Using Machine Learning to Identify Realtime Traffic Classes

Philip Branch
Centre for Advanced Internet Architectures
Swinburne University of Technology
Melbourne, Australia

Background of Philip Branch

- Mix of industry R&D and academia
- Software engineer in Tasmania and NSW during the 80s
- University of Tasmania and Research Data Networks CRC (based at Monash University) early to mid-90s
- Late 90s, early 2000s worked for an Internet startup as a Business Analyst and for a large trans-national telecommunications equipment supplier as Development Manager for Lawful Interception systems
- PhD (Monash, 1999) Computer Systems Engineering
  - Interactive networked multimedia

Current Research Interests

- First person shooter game traffic
- Wireless networks
- Skype over WLAN
- Covert channels
- Software evolution
- Machine learning to identify realtime traffic classes

Machine learning to identify realtime traffic classes

- Goal is to identify in a reliable and robust manner traffic class
  - Motivation is Lawful Interception
  - Agencies often only interested in the fact that two parties are communicating, not the content of communication
- Has applications elsewhere
  - Quality of Service provisioning
  - Internet application statistics gathering
- Technique is to segment training flows into short (a few seconds) of sub-flows
  - Use statistics calculated on training sub-flows to train a classifier
  - Test on sub-flows extracted from other flows of the same class
  - Classifier used is Naïve Bayes or J48 to produce a classifier tree

Successes so far

- We have shown that it is possible to identify Skype and Bitttorent using machine learning techniques by observing only a part (a few seconds) of the flow
  - Better than 98% reliability in both cases
  - Use the characteristics of the traffic flow (packet lengths, inter-arrival times) as features for identification
- However there are limitations in the way that we have done this
  - Primarily make use of ‘characteristic packet lengths’
  - These can change very easily with different releases (eg Skype v3.0 to v4.0)

Would like a robust way of identifying traffic classes

- What characteristics of (say) peer to peer VoIP are unlikely to change from release to release?
- Investigating statistics associated with packet lengths and interarrival times as a basis for robust classification of traffic
  - Realtime traffic packet lengths have specific timing requirements
    - Usually a trade-off between packet size efficiency (the larger the better) and delay (the more samples per packet, the greater the delay)
  - Asymmetry
    - Some traffic types, such as games and voice with silence suppression are naturally asymmetric
  - Autocorrelation
    - How self-similar is the traffic?

Some early results …

- Excellent results for classifying Games, G.729, Skype, Data transfer within versions using these statistics

Class	Precision	Recall	F-Measure
VoIP (G.729)	0.993	1.000	0.997
Skype	0.994	0.957	0.975
Non-Real-Time Data (UofT)	0.997	0.998	0.999
Game (ETPRO)	0.989	0.997	0.993

- Currently working on distinguishing Skype and Games across versions
  - Train on one game (eg Quake3) and recognise another (eg ETPRO)
  - Train on one version of skype and test on another

Some early results of version independent classification…

- Results for version independent classification using autocorrelation measures only:

Trained on Skype v3, quake3, hl2cs. Tested on Skype v2, hl2dm, etpro

Confusion matrix:

Skype	Game	Classified as
0.84	0.16	Skype
0.03	0.97	Game

Trained on Skype v3, quake3, hl2cs. Tested on Skype v4, hl2dm, etpro

Skype	Game	Classified as
0.81	0.19	Skype
0.09	0.91	Game

Further work

- Incorporate other statistics into cross-version classification
- Optimal subflow length for training and testing
- Other applications
  - Google talk (gtalk) another VoIP application
- Other traffic classes
  - Interactive video
- Application of technique to other areas
  - Quality of Service provisioning