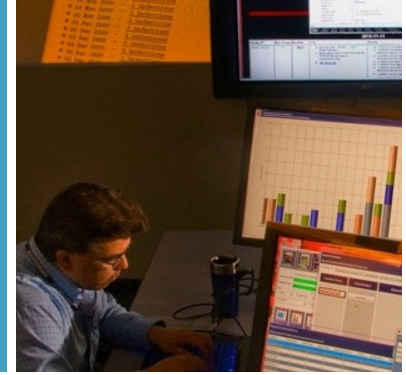
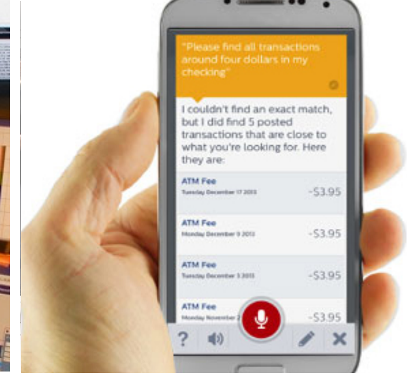
SRI International





Security, Reliability, and Accountability of Large Infrastructure Systems

Gabriela Ciocarlie, PhD
Program Manager
Cyber Analytics Group
SRI NYC

Security and Reliability: A Broad Range of Environments

- Targeted attacks on **industrial control systems (ICS)** are growing in frequency and severity
  - 7,200 Internet-facing control system devices in U.S.
- **Network cells** can suffer degraded performance or outage without raising any explicit alarms
  - explosion of mobile data traffic from use of tablets, smartphones, and netbooks for day-to-day tasks
- Critical information is migrating into the **cloud**
  - SLAs include no clauses with procedures to follow in case of forensic investigation

Data Analytics for Secure and Reliable Systems

- Modern technologies generate a wealth of data
  - Part of their functionality
  - Byproduct of their operation

- Analyze the entirety of the data that a system produces
  - Audit and monitor
  - Keep system within intended behavioral boundaries

Threat/Failure Detection Analysis

- Traditionally relies on **signature-based** detectors
  - blind to zero-day attacks
  - do not detect new types of failures

- Alternative: **anomaly-based** detection (AD) sensors
  - model **normal behavior** of systems
  - natively well-suited for detecting zero-day attacks and new types of failures
  - becoming a necessity, rather than an option

BUT….

Motivation – AD Sensors

- Major hurdles in the deployment, operation, and maintenance of AD systems:
  - Real training data is polluted
    - Manual labeling is difficult
  - Must adapt to the system under protection
    - Calibration by a human expert
  - False positives
    - Manual inspection is needed
  - Protected system evolves over time
    - Operator must keep AD sensor up-to-date

