Comparison of Internet Traffic Classification Tools

IMRG Workshop on Application Classification and Identification
October 3, 2007
BBN Technologies

Hyunchul Kim
CAIDA, UC San Diego

Joint work with
Marina Fomenkov, kc claffy, Nevil Brownlee (CAIDA, UC San Diego)
Dhiman Barman, Michalis Faloutsos (UC Riverside)

Outline

- We evaluated the performance of
  - CoralReef [CoralReef 07]
  - BLINC [Karagiannis 05]
  - Six machine learning algorithms [WEKA 07]

- Data used : 7 payload traces
  - Three backbone and four edge traces
  - From Japan, Korea, Trans-pacific, and US

- Performance metrics
  - Per-whole trace : accuracy
  - Per-application : precision, recall, and F-measure
  - Running time

Datasets

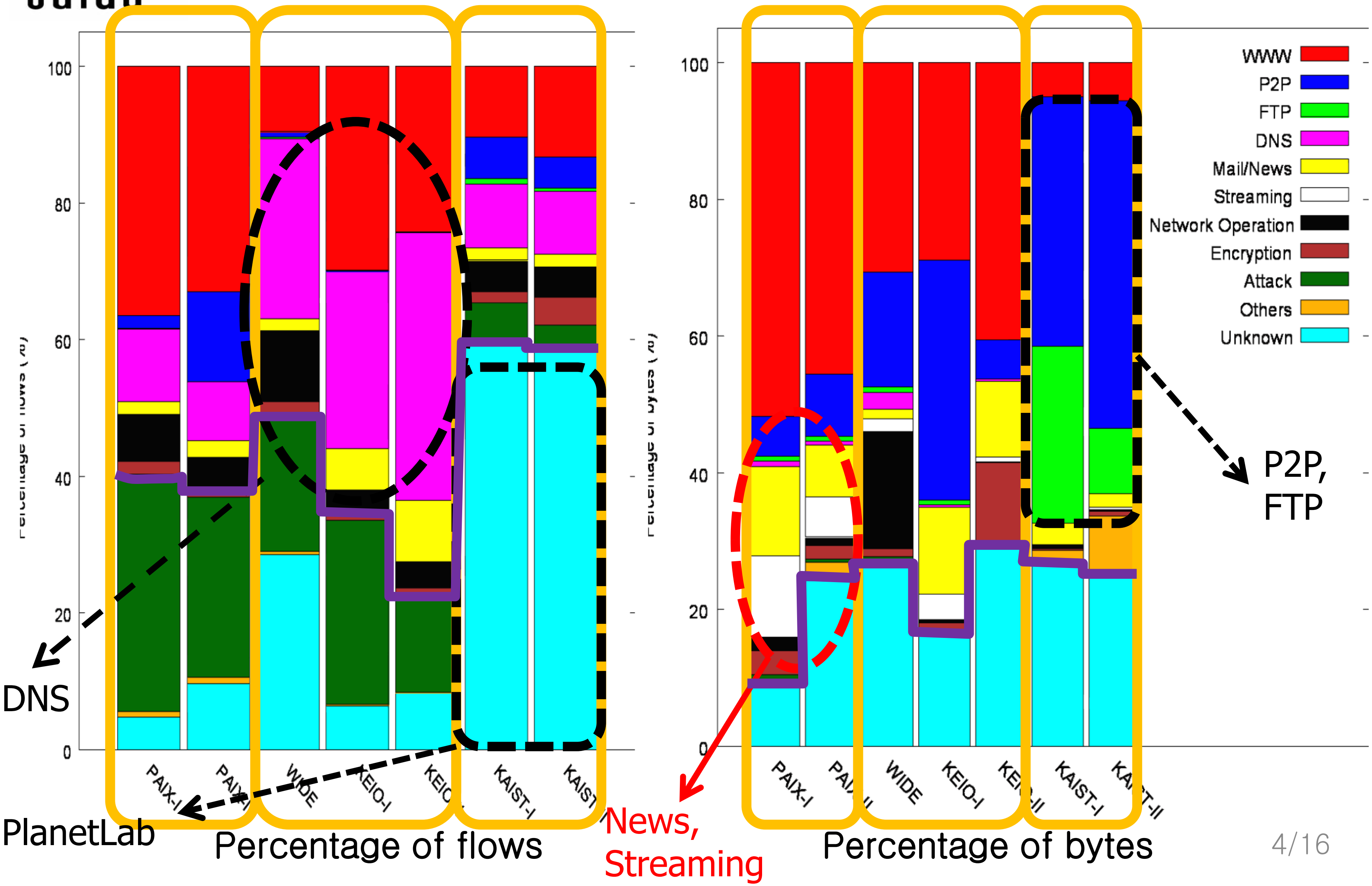
Trace (Country)	Link type	Date (Local time)	Start time & duration (Local time)	Average Utilization (Mbps)	Payload bytes per each packet
PAIX-I (US)	OC48 Backbone, uni-directional	2004.2.25 (Wed)	11:00, 2h	104	Max 16
PAIX-II (US)	OC48 Backbone	2004.4.21 (Wed)	19:59, 2h 2m	997	Max 16
WIDE (US-JP)	100 ME Backbone	2006.3.3 (Fri)	22:45, 55m	35	Max 40
KEIO-I (JP)	1 GE Edge	2006.8.8 (Tue)	19:43, 30m	75	Max 40
KEIO-II (JP)	1GE Edge	2006.8.10 (Thu)	01:18, 30m	75	Max 40
KAIST-I (KR)	1GE Edge	2006.9.10 (Sun)	02:52, 48h 12m	24	Max 40
KAIST-II (KR)	1GE Edge	2006.9.14 (Thu)	16:37, 21h 16m	28	Max 40

Payload-based classification

- Classification unit
  - 5-tuple flow
    - <srcip, dstip, protocol, srcport, dstport>
    - With 64 seconds timeout
    - 5 minute interval

- Payload signatures of 33+ applications from
  - The BLINC work [Karagiannis 05]
  - Jeff Erman et al.'s work [Erman 06]
  - Korean P2P/File sharing applications [Won 06]
  - Manual payload inspection

