SWINBURNE UNIVERSITY OF TECHNOLOGY

Rapid Identification of BitTorrent Traffic

Jason But, Philip Branch and Tung Le

jbut@swin.edu.au

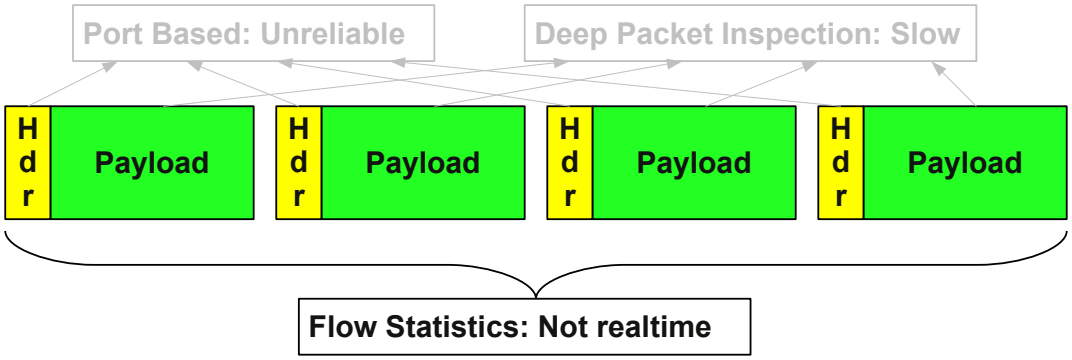
Centre for Advanced Internet Architectures (CAIA)
Swinburne University of Technology

Outline

- BitTorrent and Traffic Classification
- Traffic Observations
  - Statistical feature sets
- Classification
- Performance and classification timeliness

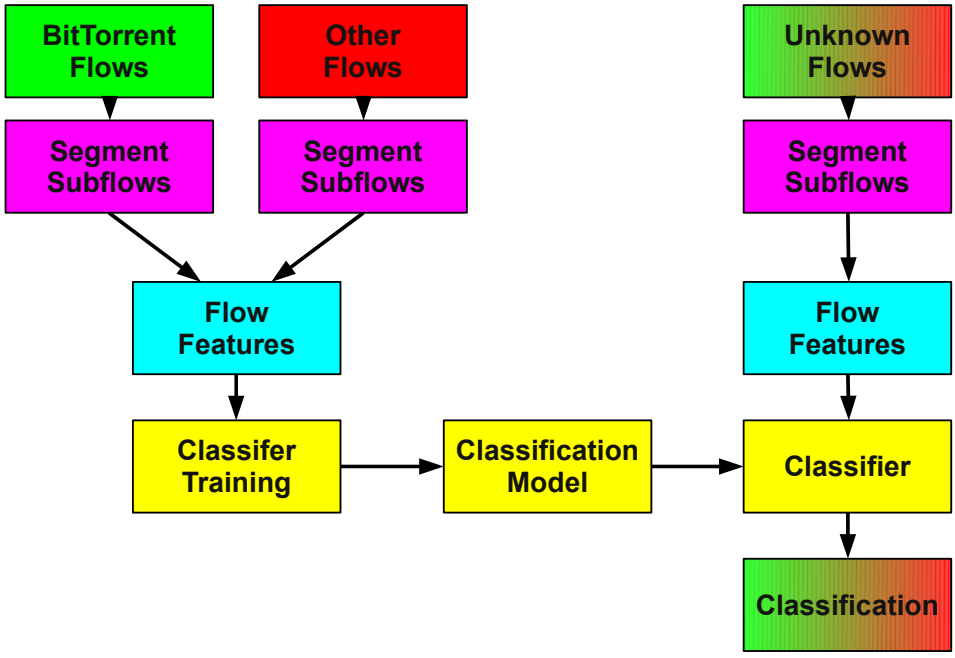
BitTorrent – Classification

- BitTorrent (and other p2p) forms the bulk of traffic on the Internet
- Classification can provide benefits to network operators and users:
  - Rapid classification can allow better management of BitTorrent traffic
  - Possible legal requirements
  - Improved service can be offered to other – interactive – traffic
- Existing techniques suffer from problems:



Rapid Flow Classification

Subflow Classification



- We train the classifier to detect sub-flows

BitTorrent – Observed Data

Common Packets

Interest and *Unchoke* packets Fixed payload length of **5** bytes

Request Piece and *Cancel Piece* packets **17** byte payload

Generic Packet Sizes

- The BitTorrent protocol performs two functions
  1. Transfer data – *large packets*
  2. Update status – *small packets*
- BitTorrent exhibits packets of both types

Data Flow

- Centralised transfer applications are typically uni-directional
- For a p2p protocol, we expect traffic flow to be bi-directional

Classification Features

r_{cbt} – Characteristic BitTorrent Packet Ratio

- A Characteristic BitTorrent Packet is one with a payload of **5** or **17** bytes
- r_{cbt} = Ratio of Characteristic BitTorrent Packets to total packets within a (sub)flow

r_{small} – Small Packet Ratio

- A Small Packet is one with a payload of less than **40** bytes
- r_{small} = Ratio of Small Packets to total packets within a (sub)flow