Outline

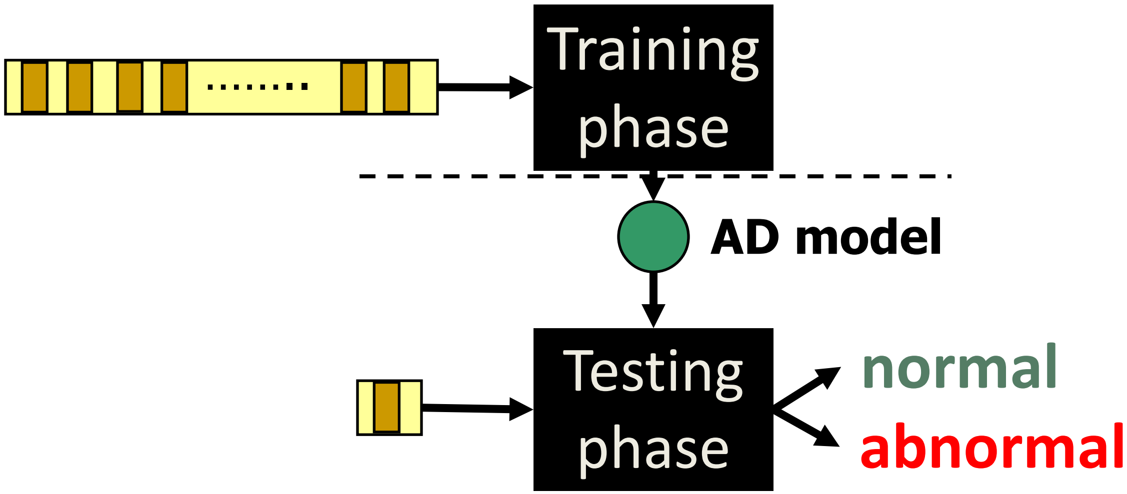
- Hands-free accurate anomaly detection

- Communication pattern monitoring for industrial control systems

- Anomaly detection in operational cellular networks

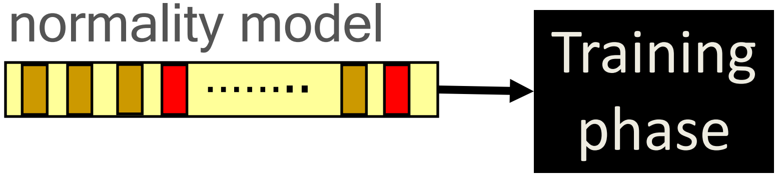
Anomaly Detection

- Trained on a stream of continuous data
- Creates a self-contained AD model
- Classifies a new data point as either normal or abnormal



Anomaly Detection

- Fundamental problem: quality of models
- Attacks and abnormalities
  - pollute training data
  - poison normality model

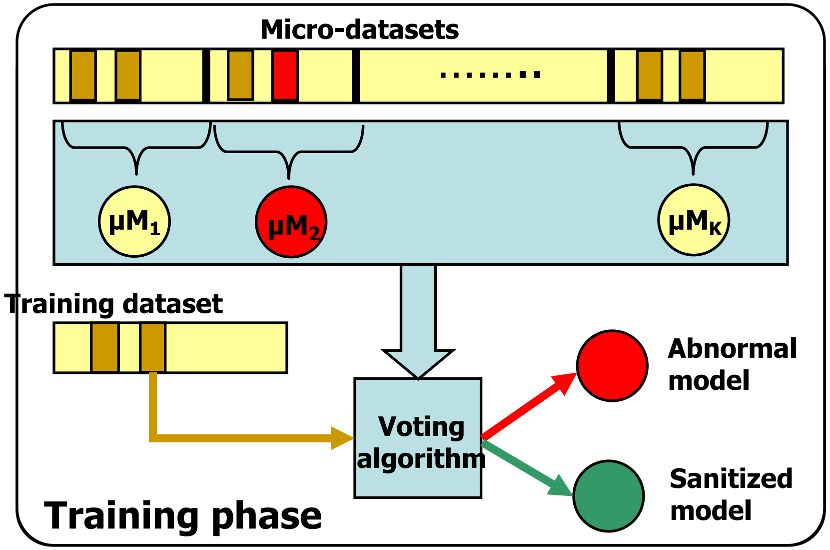


- Goal: remove them from training dataset

- Related ML algorithms: ensemble methods [Dietterich00], MetaCost [Domingos99], meta-learning [Stolfo00]

Training Strategies: Sanitization

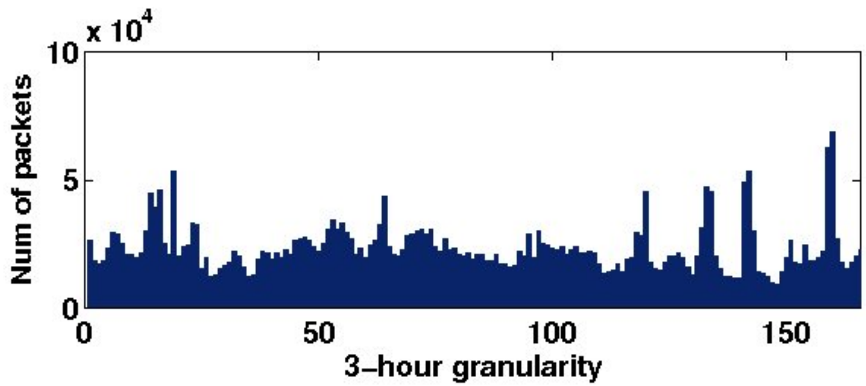
- Divide training data into multiple micro-datasets with the same time granularity
- Build **micro-models** for each micro-dataset
- Test all models against a smaller dataset
  - Hypothesis: attacks and non-regular data cause localized "pollution“
- Build **sanitized** and **abnormal** models
  - use a voting algorithm
  - V = *voting threshold*

