Early Application Identification

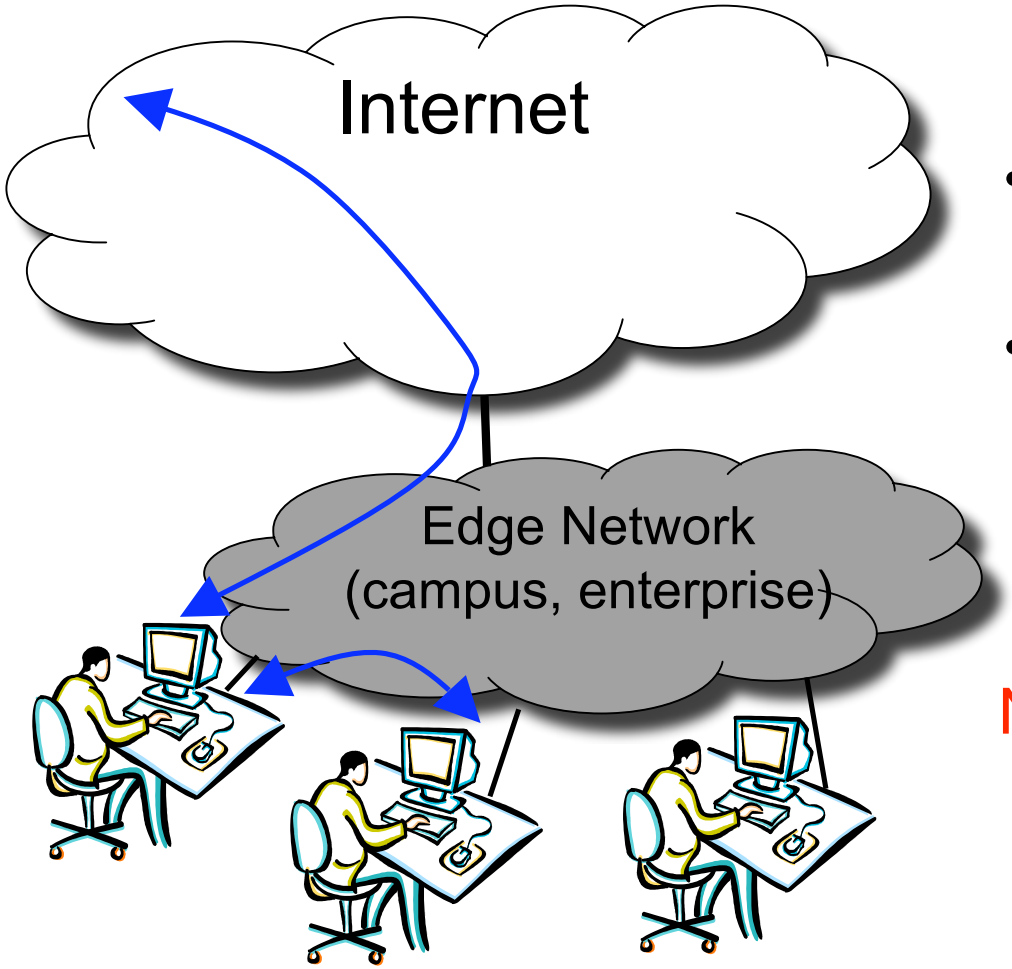
Laurent Bernaille

Renata Teixeira

Kave Salamatian

Université Pierre et Marie Curie - LIP6/CNRS

Which applications run on my network?



- Capacity planning
  - Profiling of network usage
- Security
  - Am I being attacked?
- Institutional policy
  - No illegal music sharing
  - No Skype traffic

Need to detect the application associated with network traffic as fast as possible

Problem statement

- A new classifier
  - Inspects all packets entering and exiting the network
  - Only uses information in the TCP/IP header
  - Identifies the application using the first few application packets in TCP connections

Application identification today

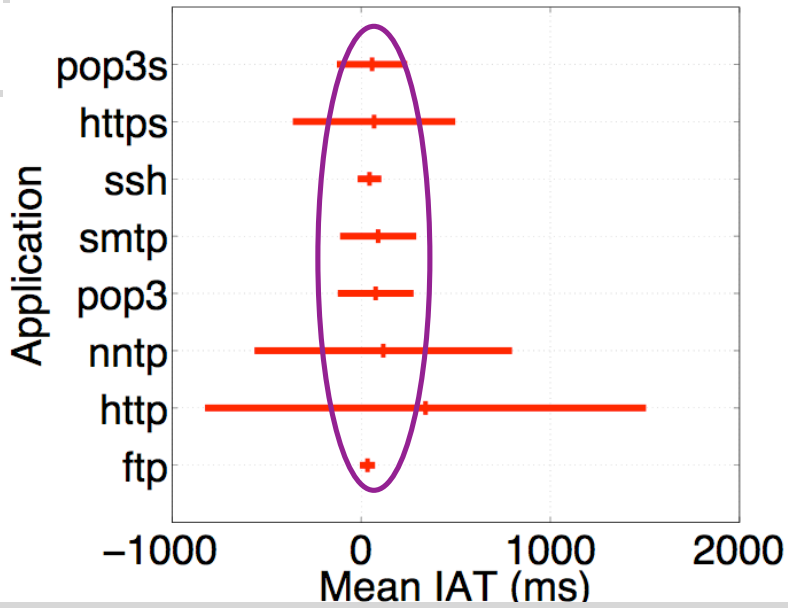
- Port-based identification
  - Use IANA well-known port numbers
  - Often inaccurate
    - Non-standard ports
    - Masquerade traffic
- Content-based identification
  - Inspect packet payload for well-known signatures
  - Very accurate, but…
    - Does not work for encrypted traffic
    - Privacy issues
    - Computationally intensive

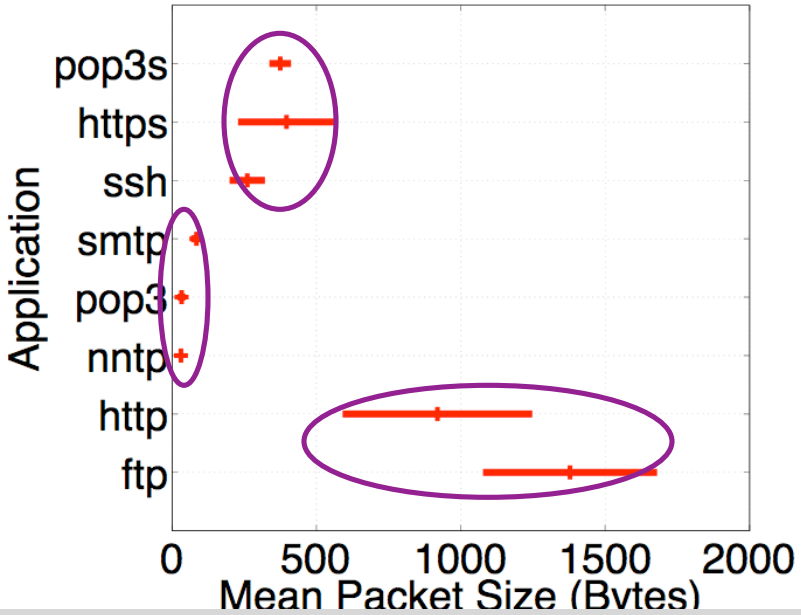
Behavior-based application identification

