REDJACK

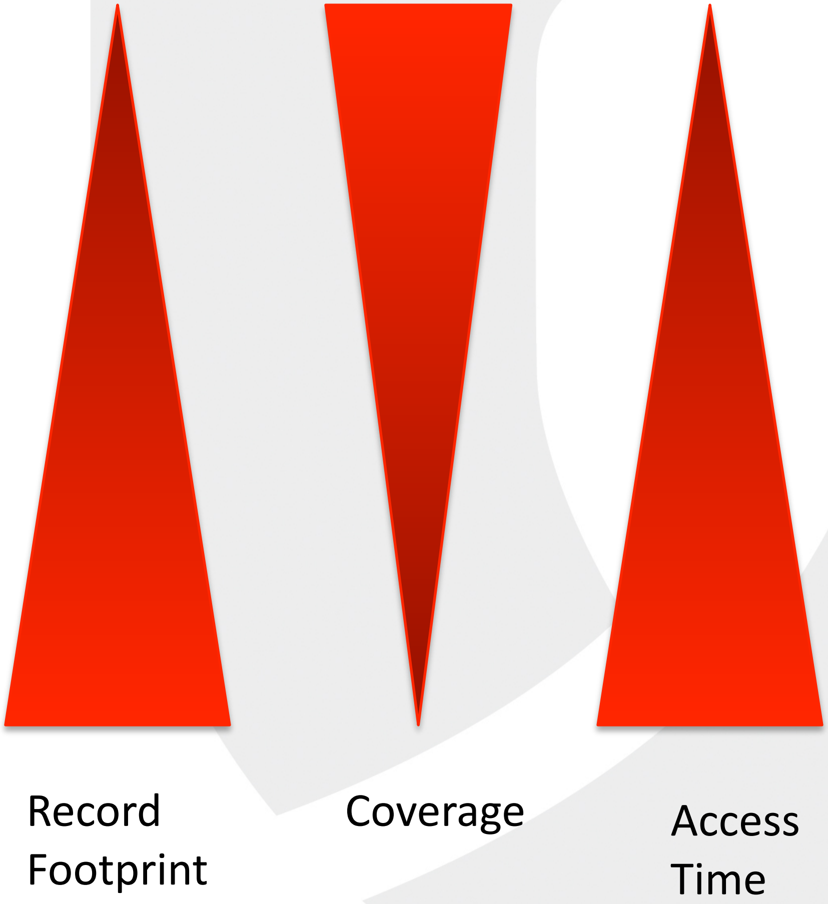
Outline

- Core Concept and Goal
  - Expressiveness *vs* Representation
- Towards a Behavioral Dictionary
- Example Behavior: Fumbling
  - What is Fumbling?
  - Why Fumbling?
  - Who Fumbles?
- Current Study: ID Crawlers Via Fumbling