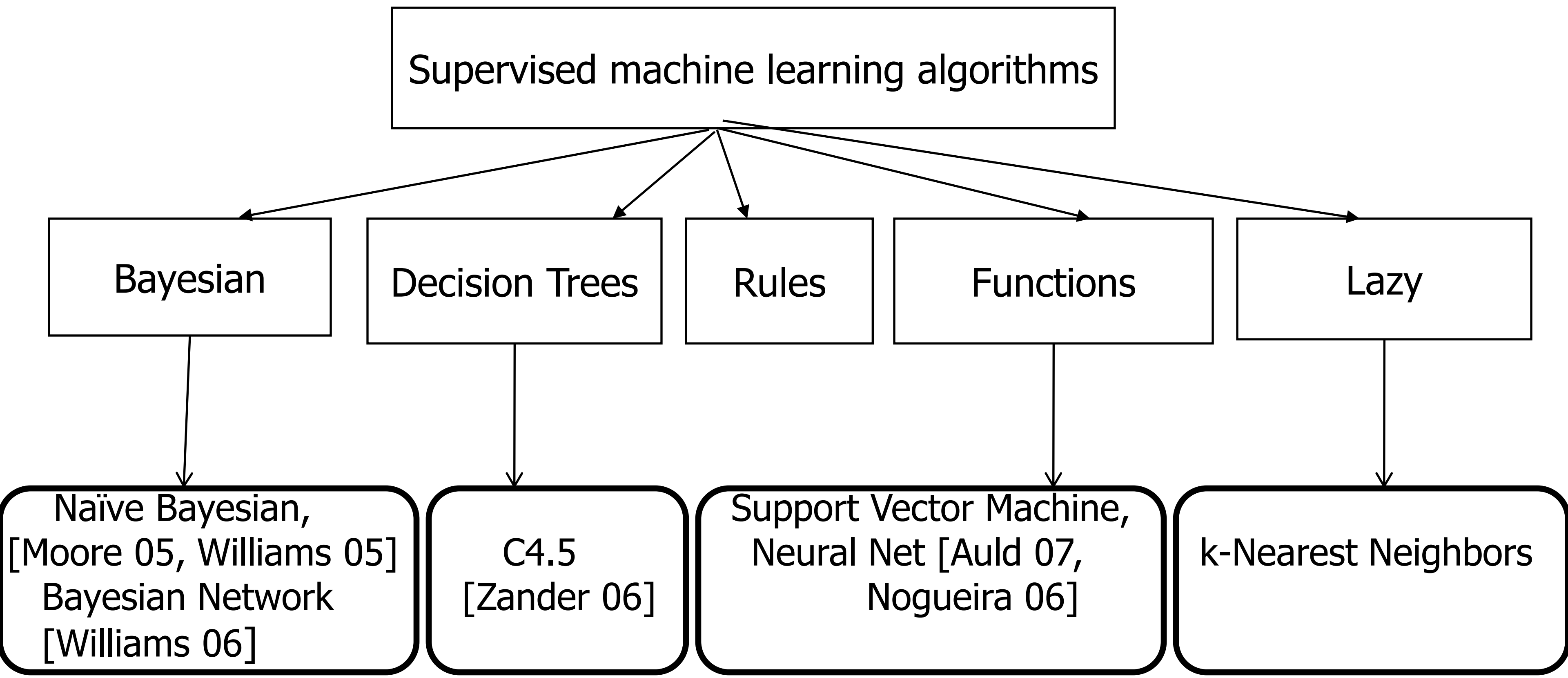
Application breakdown



Tools used

- CoralReef
  - Port number based classification
  - Version 3.8 (or later)
- BLINC
  - Host behavior-based classification
  - 28 configurable threshold parameters
- WEKA
  - A collection of machine learning algorithms
  - 6 most often used / well-known algorithms
  - Key attributes, training set size, and the best algo?

Machine learning algorithms

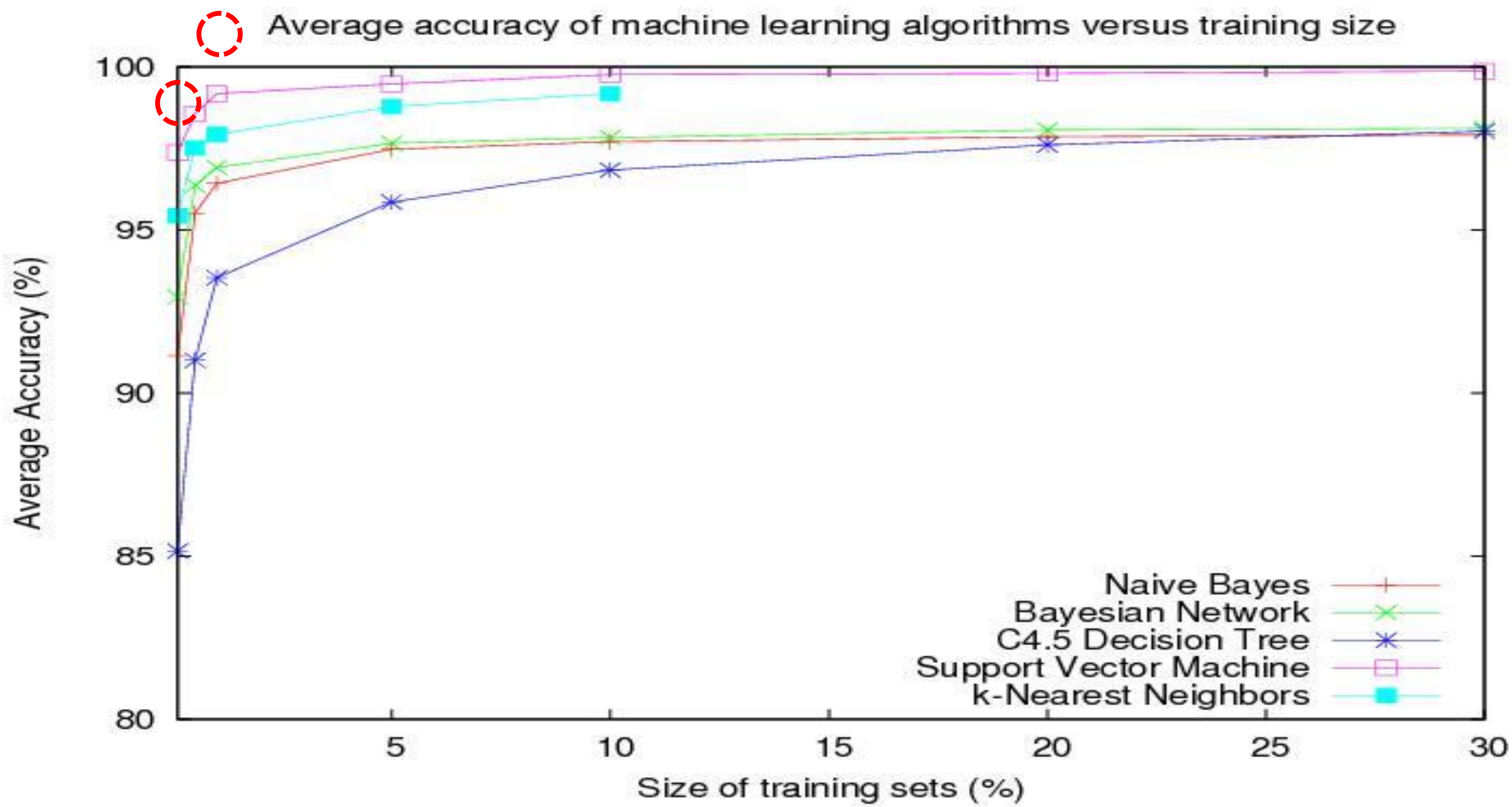


Key attributes by CFS

	Protocol	srcport	dstport	Payloaded or not	Min pkt size	TCP flags	Size of n-th pkt
Keio-I	O	O	O			PUSH	2, 8
Keio-II	O	O	O			PUSH	1, 4
WIDE	O	O	O	O		SYN, PUSH	4, 7
KAIST-I	O	O	O		O	SYN,RST, PUSH, ECN	3, 5
KAIST-II	O	O	O		O	SYN, PUSH	2, 3, 7
PAIX-I	O		O			SYN, ECN	2, 9
PAIX-II	O	O	O		O	SYN, CWR	1,4