- Newer proposals based on connection-level statistics
  - Mean or variance of packet inter-arrival time
  - Mean or variance of packet size
  - Number of packets in the connection
  - Duration of a connection
- Accuracy: ~90%
- Need to wait for connection to end

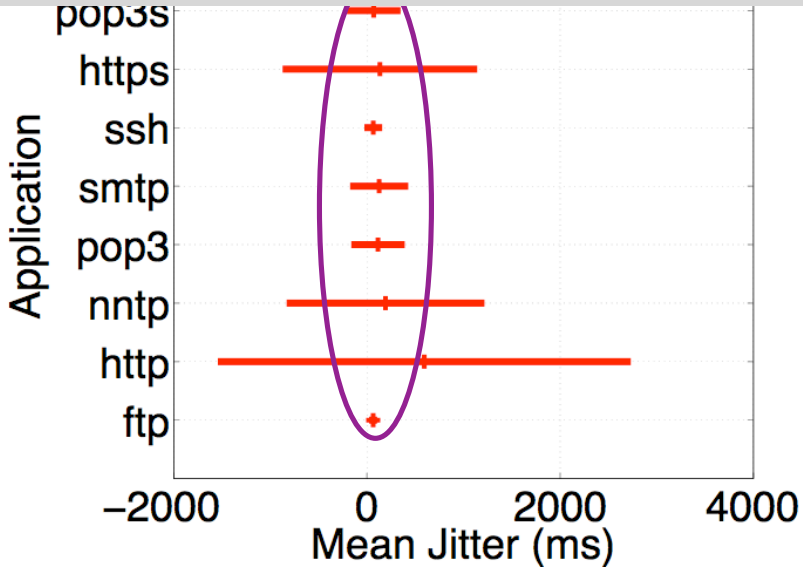
Can we apply similar techniques just to the first packets of a connection?