CORE CONCEPTS

Expressiveness Vs. Representation

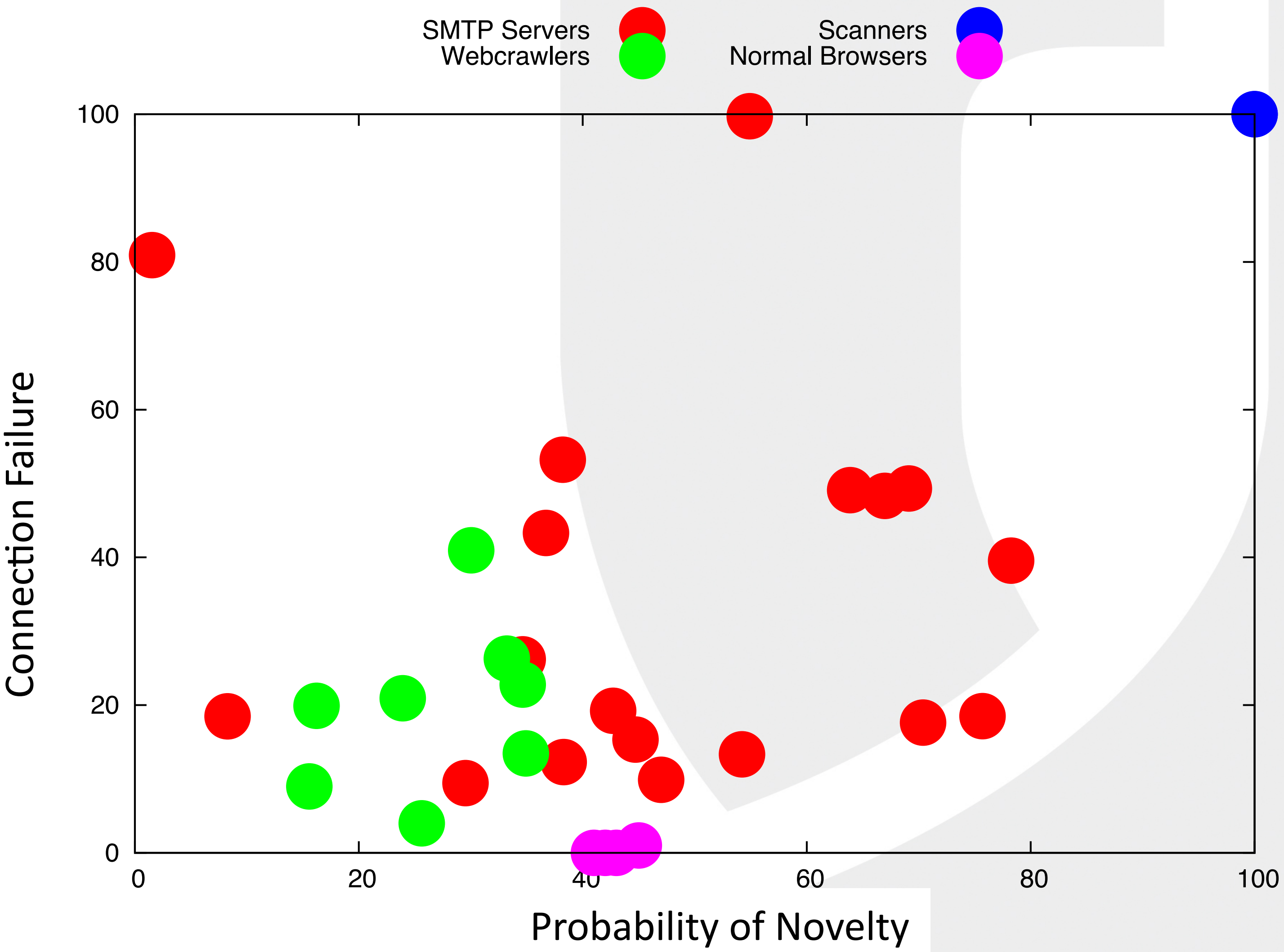
- “If I just had **TCPDUMP OF EVERYTHING** all my problems would be solved”
- Most traffic, on a flow-by-flow basis is either garbage or uninteresting



Goal

- Develop narratives which describe activity between hosts in a more abstract fashion
  - “This is fumbly”
  - “This is chatty”
- Ideally, these attributes will be
  - Intuitive (an analyst can grasp them by looking at a log)
  - Rigorous (derived from some model of behavior)
- Partly identification applications by behavior

A Crude Picture



REDJACK

Attributes For Narratives…

- This is clustering, just on different axes
- Possible attributes:
  - Probability of connection failure
  - *Locality*
  - Probability of file transfer
  - Packet size

CASE STUDY: FUMBLING

What is Fumbling?

- Intuitively, fumbling is a *consistent* failure to connect with a host
  - Previously used to identify BitTorrent [Collins06,Bartlett07]
- Challenge: differentiating fumbling
  - From scanning, where clients probe 'a lot' [Jung04]
  - From normal surfing, where clients get bored and move on

What Fumbles?

- Routed/automated lookup
  - SMTP
  - P2P
  - NNTP
- Search bots

- Scanners don’t fumble – they seek out everything
- Users don’t fumble – they lose patience

Why Care About Fumbling?