Classification Features

r_{large} – Large Packet Ratio

- A Large Packet is one with a payload greater than **1350** bytes
- Based on Ethernet maximum segment size
- r_{large} = Ratio of Large Packets to total packets within a (sub)flow

σ_{small} – Smaller Payload Standard Deviation

- Calculate the standard deviation of the TCP payload size for packets flowing in each direction
- If traffic flow is uni-directional, one of these values will be very small
- σ_{small} = Smaller of the two calculated standard deviations

Traffic Features

Traffic Traces

BitTorrent	Captured from 4 swarms of up to 40 peers connected at ADSL1-like line rates, over 38,000,000 packets
Other	University of Twente Public Traffic Trace[1]
FTP	Captured distraction traffic – nearly 1,000,000 packets from about 1,000 flows

- Our analysis indicates that these features differ
  - Some features differentiate only two traffic types
- Features appear to differentiate BitTorrent (p2p) from other bulk transfer protocols (FTP)
- Features hold validity over sub-flows

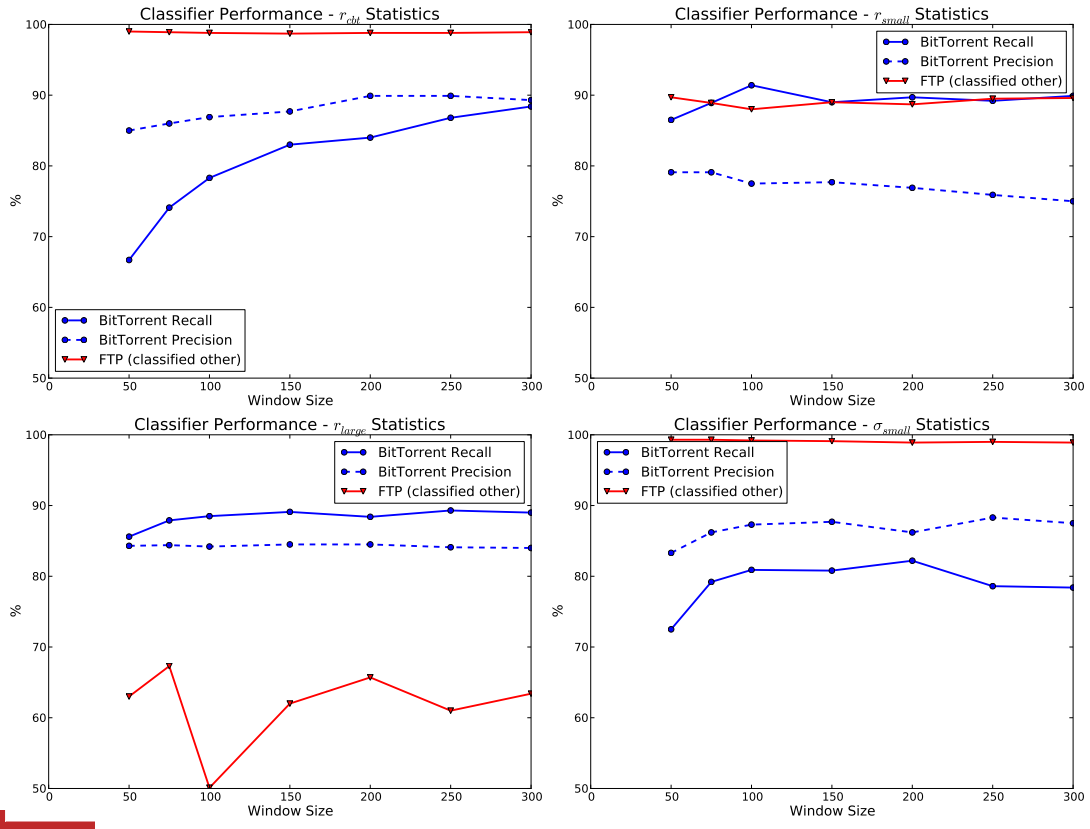
[1] Available at: `http://traces/simpleweb.org`