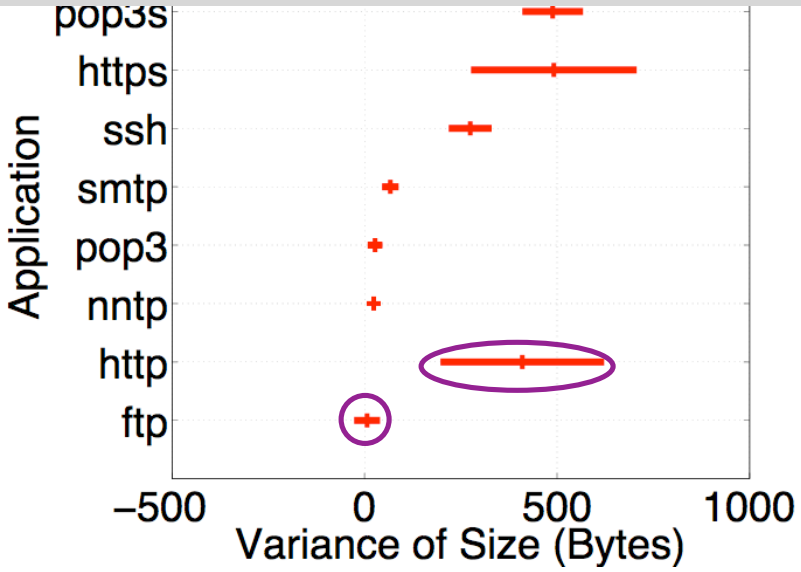
Statistics for the first four packets



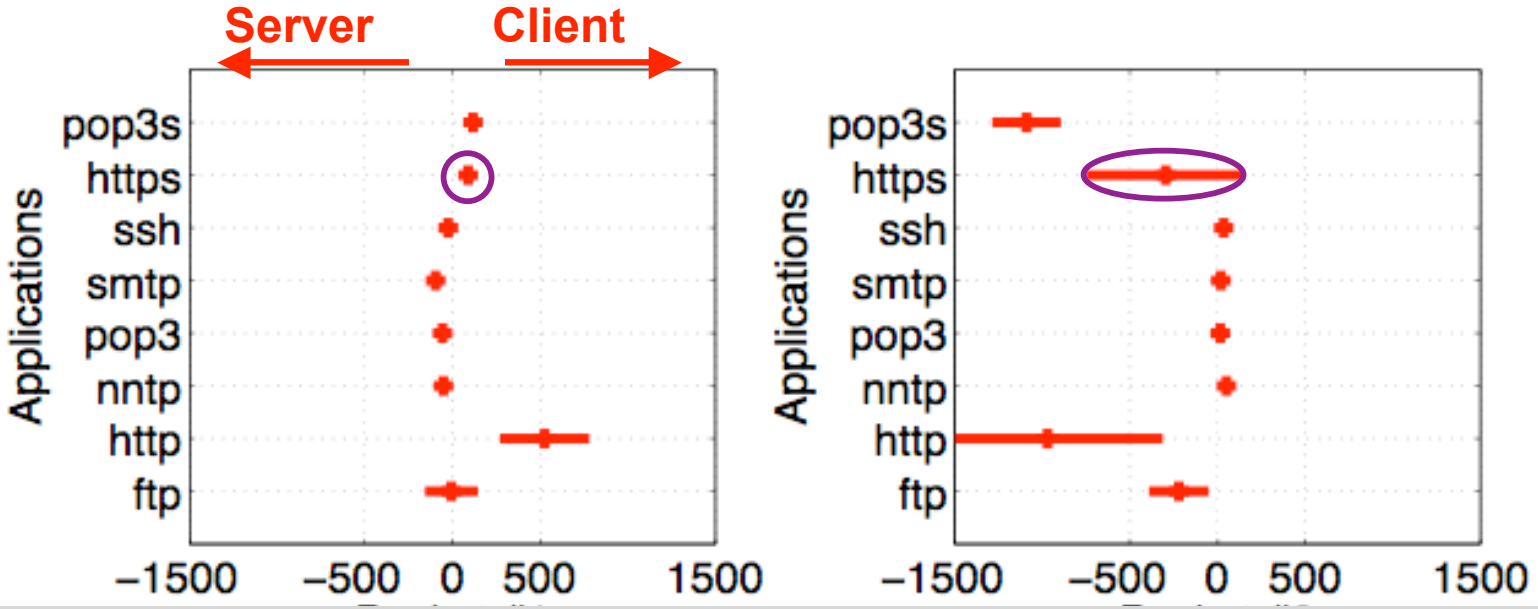


The size of the first packets helps distinguish applications

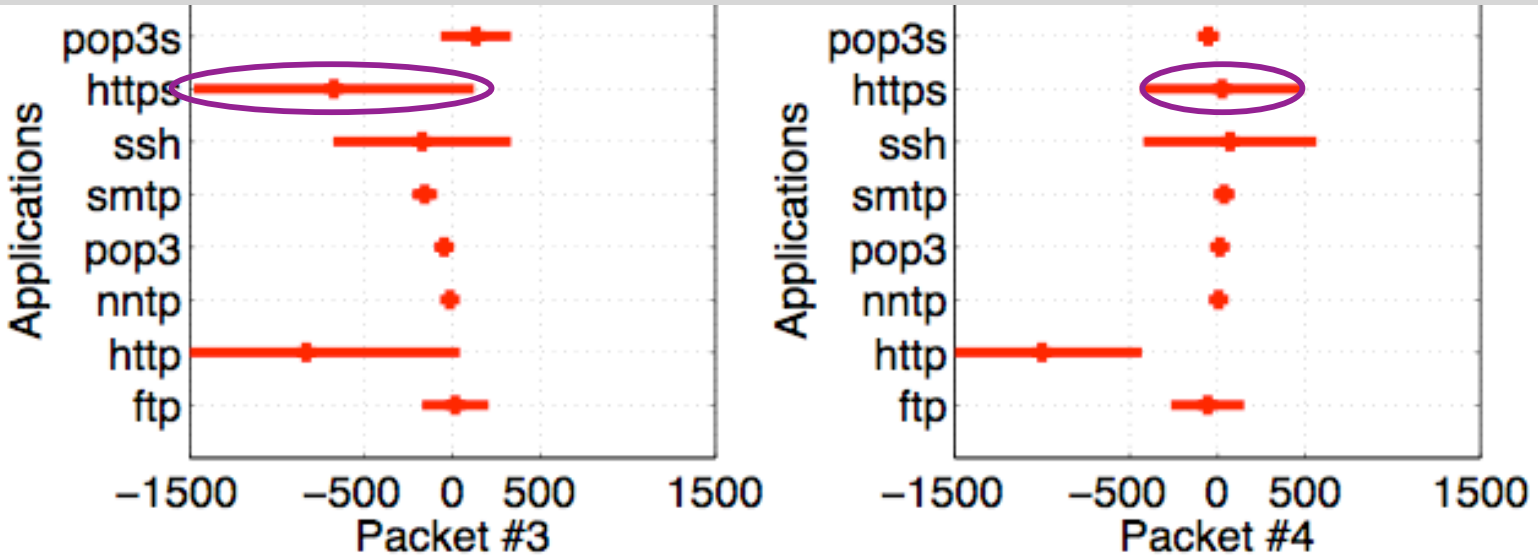




The size of each packet adds information



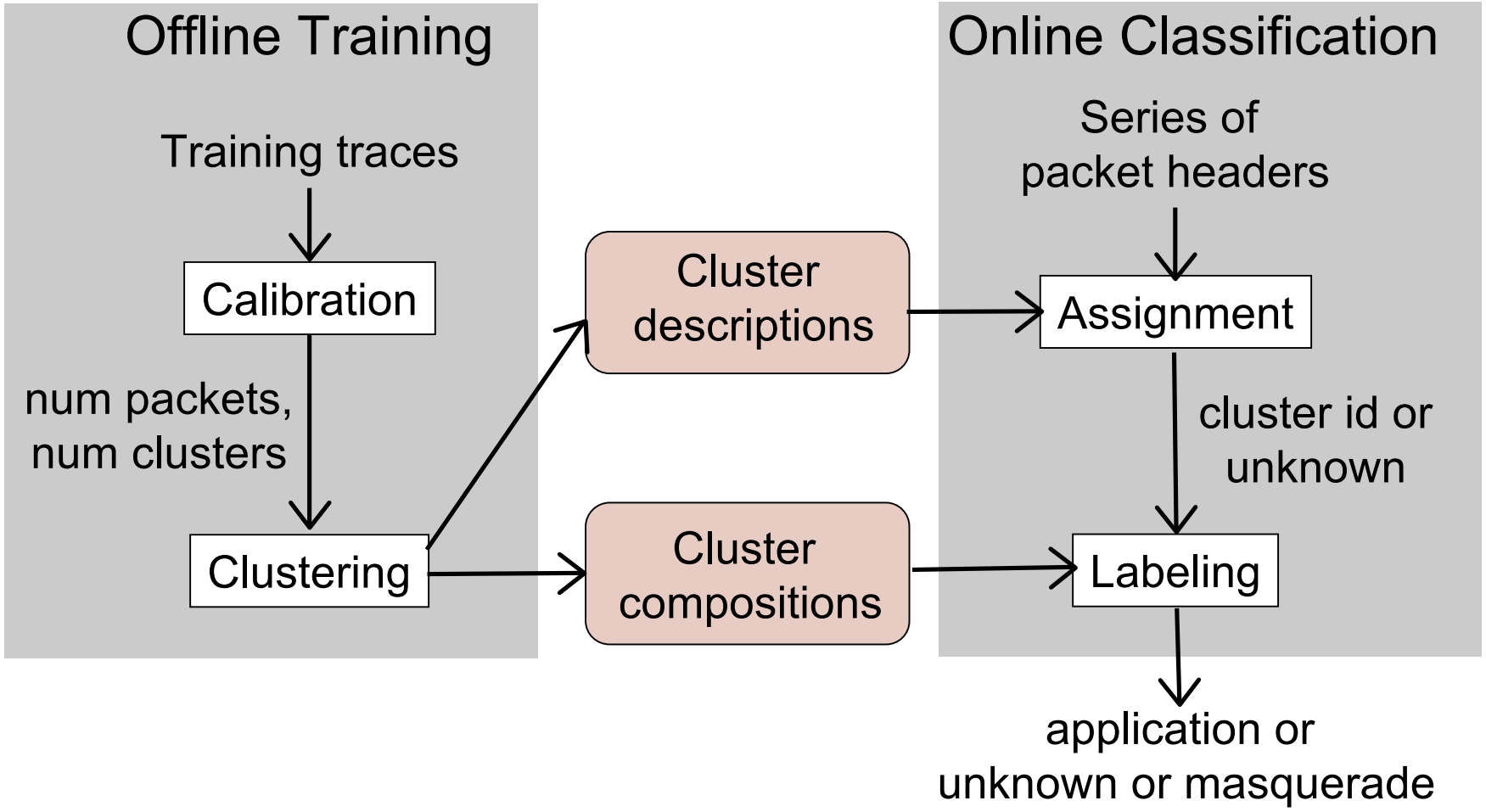
Why: Initial packets contain the application negotiation



How to model applications with first packets?