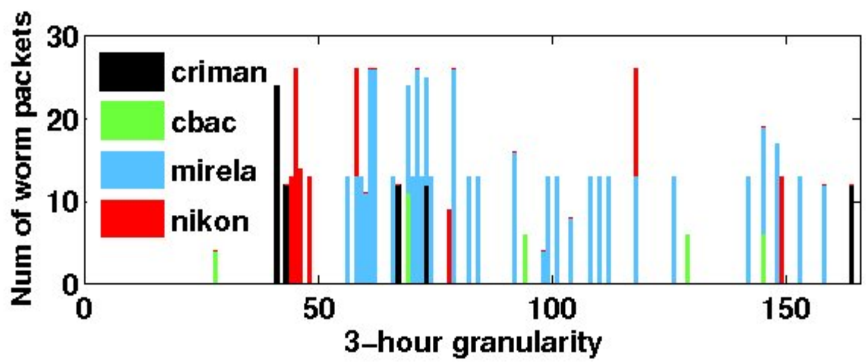


Published at HotDep07,S&P08: Cretu et al.

Evaluation Dataset

- 300/100/100 hours of real network traffic
- Three different http traces
- Implementation using two content-based AD:
  - Anagram [Wang06] - n-gram analysis
  - Payl [Wang05] - byte frequency distributions

