traffic (e.g., P2P: searching, downloading, advertising, messenger, etc)
  - Fine-grained traffic classification can be used for extracting information about application usage

- Need a new methodology to classify certain application’s traffic according to usage of the traffic

Proposing New Approach

- LASER + Flow similarity
  - Stage 1: Preprocess network traffic using 'flow similarity' to classify usage types of traffic
  - Stage 2: Extract application signatures from flows which are grouped by 'flow similarity'
- Types of traffic generated by a network application (especially P2P app.) are limited
- Flow similarity might efficient for classifying types of network flow (without scalability problem)
- Combining two methods can enable to generate application signature fully automated manner

Conclusion

- Traffic classification using flow similarity
  - Converting payloads into vector representations
  - Utilizing document classification approach to traffic classification
  - Provide soft-classification that is represented as a numerical value ranges from 0 to 1
  - Provide about 95 % classification result regardless of asymmetric routing environment
  - Linear time complexity

- Fine-grained traffic classification
  - **Goal**: Develop a methodology to classify certain application's traffic according to usages of the traffic
  - Fine-grained traffic classification can be used for extracting information about application usage
    - Top n applications → Top n operations
  - **Approach**: combining LASER and document classification methodologies

Q&A