- Model behaviors, instead of applications
  - An application can have multiple modes of operation
  - A behavior is a sequence of the sizes of the first application packets
  - Several applications may share a behavior
- Represent behavior of TCP connections
  - Euclidean space: each dimension is the size of a packet
  - Hidden Markov Model: each state represents a packet
- Find groups of connections that have similar behavior
  - Using well-known clustering algorithms

Method overview



Training traces

- Requirements
  - Sample connections of all target applications
  - Representative number of samples
  - Equivalent number of samples across applications
- Two methods to obtain sample of connections
  - Extract connections from packet traces
  - Manual generation of traces for each application
- Our training traces
  - 500 connections per application from packet traces
  - NNTP, POP3, SMTP, SSH, HTTPS, POP3S, HTTP, FTP, Edonkey, Kazaa

Clustering algorithms

Representation	Distance	Clustering	Cluster Model
Euclidean	Euclidean	K-Means	Center
		GMM	Gaussian
HMM	Baulm-Welch	Spectral Clustering	HMM

Calibration

- Select number of packets and clusters
  - Compute clustering spanning a vast parameter space
    - Number of packets: [1, 10]
    - Number of clusters: [5, 100]
- Metric of clustering quality
  - Normalized Mutual Information (NMI)
  - NMI = 1: ideal clustering