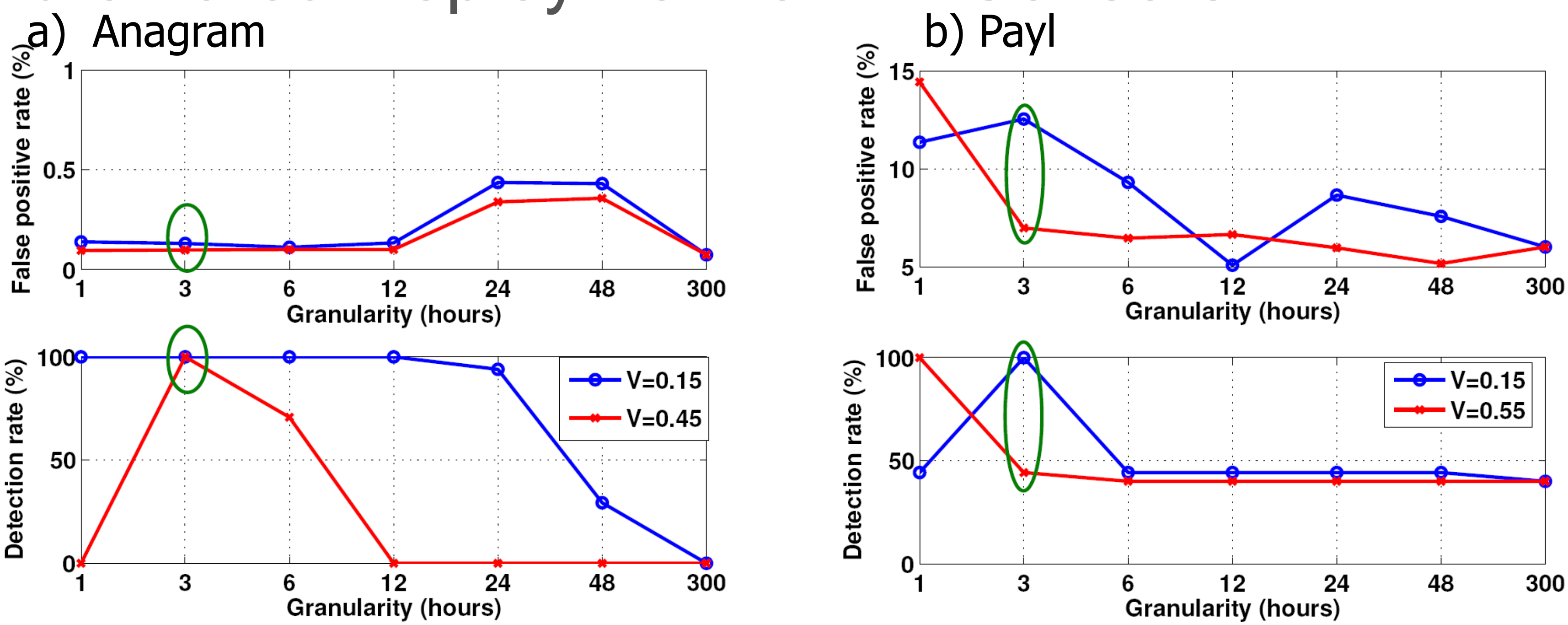




AD Sensors Comparison

Sensor	www1		www		lists	
	FP (%)	TP (%)	FP (%)	TP (%)	FP (%)	TP (%)
Anagram	0.07	0	0.01	0	0.04	0
Anagram with Snort	0.04	20.20	0.29	17.14	0.05	18.51
Anagram with sanitization	**0.10**	**100**	**0.34**	**100**	**0.10**	**100**
Payl	0.84	0	6.02	40	64.14	64.19
Payl with sanitization	**6.64**	**76.76**	**10.43**	**61**	**2.40**	**86.54**

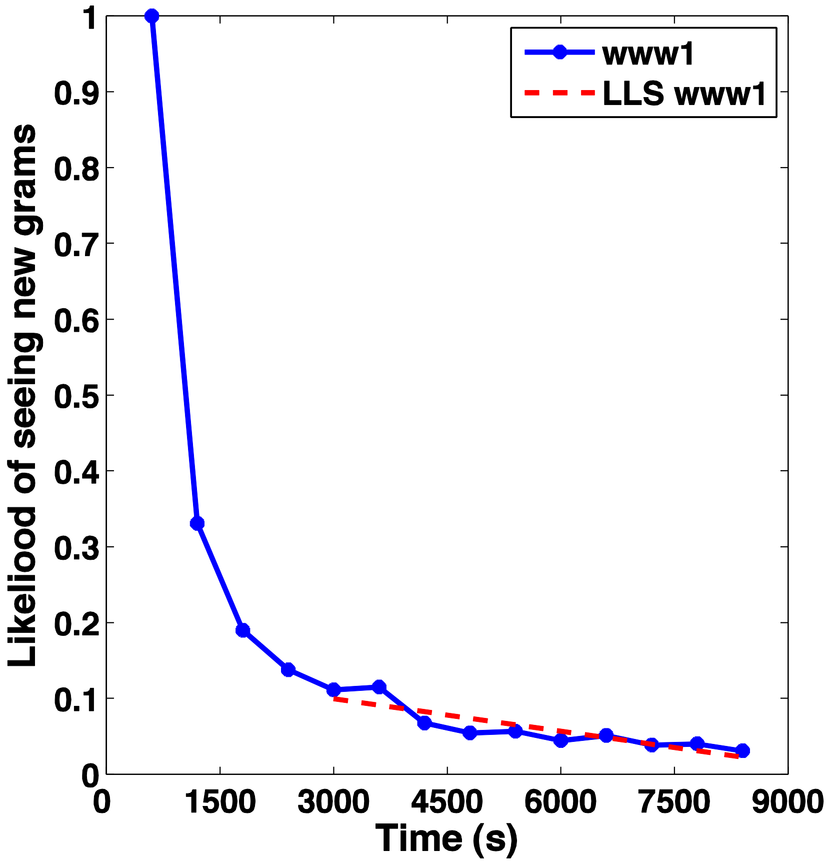
Automated Deployment of AD Sensors



- Towards fully automated AD deployment and operation:
  - identify the intrinsic characteristics of the training data (*i.e.* **self-calibration**)
  - automatically select an adaptive voting threshold (*i.e.* **self-sanitization**)

Training Dataset Stabilization

- Compute the likelihood of seeing new traffic
- Linear least squares approximation detects the stabilization point