- Scanning false positives
  - Uncleanliness Data – don't mark yahoo unclean
  - Differentiate scanners in a naturally noisy set -- SMTP
- Identify applications that require blind lookup
  - Internal p2p applications (unknown ports)
  - Google doesn't publish crawler IP addresses

REDJACK

CHARACTERIZING FUMBLING

Source Data

- Task: differentiate crawlers by quantifying fumbling
- 4 days of crawling data
- Crawlers identified by IP space
  - Cuil: Google "competitor", embarrassing launch last year
  - Yeti: Naver.com (Korean search engine)
  - "Twiceler": Some searchbots use twiceler as an ID, refers to twiceler from domains cuil says are not its domains
  - Voila: Voila.fr search engine (French)

REDJACK

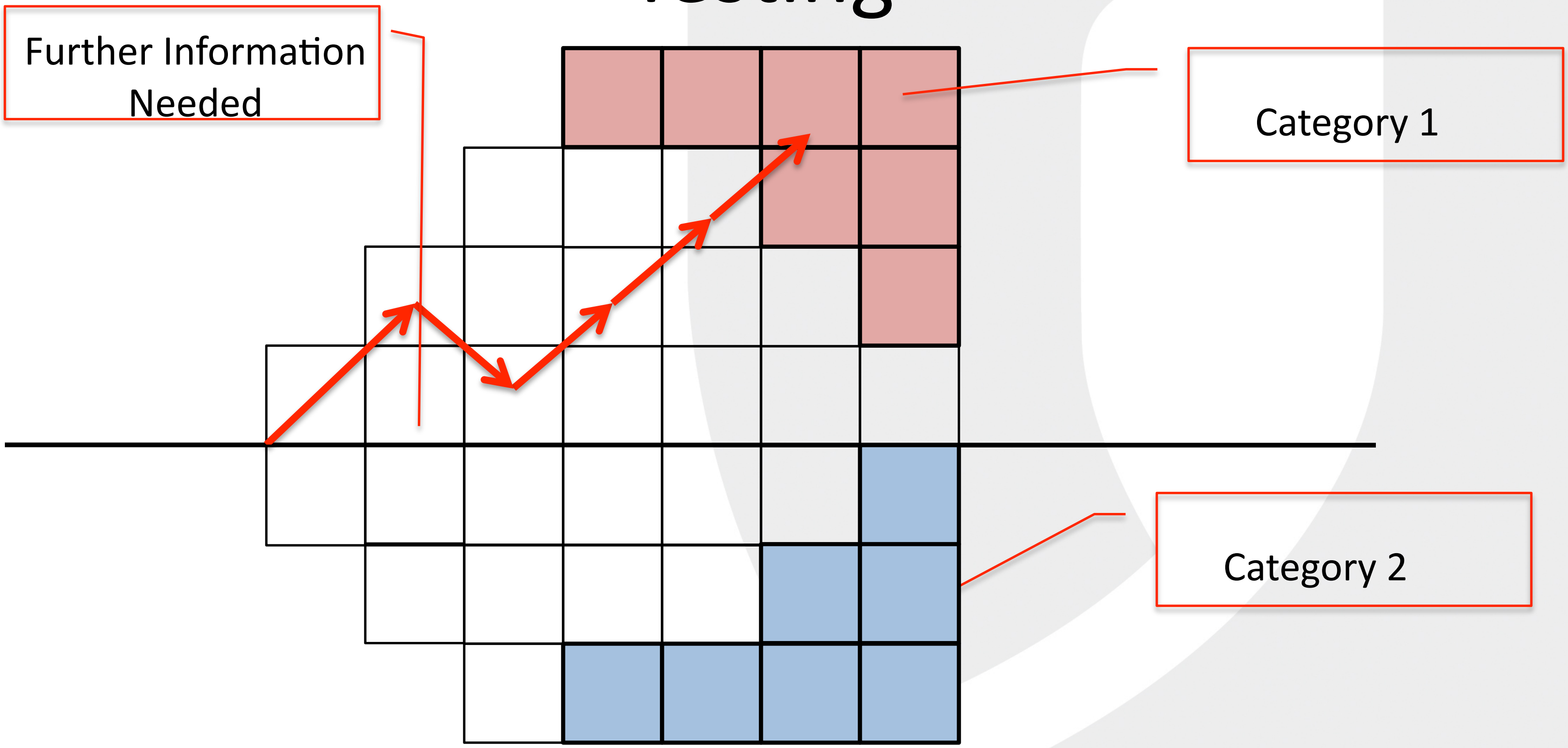
Basic Numbers

Engine	ID#	Flows	IPs	Failed Flow	Failed IP
Cuil	1	3760	189	504	45
	2	4945	170	195	42
	3	3128	204	1033	43
Yeti	4	2635	247	84	28
“Twiceler”	5	5338	185	829	51
Voila	6	12808	680	2745	75
	7	12506	679	2306	73
“Twiceler”	8	2252	172	101	45