$$NMI(X,Y) = \frac{MI(X,Y)}{\sqrt{H(X)H(Y)}}$$

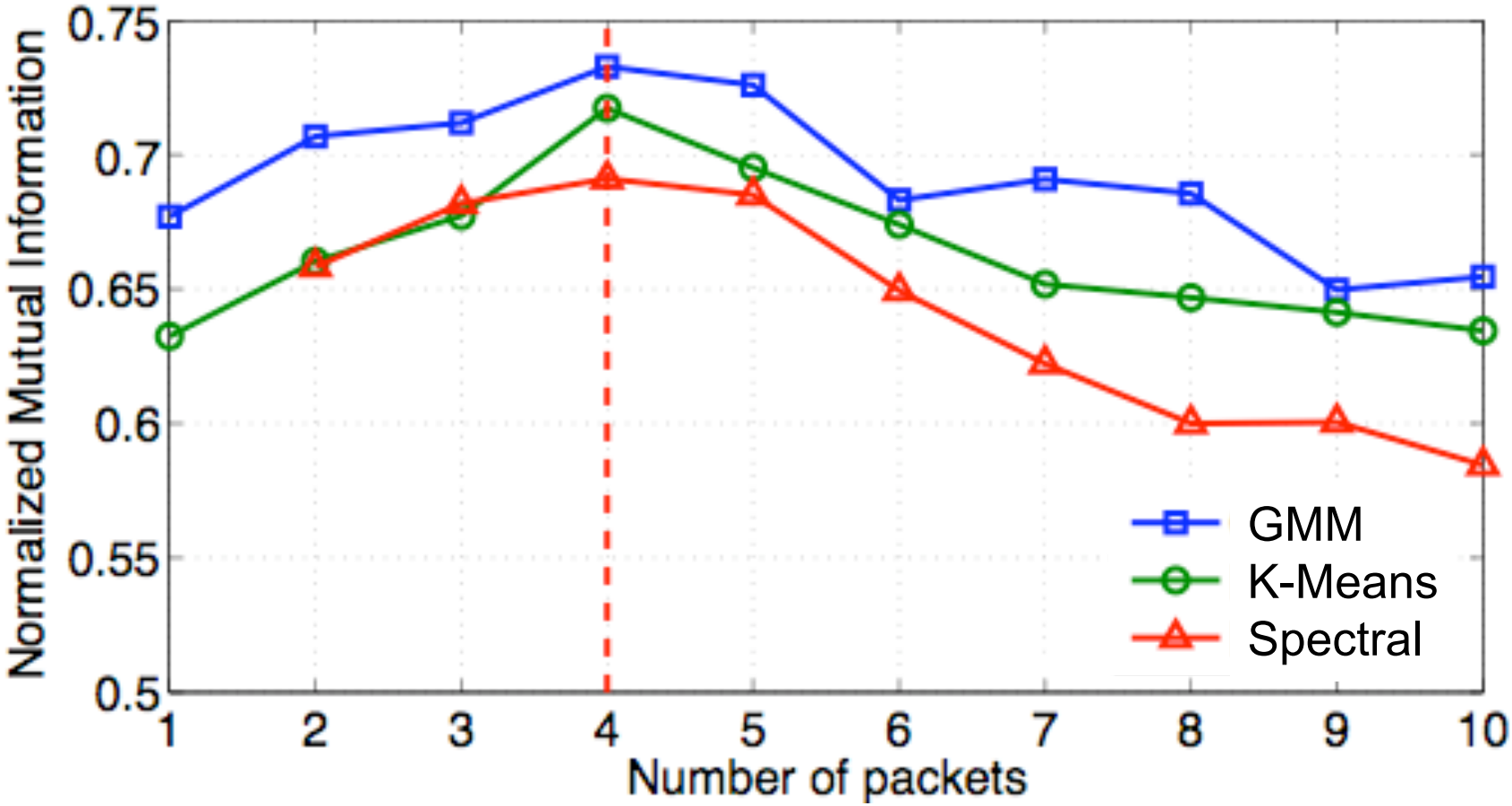
Mutual Information

Normalized by the entropies

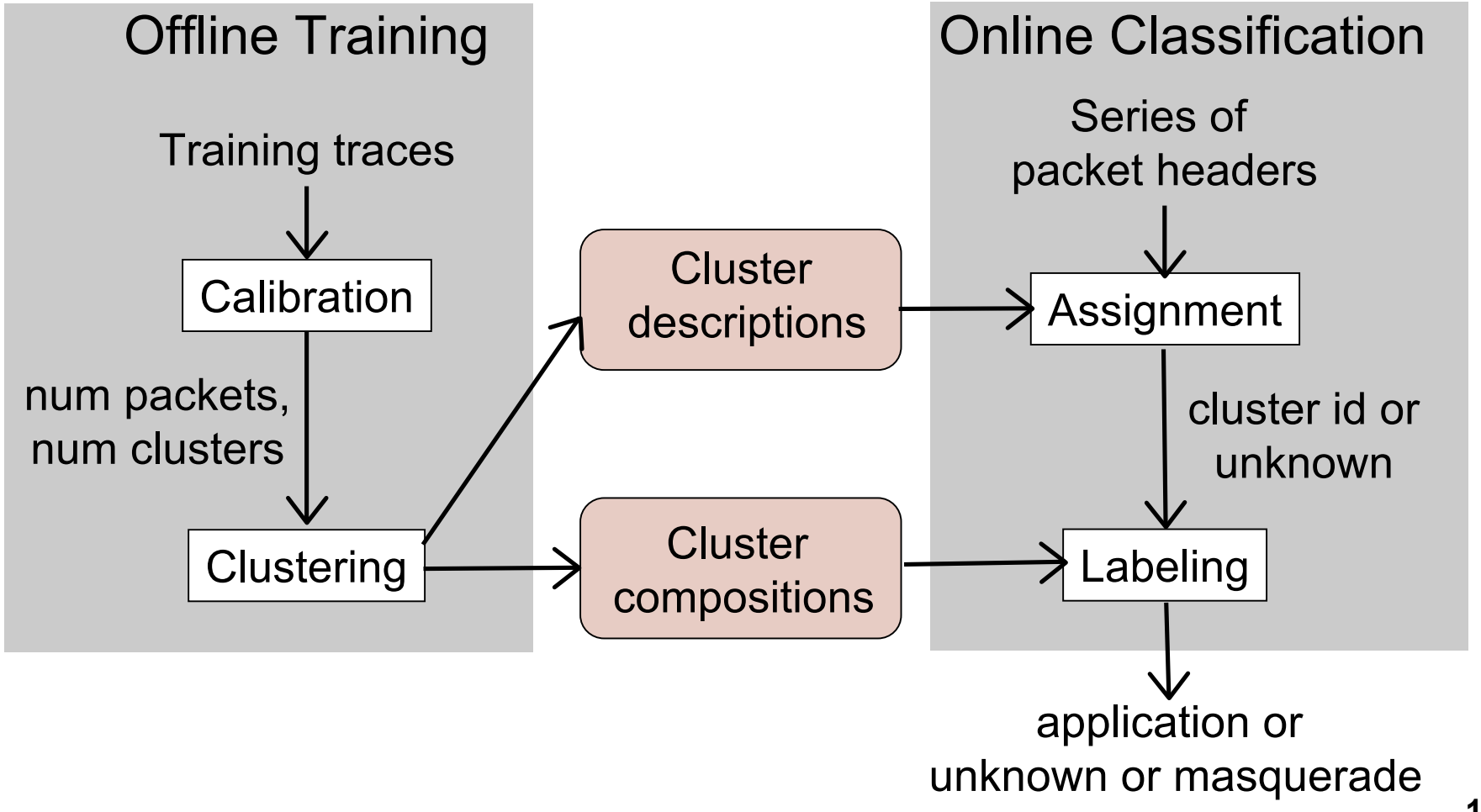
Distribution of applications

Distribution of clusters

What is the best number of packets?

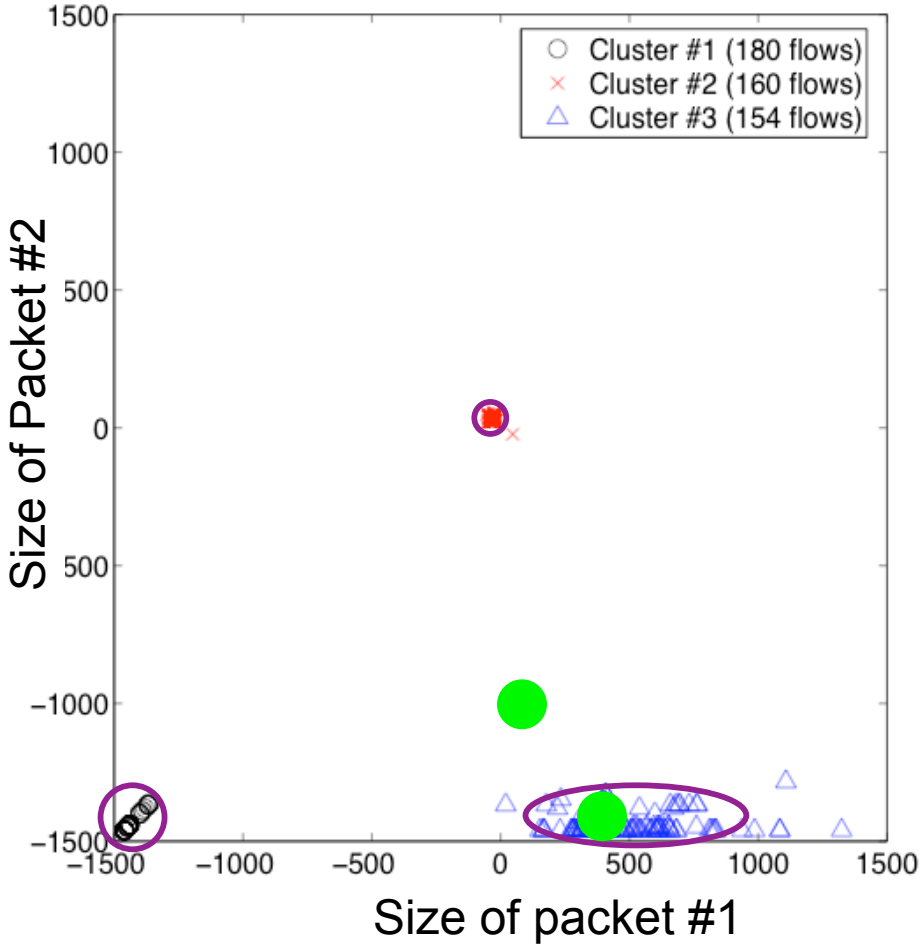


Method overview



How do we assign a new connection to a cluster?

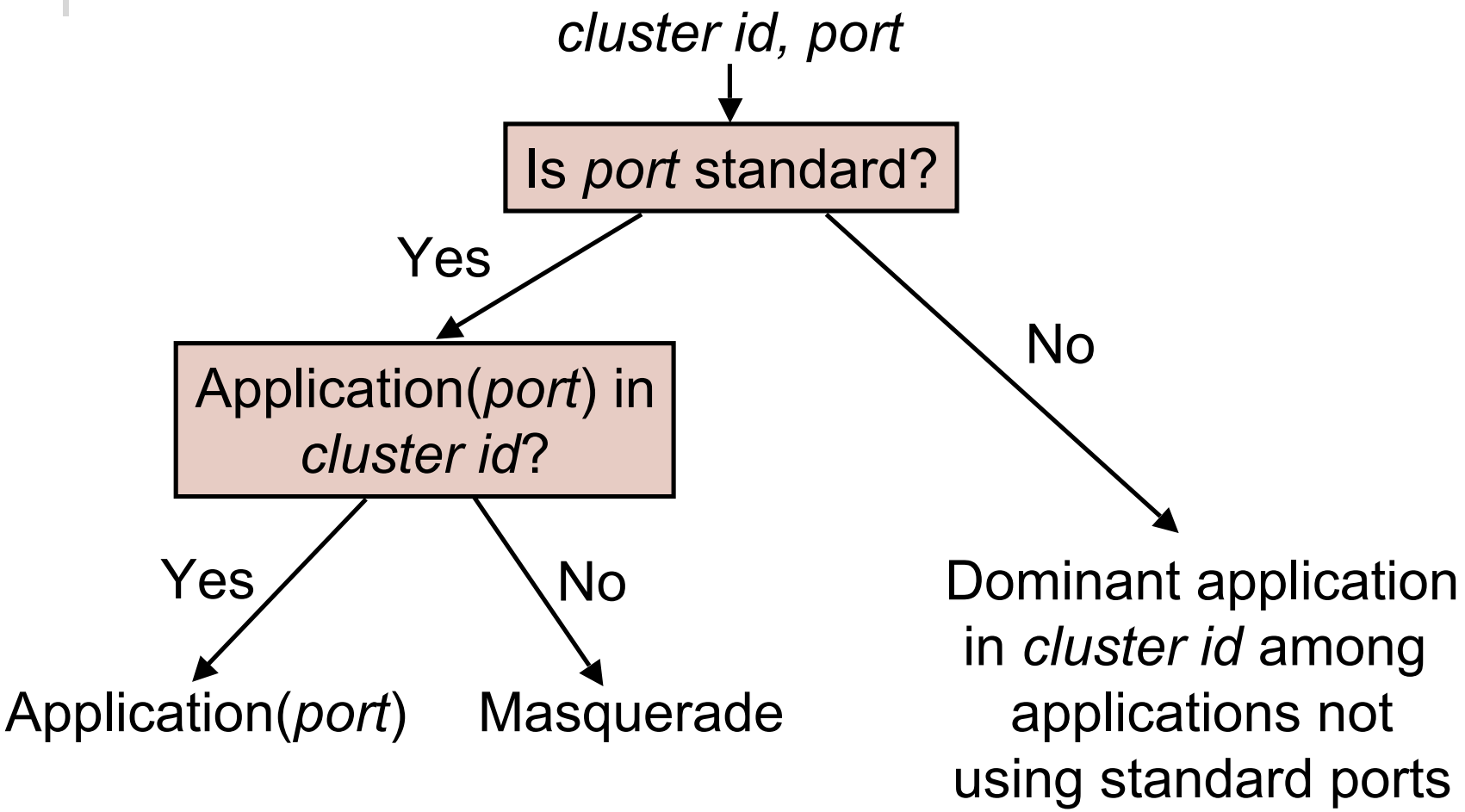
- Maximum likelihood
  - Compute probability that connection belongs to each cluster
  - Select cluster with maximum probability
- With threshold
  - Detection of unknown traffic