* CFS : Correlation-based Feature Selection [Williams 06]

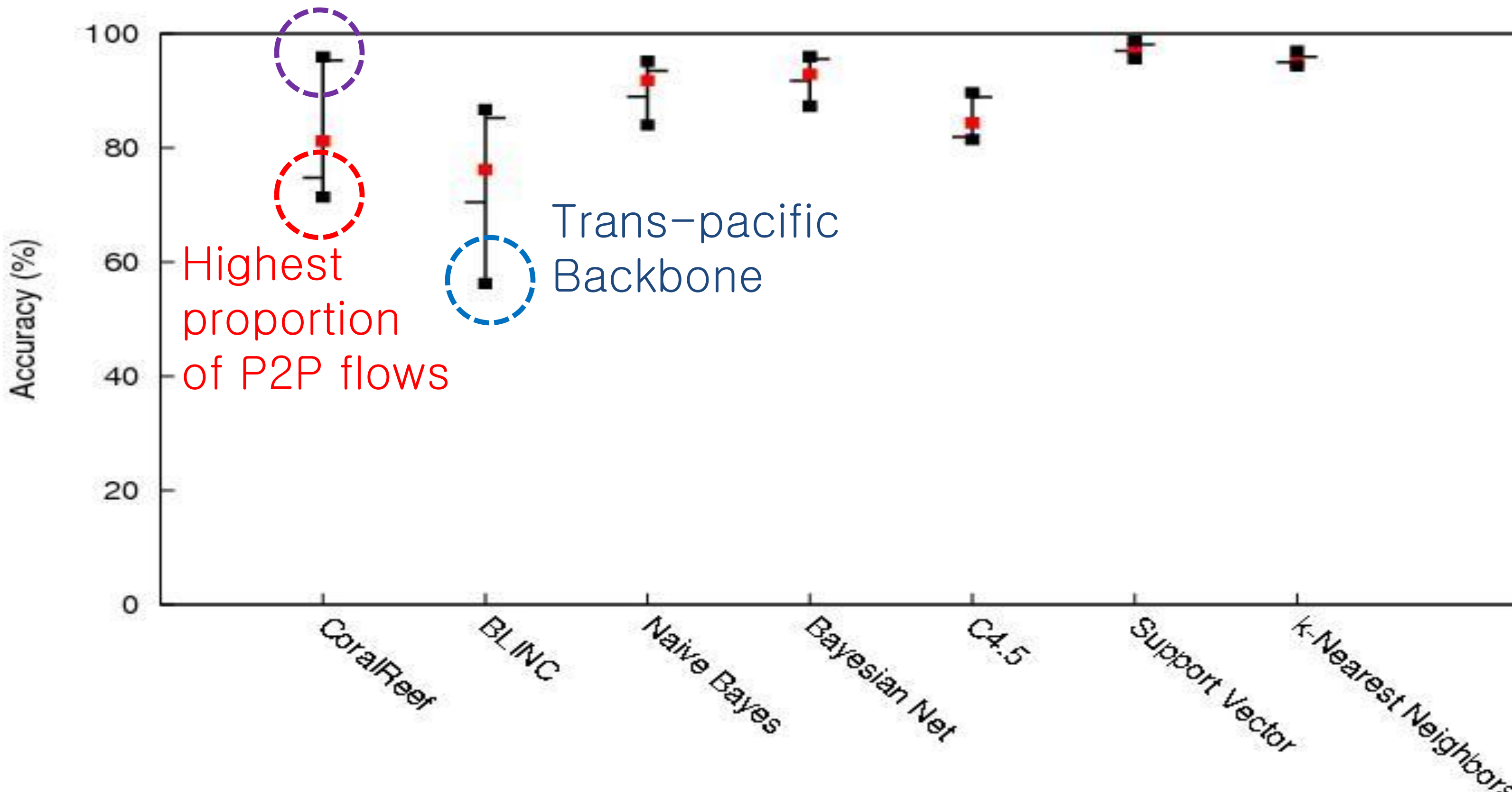
Training set size vs. accuracy



- Support Vector Machine achieves over 97.5 and 99%% of accuracy when only 0.1% and 1% of a trace is used to train it, respectively.

Accuracy

Lowest proportion of P2P flows



$$\text{Accuracy} = \frac{\textit{\# of correctly classified flows}}{\textit{total number of flows in a trace}}$$

* Only 0.1% of each trace is used to train machine learning algorithms

Per-application performance metrics

- Precision : "How precise is an application fingerprint?"

$$\frac{\textit{True Positives}}{\textit{True Positives} + \textit{False Positives}}$$

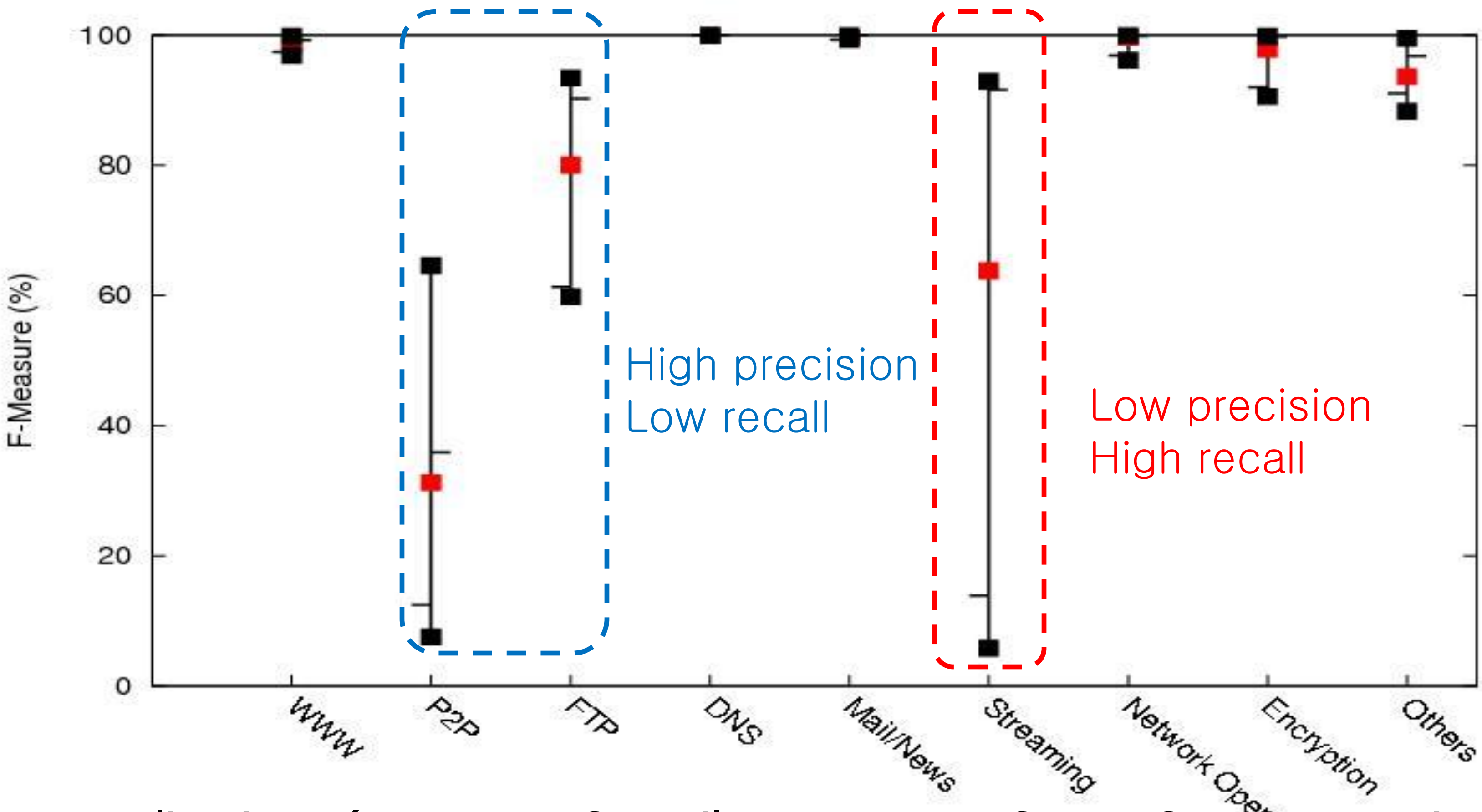
- Recall : "How complete is an application fingerprint?"