Traffic Classification

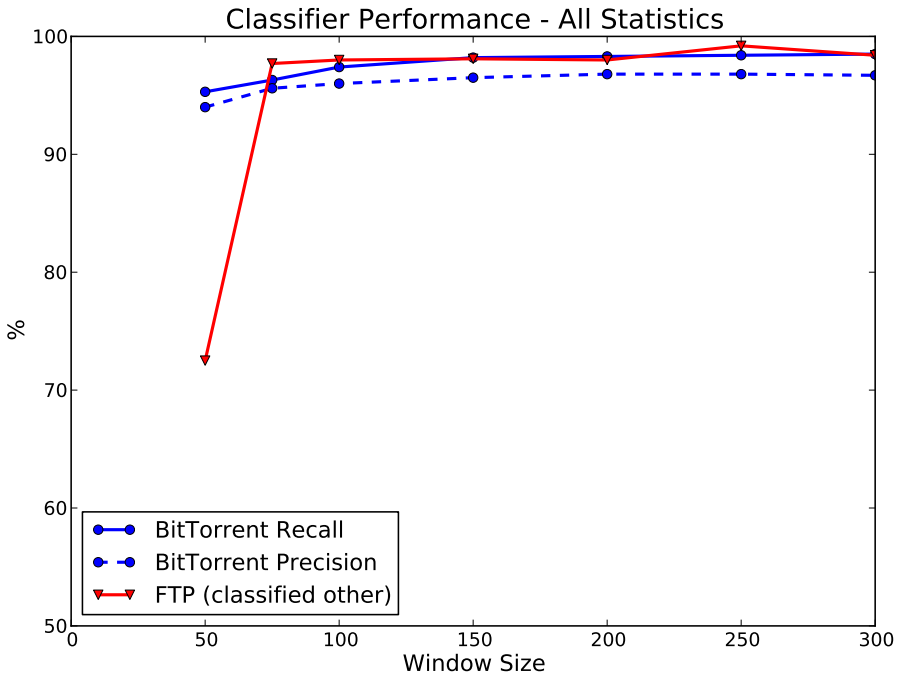
- We calculated these four features over our data set for a number of sub-flow sizes
- We used the WEKA implementation of C4.5[2] to train and test the classifier
  - Decision Tree based classification
  - Standard 10-fold Cross-Validation test
- Used captured FTP traffic as distractor traffic
- Results for whole-of-flow classification very good – see paper for details
- Tested for all sub-flow sizes and each combination of features

[2]N. Williams, S. Zander and G. Armitage, "*A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification*", **ACM SIGCOMM Computer Communication Review**, vol. 36 no. 5 pp. 7–15, October 2006

Classifier Performance – individual features



Classifier Performance – all features



Classifier Performance – 150 packet subflows

All four features combined

Classification Recall		
BitTorrent Recall	BitTorrent Precision	FTP (Recall as other)
98.2%	96.5%	98.1%

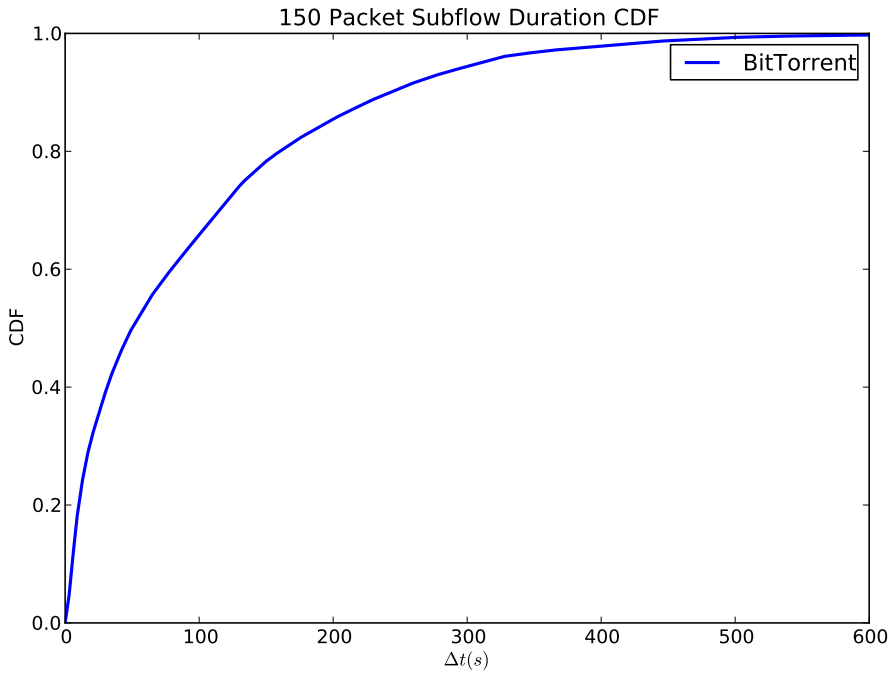
Excluding r_{cbt} feature

Classification Recall		
BitTorrent Recall	BitTorrent Precision	FTP (Recall as other)
97.5%	93.7%	97.6%

- Minor drop in performance
- Expected to be more robust for protocol changes or deliberate attempts to circumvent detection

Classification Timeliness

- How long does it take to capture 150 packets for classification?



Conclusions

- Existing BitTorrent classification schemes are either non-scalable or use properties that preclude real-time classification
- We present four features – r_{cbt}, r_{small}, r_{large} and σ_{small}
- Suitable sub-flow features to allow for rapid classification
- Using an ML-based C4.5 classifier and these features:
  - Can classify entire BitTorrent flows with 98.9% Recall and 97.9% Precision
  - 150 packet sub-flows – 98.2% Recall and 96.5% Precision
  - Ignoring r_{cbt} – 97.5% Recall and 93.7% Precision
- Analysis indicates that 150 packets subflows would typically be captured in under 3 minutes at ADSL like line rates
- Since these features are based on packet sizes, we expect this classifier to be robust for:
  - Encrypted BitTorrent
  - BitTorrent over UDP