How do we label a connection?

- Dominant
  - Label connection with the most common application
  - Problem
    - A cluster contains 80% HTTP and 20% SMTP
      - SMTP connections are misclassified
    - Approximately, 40% clusters with multiple applications
- Cluster+Port
  - Port numbers are meaningful for standard applications
  - Connections assigned to cluster with port 25 are labeled SMTP

Cluster+Port heuristic



Evaluation: Test traces

- Packet traces with TCP payload
  - TCP payload necessary to establish ground truth
  - Paris 6 and enterprise network
  - 50 000 connections
- Manually-generated
  - Applications not in training set to evaluate the detection of new traffic
  - Bittorent, IMAP, Gnutella, IRC, LDAP, MSN, MySQL
  - Between 100 and 10 000 connections

Evaluation: Assignment accuracy

$$\frac{Conn.\ assigned\ to\ clusters\ containing\ their\ app}{Total\ number\ of\ connections}$$

Trace	Accuracy
P6-1	99.6%
P6-2	99.6%
P6-3	99.6%
Enterprise	99.3%

Evaluation: Labeling accuracy

$$\frac{Connections\ accurately\ labeled}{Total\ number\ of\ connections}$$

Heuristic	Trace	Accuracy
Dominant	P6-1	93.5%
Dominant	Enterprise	95.6%
Cluster+Port	P6-1	98.5%
Cluster+Port	Enterprise	99.1%

What about applications not in the training set?

New applications should be labeled “unknown”

Application	Misclassified	Masquerade	Unknown
Bittorent	32.9%	0.3%	66.8%
IMAP	8.6%	57.8%	33.6%
Gnutella	100%	0%	0%
IRC	10%	0%	90%
LDAP	8.8%	0%	91.2%
MSN	38.5%	0%	61.5%
MySQL	0%	0%	100%

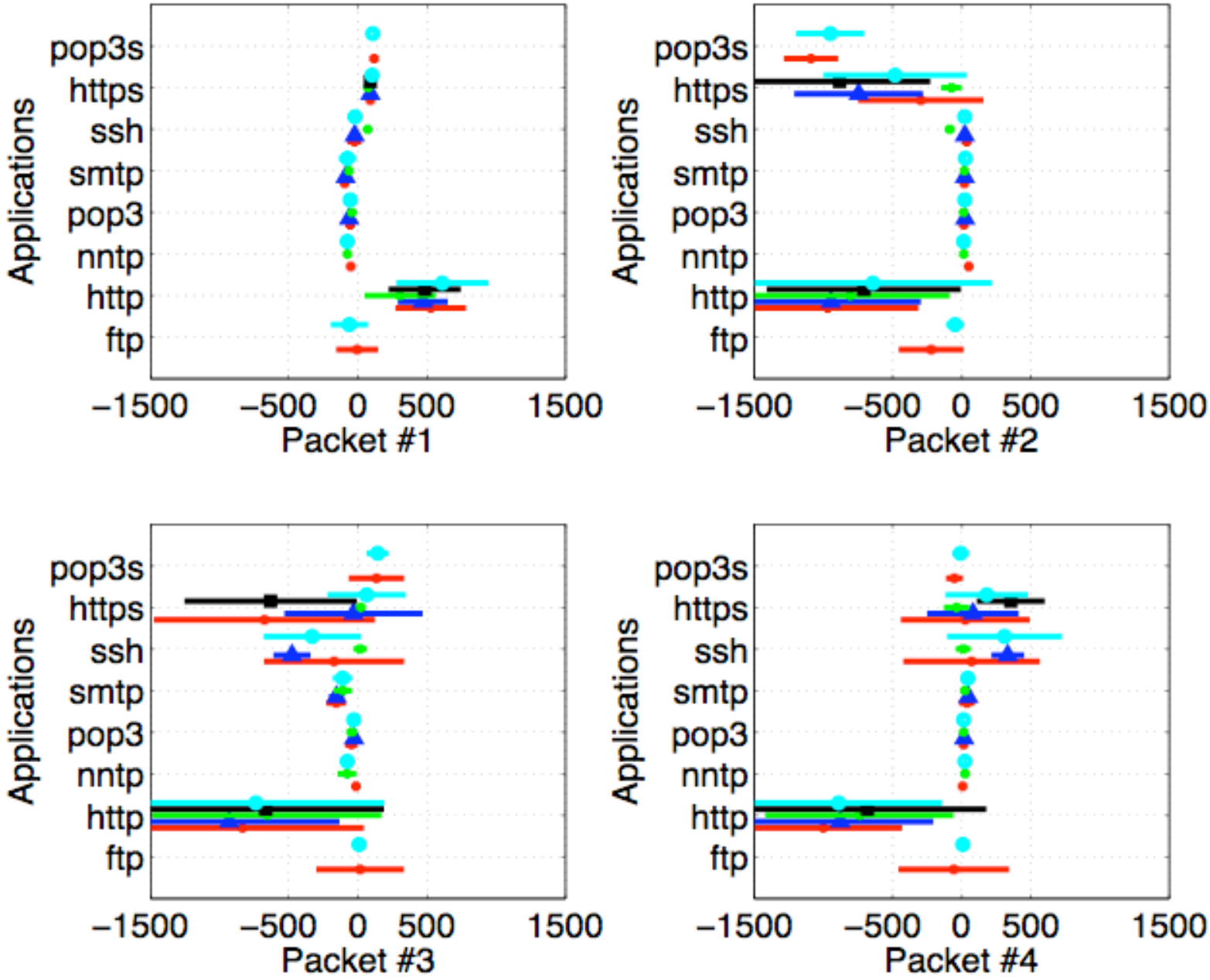
Conclusion

- Exploration of design space for early classification
  - First packet sizes capture application behavior
- New classifier for TCP connections
  - Clustering on size of first four application packets
  - Use of port numbers when relevant
- Ongoing work
  - Refine the detection of unknown traffic
  - Classification of traffic encrypted with SSL
  - Online implementation