$$\frac{\textit{True Positives}}{\textit{True Positives} + \textit{False Negatives}}$$

- F-Measure : Combination of precision and recall

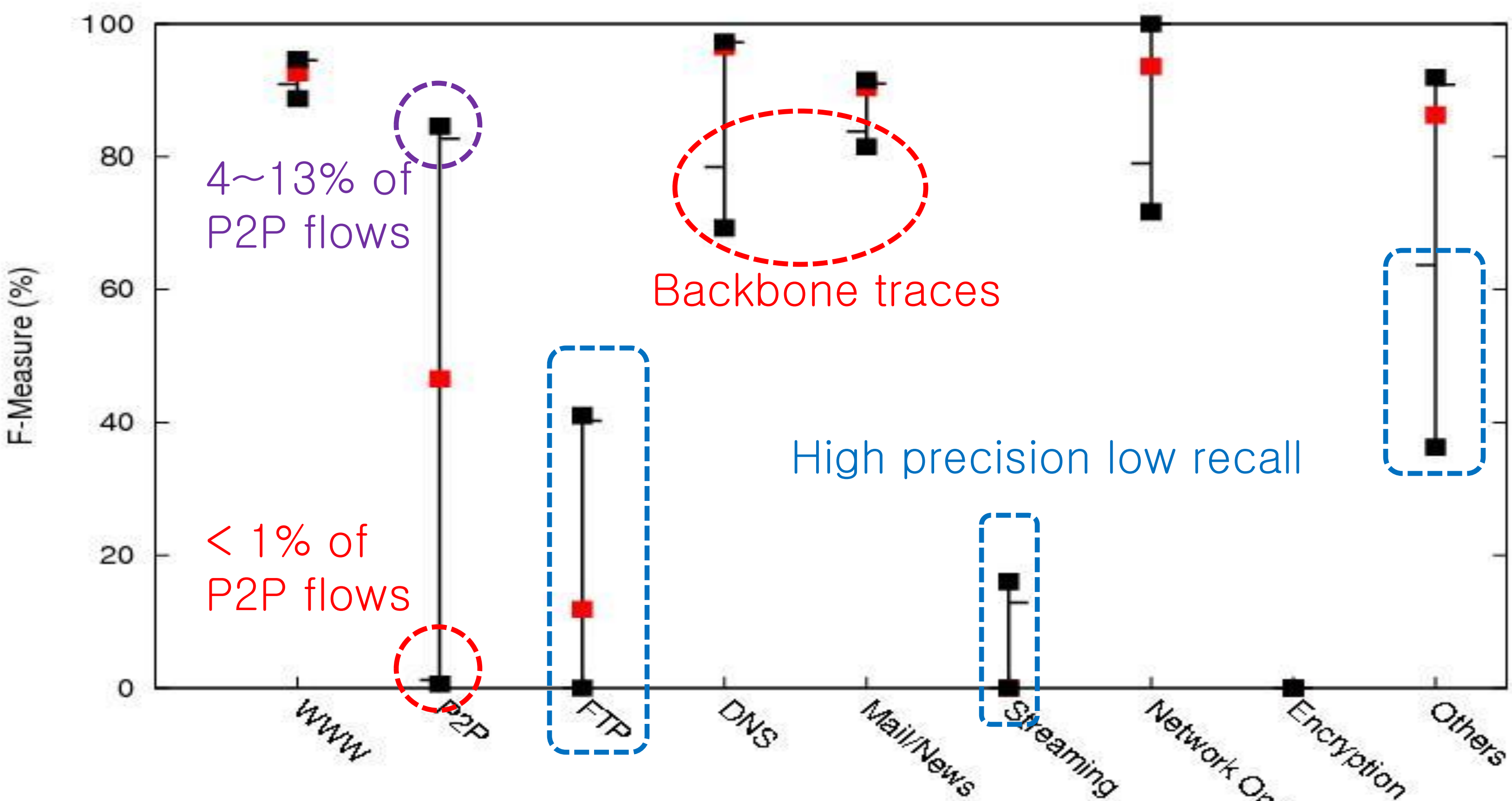
$$\frac{2 \times \textit{Precision} \times \textit{Recall}}{\textit{Precision} + \textit{Recall}}$$