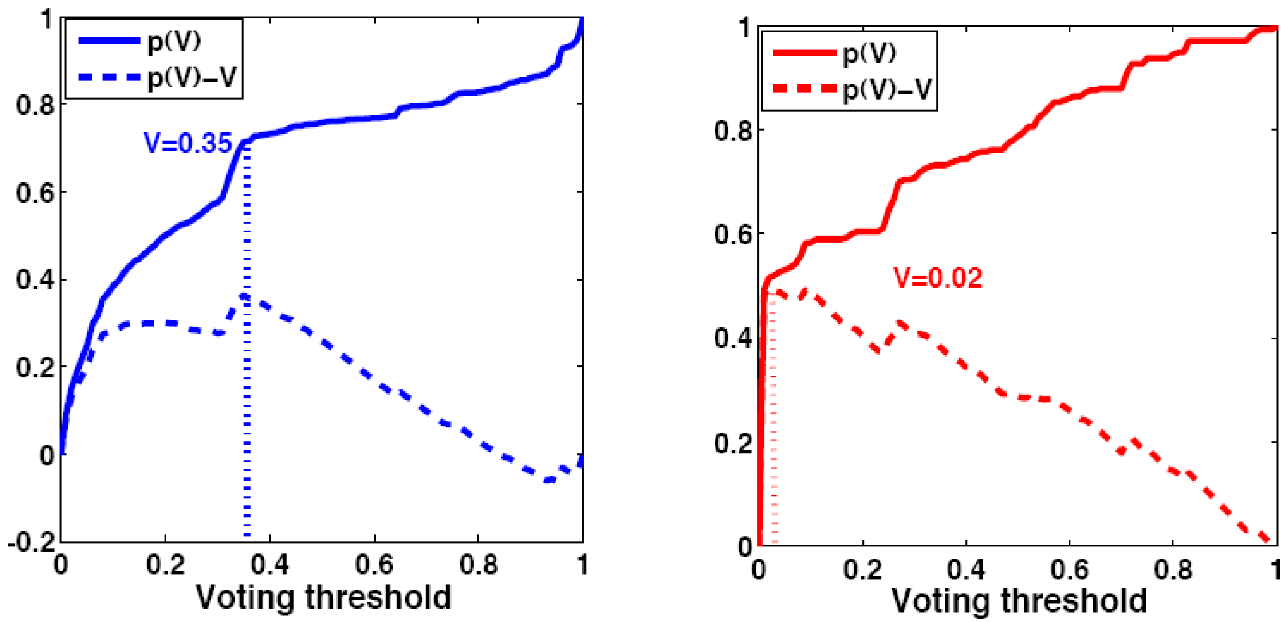


Voting Threshold Detection

- $p(V_i) = \frac{P(V_i) - P(0)}{P(1) - P(0)}$ where P(V) - number of packets

nal

- *Separation problem:*
  - find the smallest threshold (minimize V) that
  - maximizes the level of normal data (maximize p(V))



(a) (b)

Overall Performance

Parameters	www1		lists	
	FP (%)	TP (%)	FP (%)	TP (%)
N/A (no sanitization)	0.07	0	0.04	0
empirical	0.10	100	0.10	100
fully automated	**0.16**	**92.92**	**0.10**	**100**

- Self-sanitize the training data and achieve performance comparable to best empirical case

Published at RAID09: Cretu et al.

Anomaly Detection

- Self-Adaptive AD Sensors
  - Training dataset sanitization
  - Self-calibration
  - Cross-site sanitization (S&P08, Cretu et al.), extended by Boggs at el. (RAID11)
  - Model self-update (Cretu et al., NIPS Workshop 07 and RAID 09)

- Beyond enterprise network-based intrusion detection….

Outline

- Hands-free accurate anomaly detection
- Communication pattern monitoring for industrial control systems
- Anomaly detection in operational cellular networks

SRI International

Communication Pattern Monitoring for Industrial Control Systems

Industrial Control Systems (ICS)

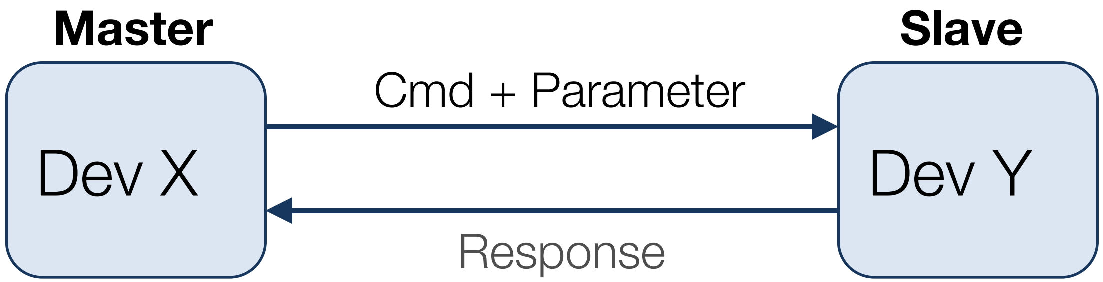
- Targeted attacks on ICS are growing in frequency and severity
  - 7,200 Internet-facing control system devices in U.S. [1]