More information on our web page

http://rp.lip6.fr/~bernaill/earlyclassif.html

Influence of the network



Normalized Mutual Information

$$MI(X,Y) = \sum_{i,j} p_{i,j} log(\frac{p_{i,j}}{p_i p_j}) \qquad H(X) = -\sum_i p_i \log(p_i)$$

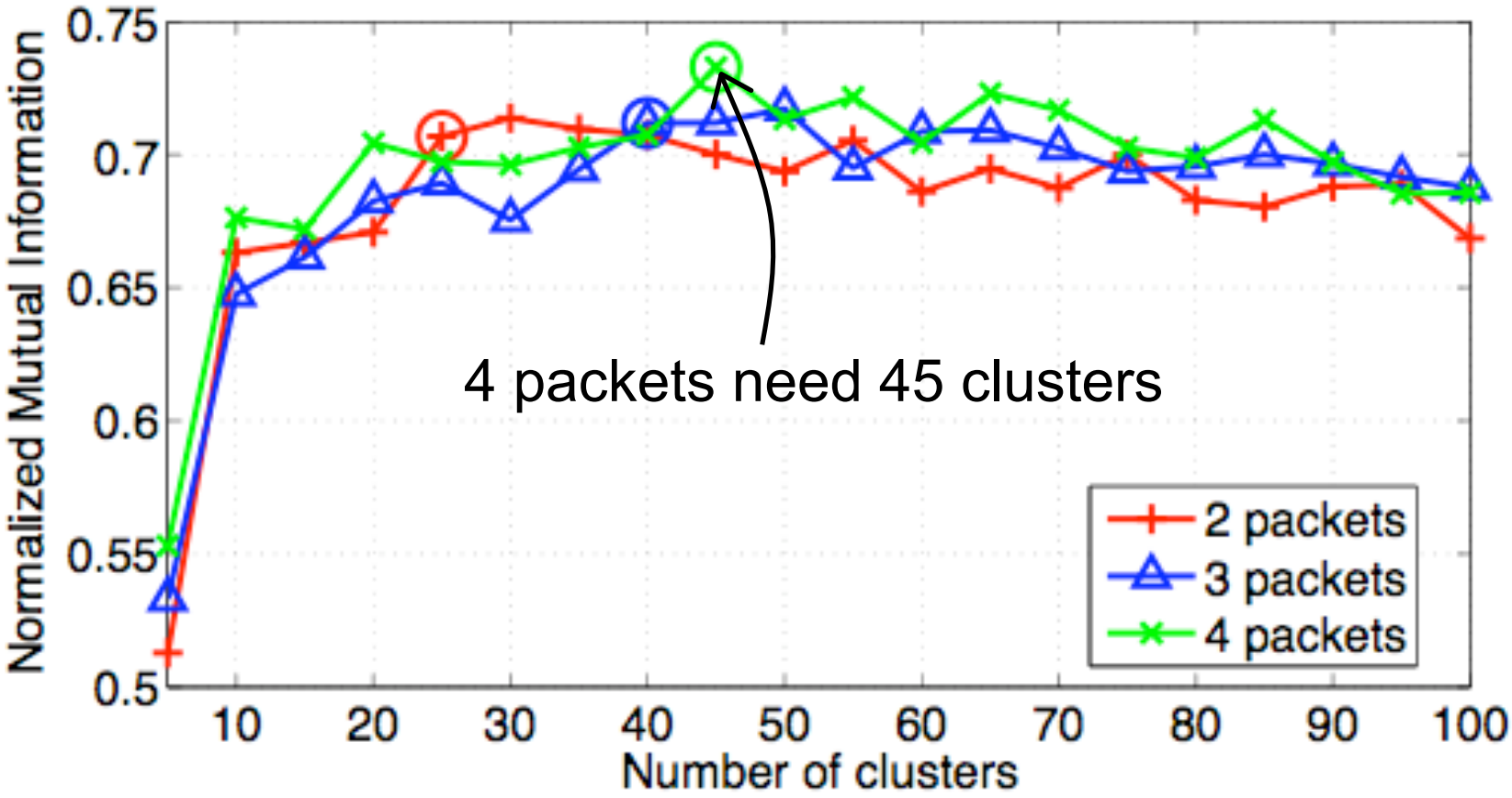
$p_{i,j}$: proba that a connection in cluster j belongs to application i
p_i: proba of application i
p_j: proba of cluster j

MI(X,Y): shared information between X and Y

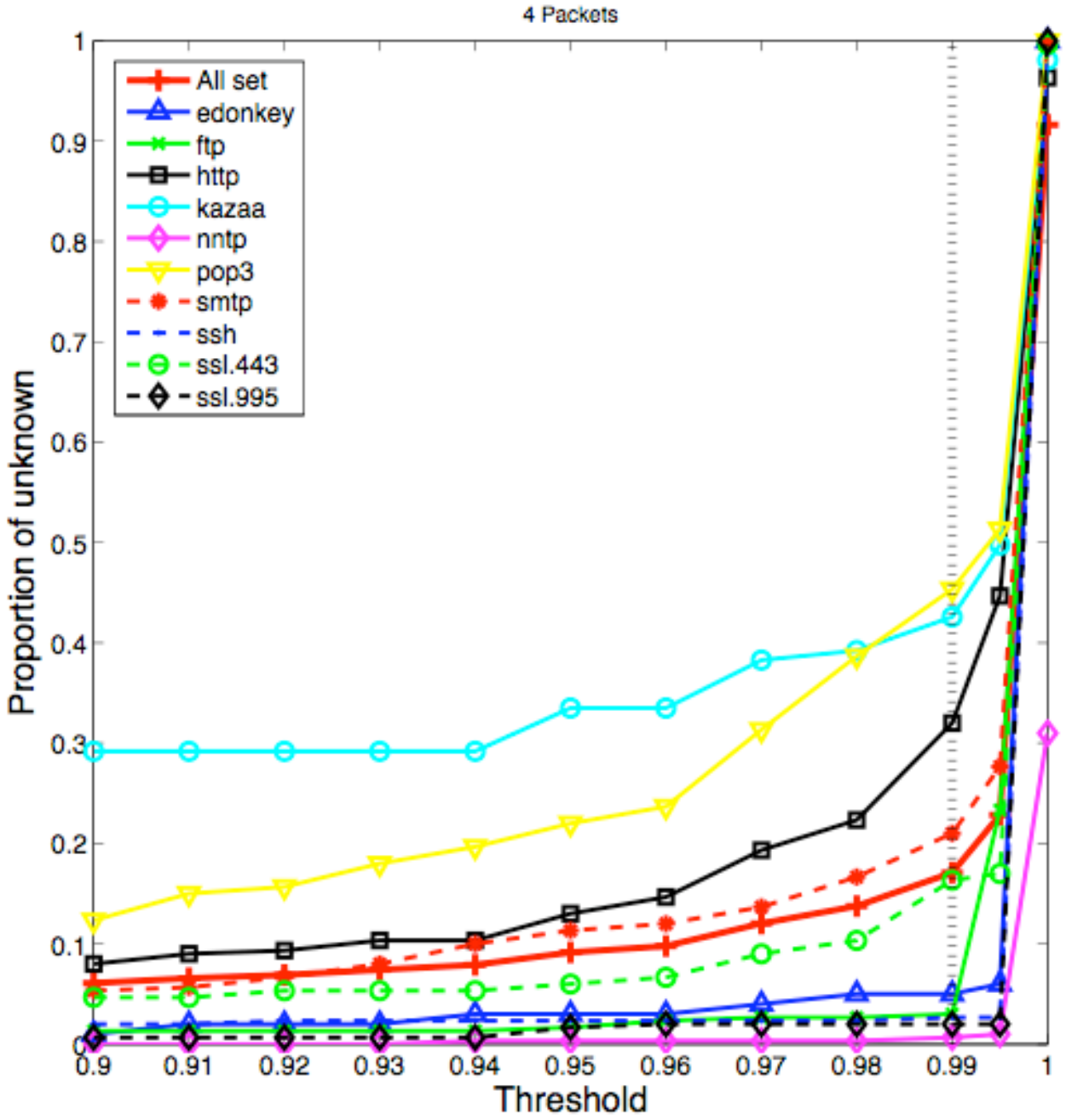
NMI: normalized with entropy: $NMI(X,Y) = \frac{MI(X,Y)}{\sqrt{H(X)H(Y)}}$