F-Measure of CoralReef



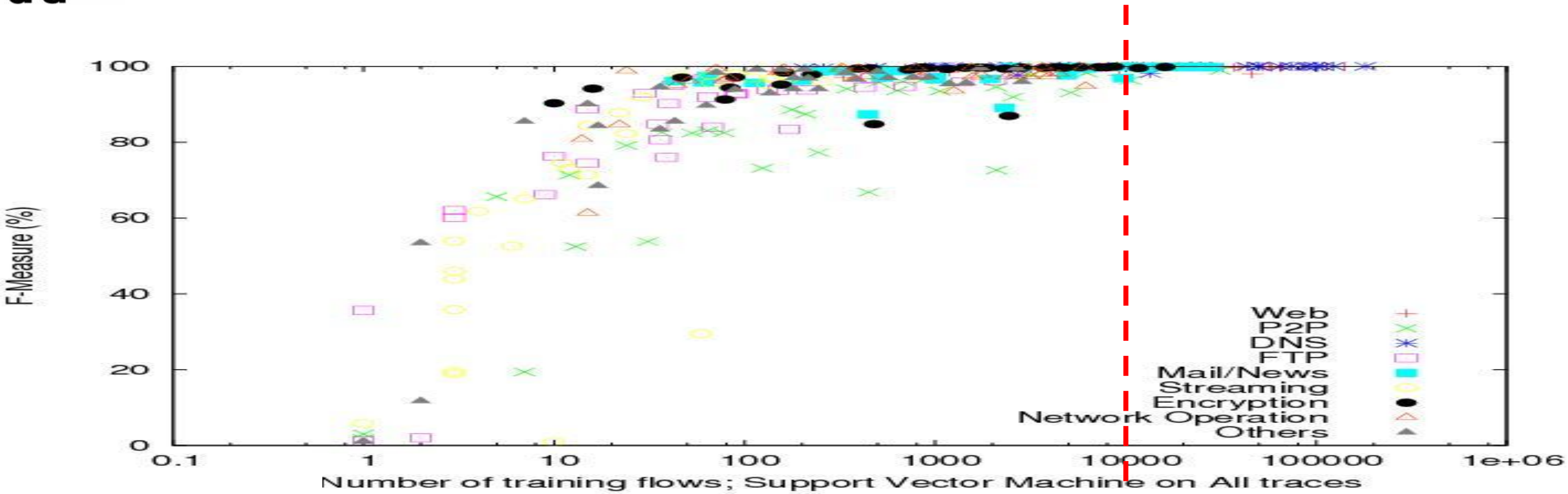
• Most applications (WWW, DNS, Mail, News, NTP, SNMP, Spam Assassin, SSL, Chat, Game, SSH, and Streaming) use their default ports in most cases.

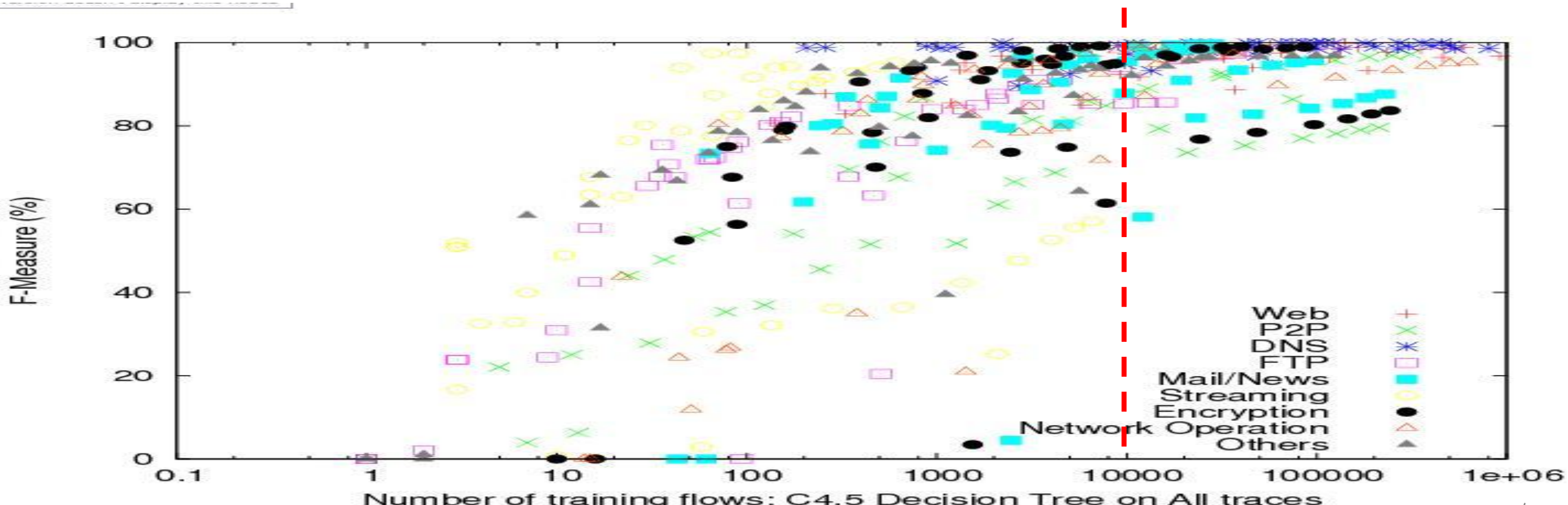
F-Measure of BLINC



- Incomplete fingerprints for the behavior of FTP, Streaming, and Game.
- Threshold-based mechanism mandates enough behavior information of hosts.
- Often misclassifies DNS and Mail flows on backbone traces.

F-Measure vs training set size (SVM vs C4.5)