Consecutive Failure Rate

- Number of times that a failure occurs repeatedly
- Used in darkspace analysis – scans are marked as such when > 3-5 consecutive failures [Jung, 2004]
- Fumblers are different because they have a nontrivial success rate

Visualizing Sequential Hypothesis Testing

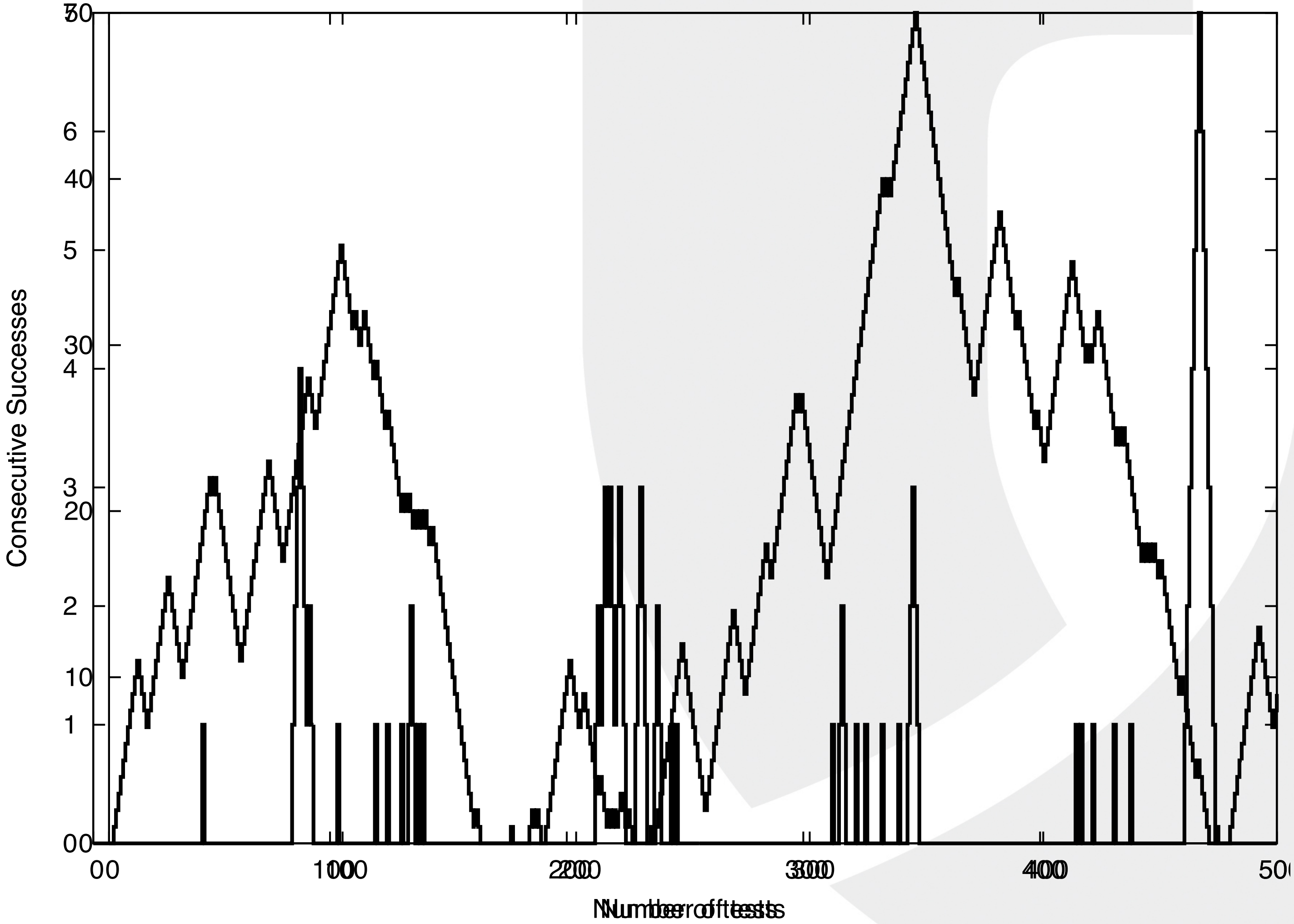


Further Information Needed

Category 1

Category 2

Some Failure Plots

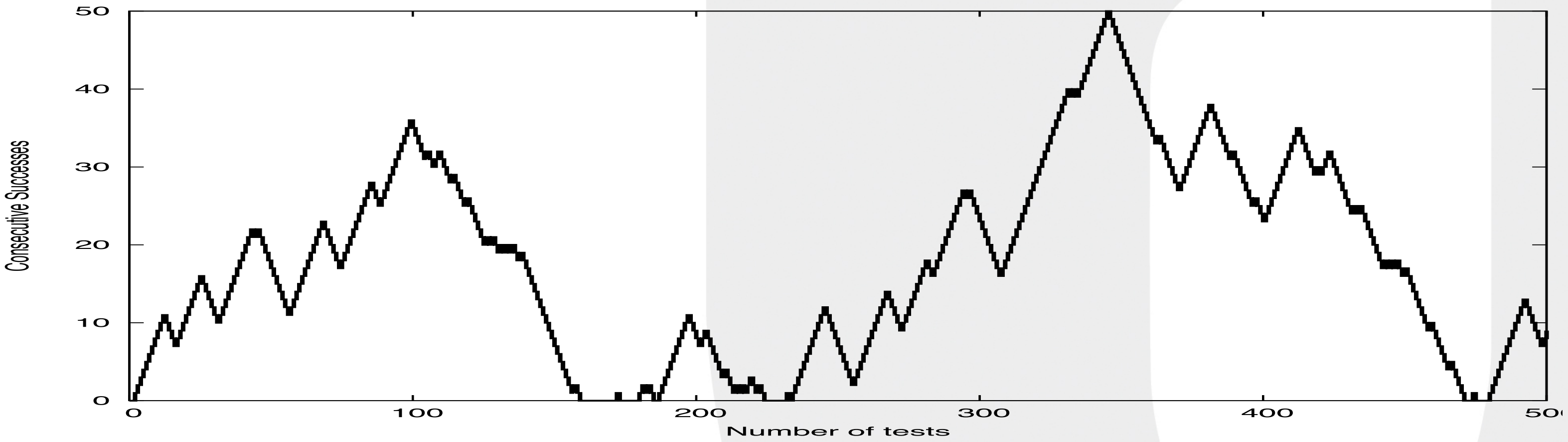


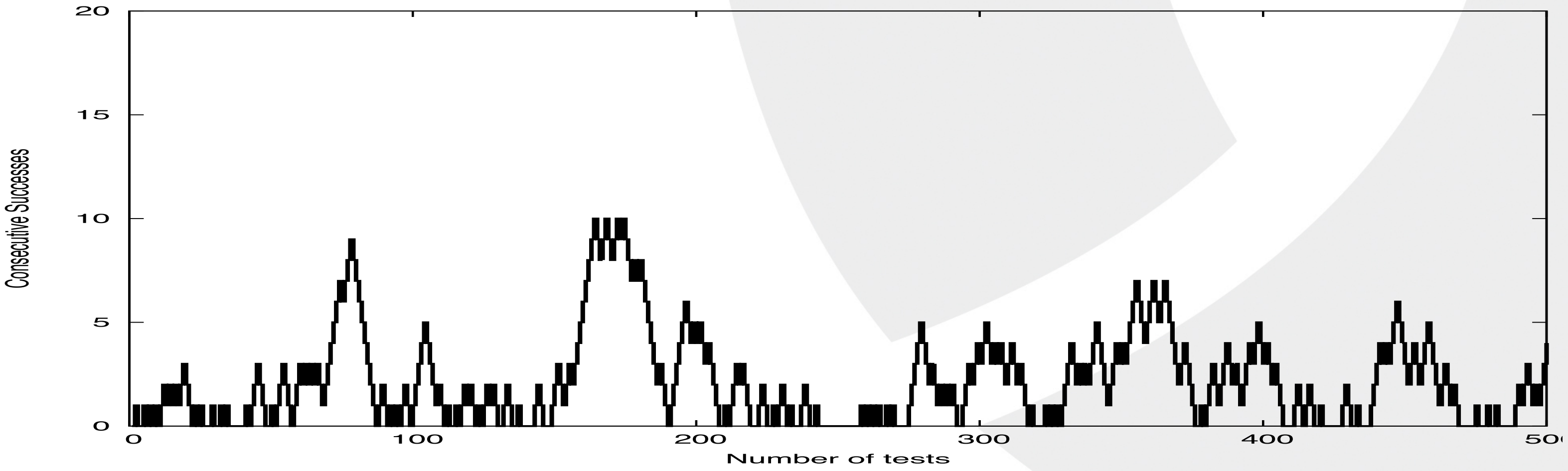
Why The Mountains?