0<NMI<1, NMI=1 <=> one to one mapping between X and Y

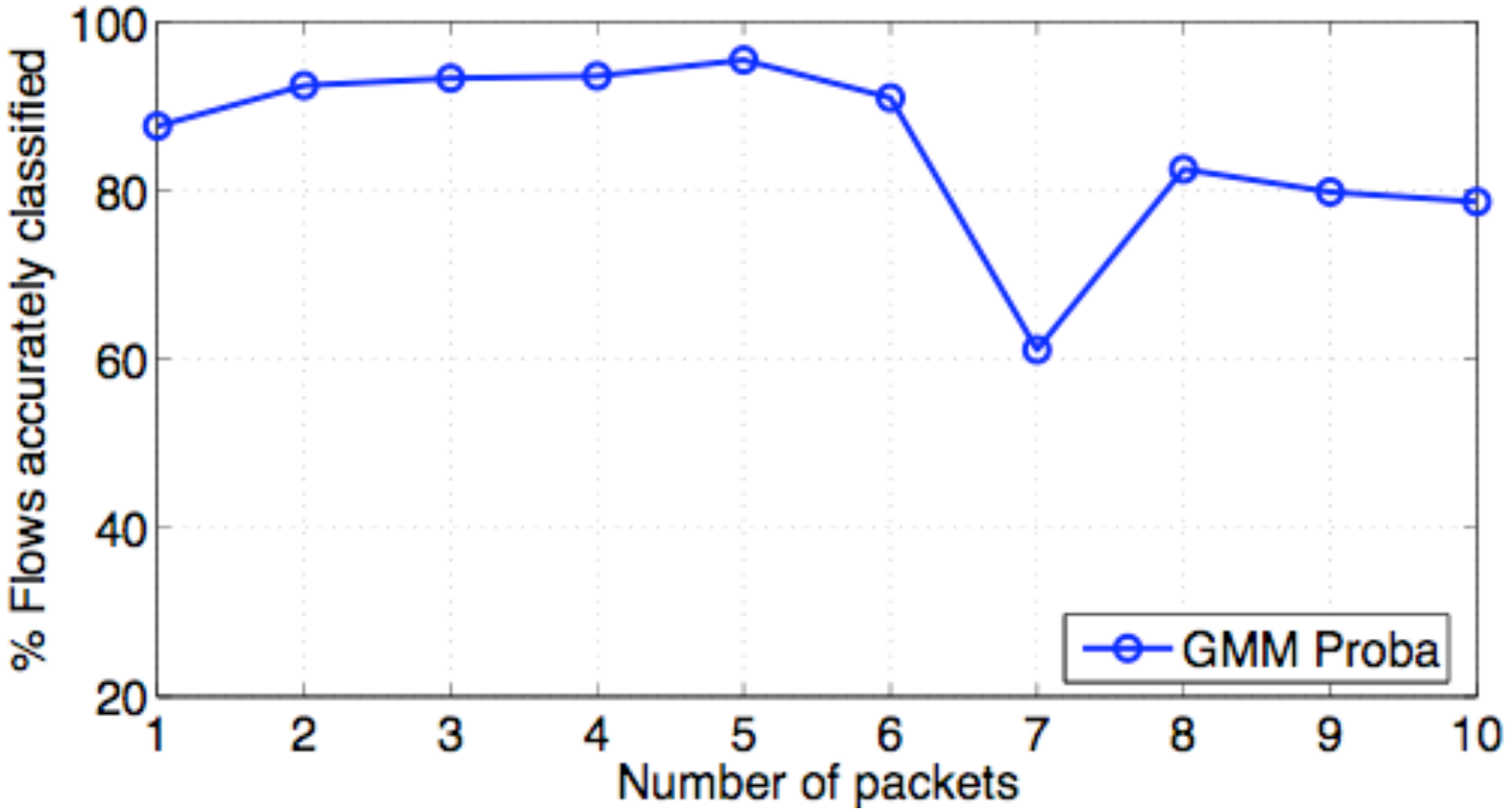
Number of clusters



Threshold Choice



Overall Accuracy



Detection on unknown traffic

		TP	FN	Masq	Unknown	FP
Known	NNTP	98.90%	0.50%	0.20%	0.40%	0.00%
	POP3	52.30%	0.00%	0.40%	47.30%	0.00%
	SMTP	60.80%	0.30%	0.90%	38.00%	0.00%
	SSH	85.40%	0.00%	4.20%	10.40%	0.00%
	HTTPS	78.40%	0.00%	1.10%	20.50%	0.00%
	POP3S	100.00%	0.00%	0.00%	0.00%	0.00%
	HTTP	65.20%	1.00%	0.00%	33.80%	0.00%
	FTP	97.80%	1.00%	0.10%	1.10%	0.60%
	Edonkey	87.50%	0.90%	1.80%	9.80%	0.00%
	Kazaa	45.20%	0.00%	3.20%	51.60%	0.70%
	Overall	67.00%	1.00%	0.20%	31.80%	X
New	Bittorrent	X	32.90%	0.30%	66.80%	X
	IMAP	X	8.60%	57.80%	33.60%	X
	Gnutella	X	100.00%	0.00%	0.00%	X
	IRC	X	10.00%	0.00%	90%	X
	LDAP	X	8.80%	0.00%	91.20%	X
	MSN	X	38.50%	0.00%	61.50%	X
	Mysql	X	0.00%	0.00%	100.00%	X