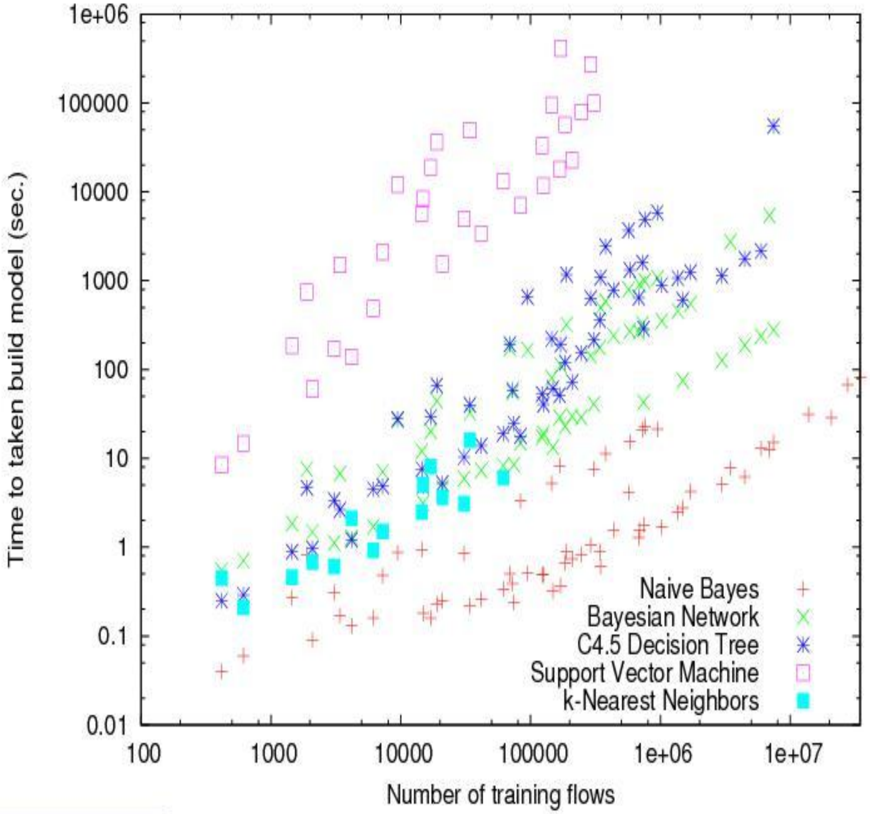




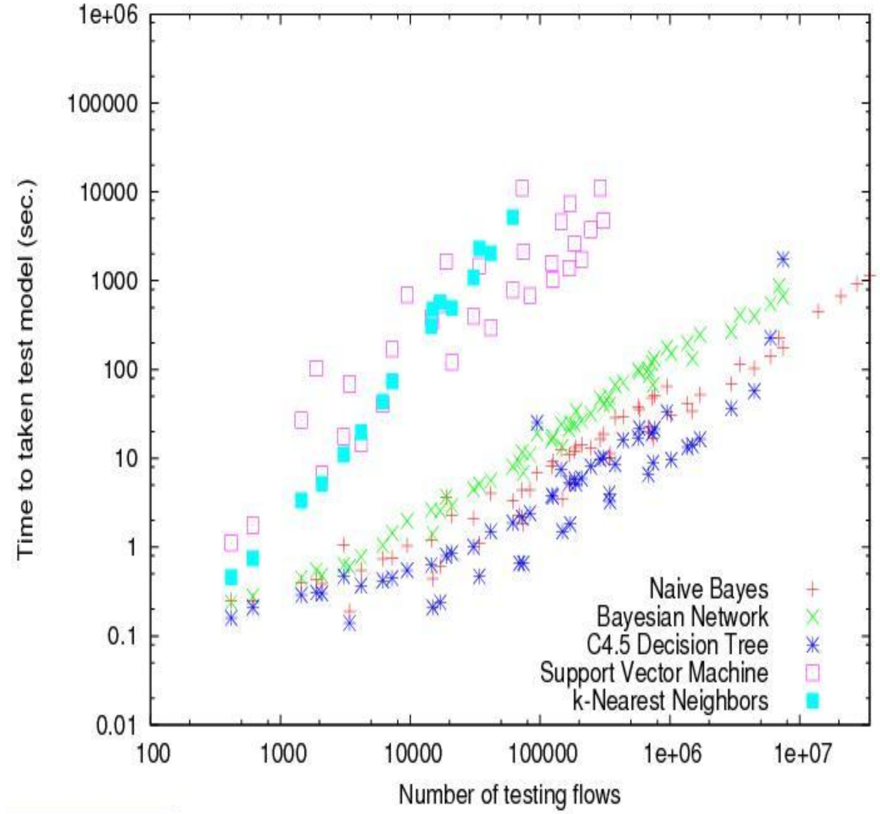
For all applications, Support Vector Machine requires the smallest # of training sets

Running time of machine learning algos

Training time



Testing time



"WEKA is very slow on large data sets." [Dimov 07]

Conclusions

1. Diverse traces & per-application performance eval.
   - → Understand contributions and limitations of each method
2. Port number
   - Still discerning attributes for many applications
3. BLINC
   - Highly depends on the link characteristics
   - Parameter tuning is too…
4. Support Vector Machine worked the best
   - Requires the smallest number of training set

Futurework

1. A robust traffic classifier
   - SVM trained with samples from our traces
   - Evaluating on 10 different payload traces
     - 7 existing + 3 new traces [DITL 07]
     - So far, >= 94~96% of accuracy on all of them
2. Longitudinal study of traffic classification
   - Internet2/NLANR trace archive, etc.
   - With all the tools?
3. Graph-similarity based traffic classification
   - Automatically tuning 28 parameters of BLINC
4. Internet host behavior analysis
   - For realistic Internet traffic modeling and regeneration

Acknowledgements

This research was supported in part by IT Scholarship program of Institute for Information Technology Advancement & Ministry of Information and Communication, South Korea.

- http://www.mic.go.kr

This research was supported in part by the National Science Foundation through TeraGrid resources provided by SDSC.

- http://www.nsf.gov

This research was supported in part by the San Diego Supercomputer Center under SDS104 and utilized the Datastar system.

- http://www.sdsc.edu

References

[Auld 07] Auld et al., "Bayesian Neural Networks For Internet Traffic Classification," *IEEE Transaction on Neural Networks*, 18(1):223-239, January 2007.

[Bennet 00] Bennet, "Support Vector Machines: Hype or Halleujah?," *ACM SIGKDD Explorations,* 2(2):1-13, October 2000.

[CoralReef 07] CoralReef. http://www.caida.org/tools/measurement/coralreef

[Erman 06] Erman et al., "Traffic Classification Using Clustering Algorithms," *ACM SIGCOMM Workshop on Mining Network Data (MineNet),* Pisa, Italy, September 2006.

[Dimov 07] Dimov, "Weka: Practical machine learning tools and techniques with Java implementations," *AI Tools Seminar,* University of Saarland, April 2007.

[DITL 07] Day in the Life of the Internet. http://www.caida.org/projects/ditl

[Karagiannis 05] Karagiannis et al., "BLINC: Multi-level Traffic Classification in the Dark," *ACM SIGCOMM 2005,* Philadelphia, PA, August 2005.

[Moore 05] Moore et al., "Internet Traffic Classification Using Bayesian Analysis Techniques," *SIGMETRICS 2005,* Karlsruhe, Germany, August 2003.

[Nogueira 06] Nogueira et al., "Detecting Internet Applications using Neural Networks," *IARIA Internet Conference on Networking and Services,* July 2006.

[WEKA 07] WEKA: Data Mining Software in Java. http://www.cs.waikato.ac.nz/ml/weka