- In the long run, high success rate:
  - 0.5% failure to 70% in the worst case
  - Vs. 99.95% failure rate for scanners
- However, failures are common mode
  - IP address X is down
  - IP address X is hit repeatedly

Engine	ID#	FPR (4 failures)
Cuil	1	9.10%
	2	1.50%
	3	34.4%
Yeti	4	10.3%
"Twiceler"	5	17.9%
Voila	6	13.9%
	7	1.00%
"Twiceler"	8	1.00%

Permuting Addresses





Results of Permutation

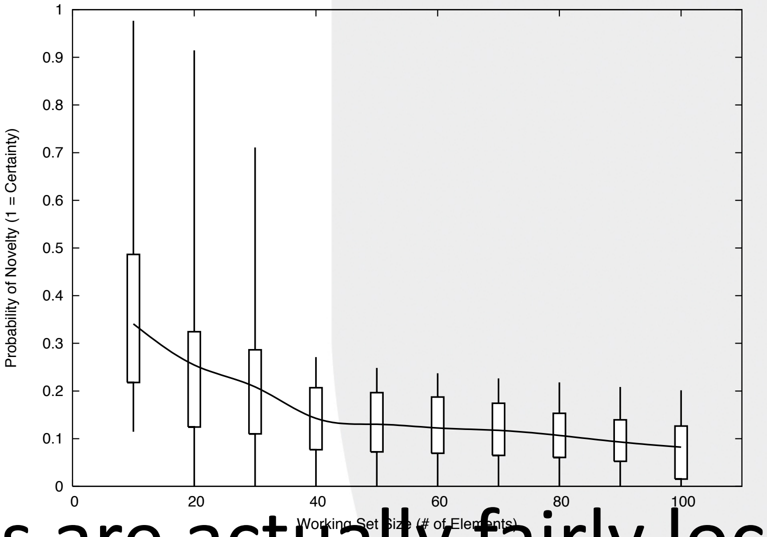
- Drops expected values down
- “realtime” detection is no longer valid
- Fumbling requires both a presence and absence…

Engine	ID #	Norm FPR	Random FPR
Cuil	1	9.10%	0.00%
	2	1.50%	0.00%
	3	34.4%	15.5%
Yeti	4	10.3%	0.00%
“Twiceler”	5	17.9%	0.00%
Voila	6	13.9%	0.00%
	7	1.00%	0.00%
“Twiceler”	8	1.00%	0.00%

Locality

- Propensity of users to sit around a set of common hosts [McHugh03]
- Modeled using a working set:
  - LRU stack, fixed size
  - Locality is then the probability, when an address is presented, of not replacing an address in the working set