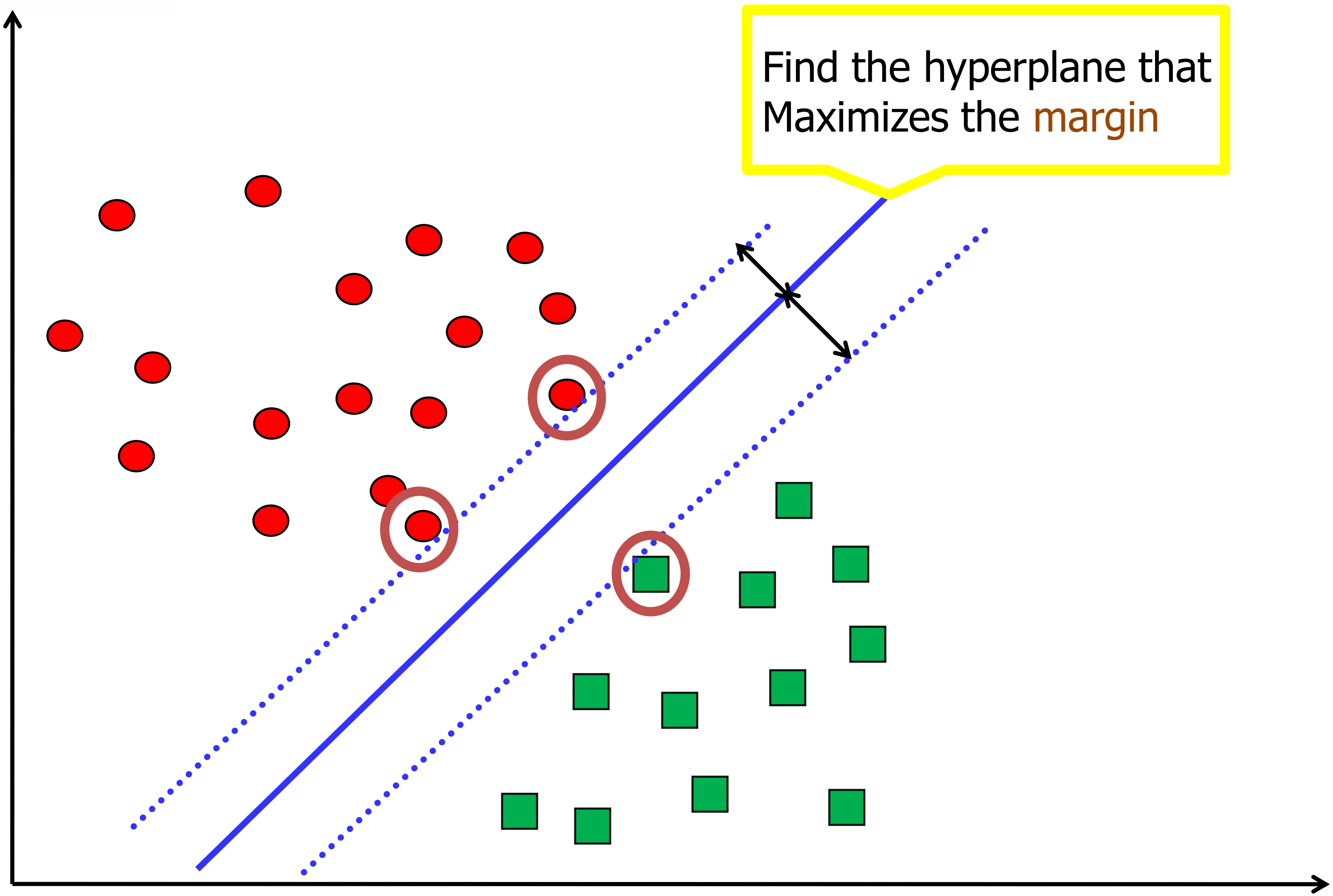
[Williams 06] Williams et al., "A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification," *ACM SIGCOMM Computer Communication Review,* 36(5):7-15, October 2006.

[Won 06] Won et al., "A Hybrid Approach for Accurate Application Traffic Identification," *IEEE/IFIP E2EMON*, April 2006.

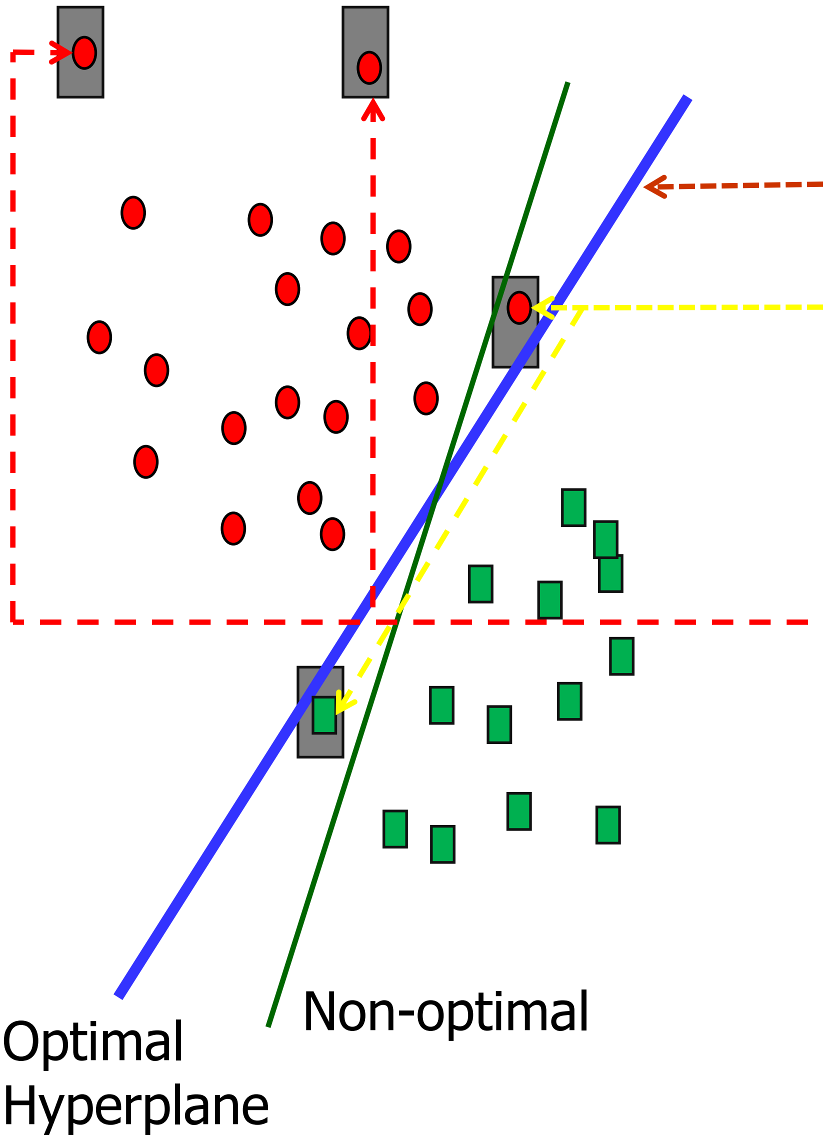
[Zander 06] Zander et al., "Internet Archeology: Estimating Individual Application Trends in Incomplete Historic Traffic Traces," CAIA Technical Report 060313A, March 2006.

Backup slides

SVM Design Objective [Bennet 00]



Why maximum margin Hyperplane? [Bennet 00]