Searchbots Are Local

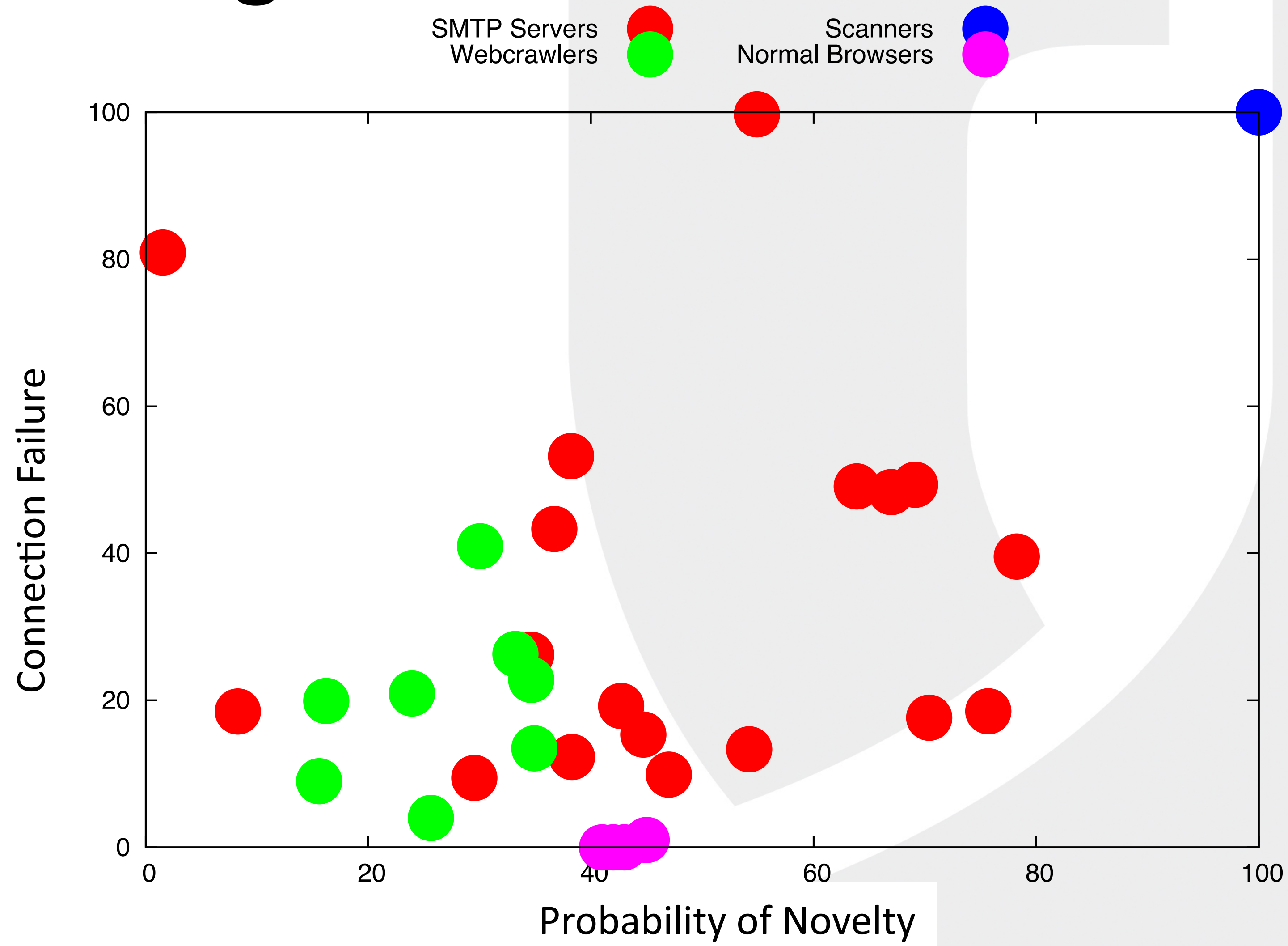


- Searchbots are actually fairly local
  – Slightly *more* localized than humans, it turns out
  – CDNs? Single-page sites?
- *Much* more local than scanners

Start Classifying

Local?	Connects?	
	Yes	No
Yes	Surfer	Searchbot
No	Hitlist Scanner?	Scanner

Leading Us Back To This Picture…



Conclusions

- Combining locality with detection failure may provide an indicator of fumbling
  - Have to develop a suitable *n* (working set size)
  - *N* also changes over time
- A false positive is an indicator your IDS isn't done yet
  - We can differentiate searchbots from scanners with more information, but it may cost us 'realtime'
  - Whatever 'realtime scan detection' is worth…