Intuitively, this feels safest

A hyperplane is really simple

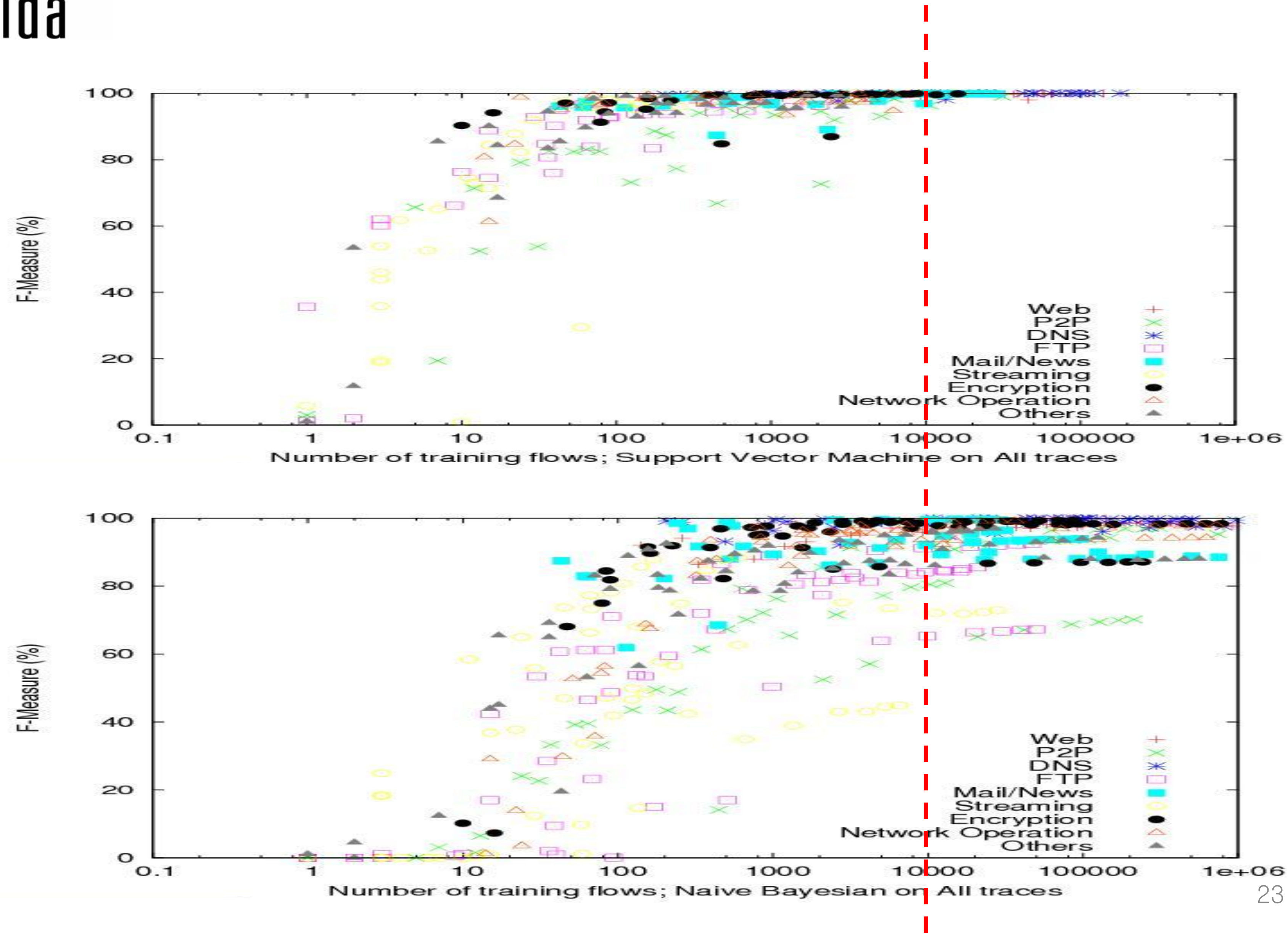
If we've made a small variation near the boundary this gives us least chance of causing a misclassification.

It is robust to outliers since non-support vectors do not affect the solution at all.

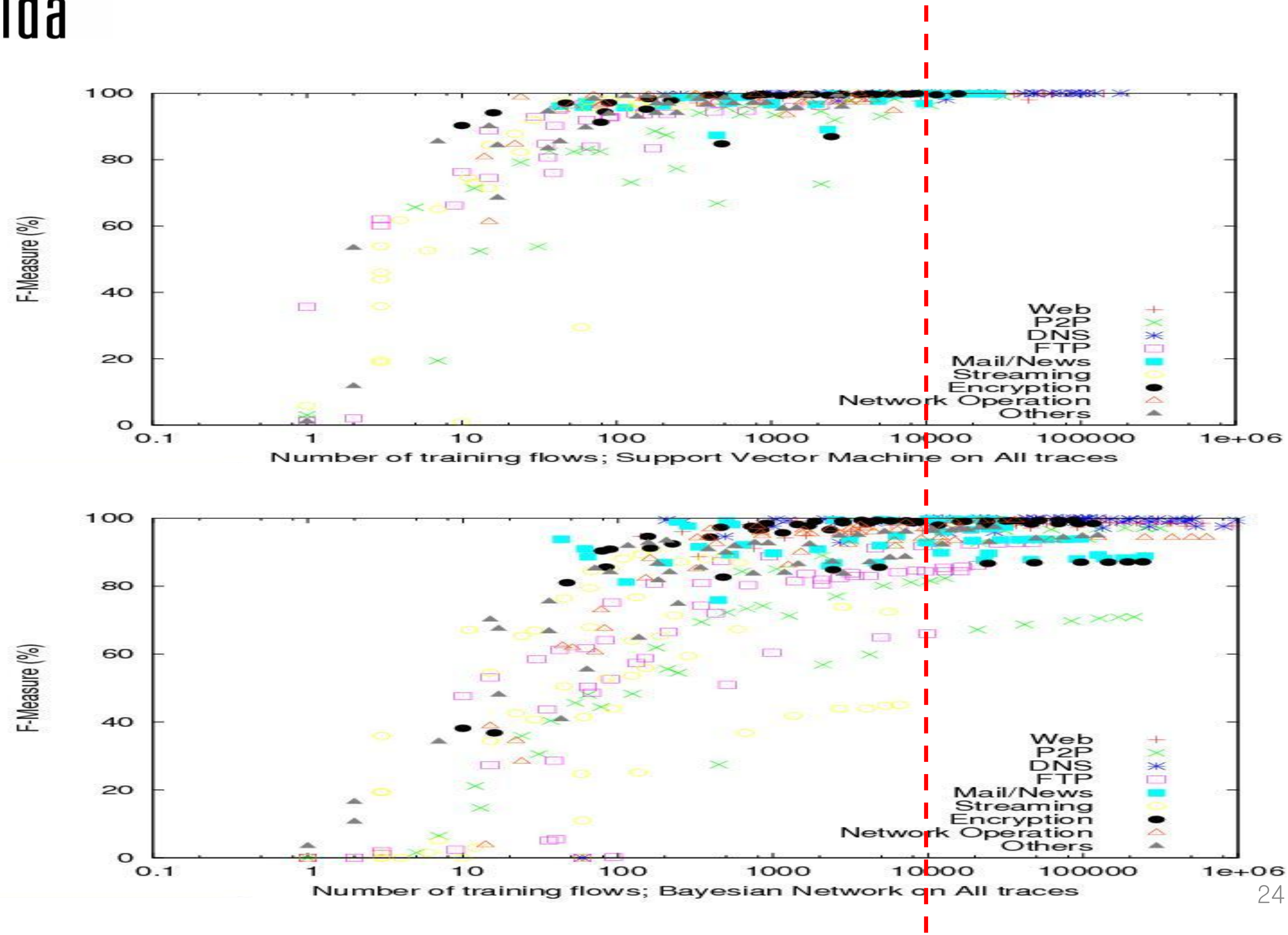
Empirically it works very well

There is Structural Risk Minimization theory (using VC D.) that gives the upper bound of generalization error.

F-Measure vs training set size (SVM vs Naïve Bayes)



F-Measure vs training set size (SVM vs Bayesian Net.)



F-Measure vs training set size (SVM vs k-Nearest.)