[1] DHS ICS-CERT Monitor, October-December 2012

- ICS use specialized but insecure communication protocols
  - Enterprise security tools cannot detect zero-day attacks specific to these protocols

- ICS exhibit constrained behavior:
  - Fixed topology
  - Regular communication patterns
  - Limited number of protocols
  - Simpler protocols

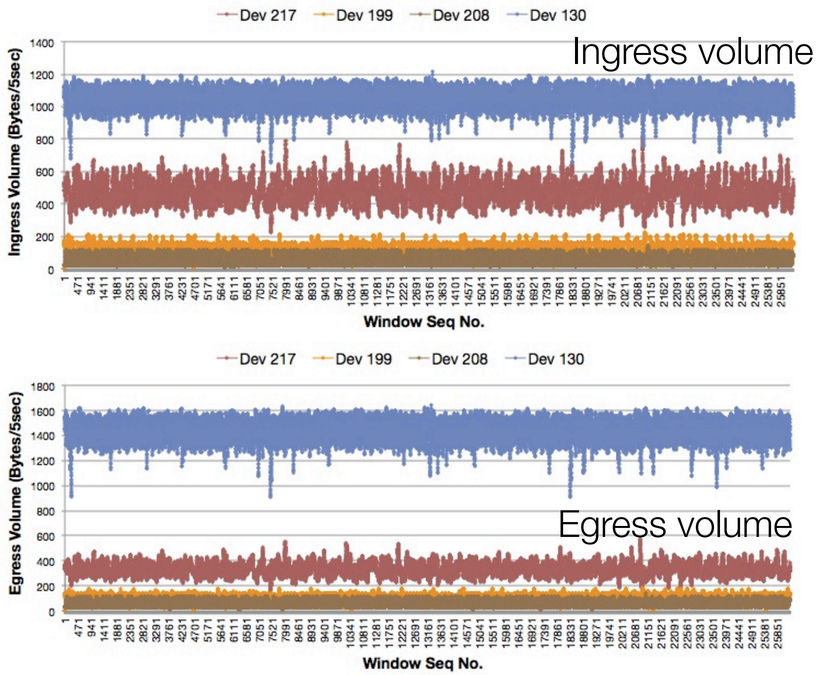
Connection Model



- Slave can receive N command types
- For the same command type,
  - Parameters can vary, but not much
  - Responses depend on the <Cmd, Parameter> pair
- Devices will have an ‘internal’ state
  - May not be directly visible
  - Operational modes, normal/compromised

Predictable Behavior of ICS Network

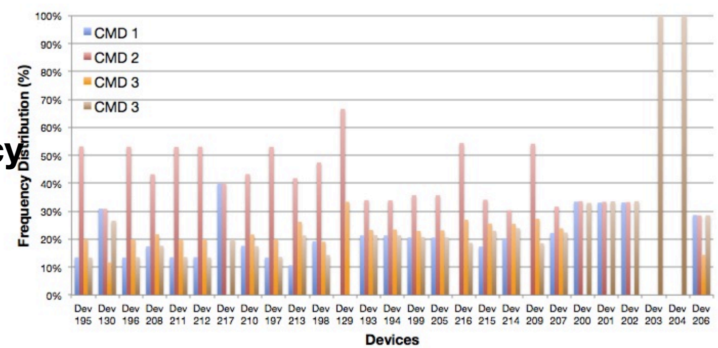
- Globally (across entire network)?
  - No. Devices behavior change with different frequencies.
- At device level?
  - Better, but still not deterministic as a device may communicate with many devices



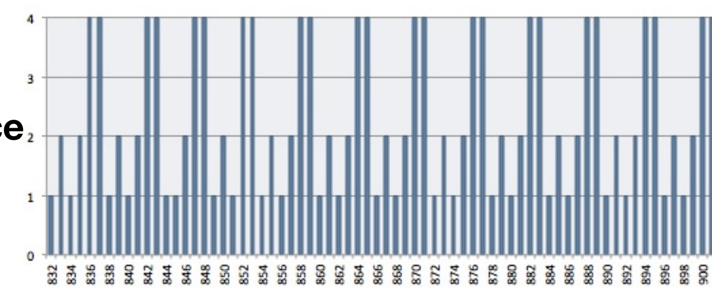
Predictable Behavior of ICS Network

- Globally (across entire network)?
  - No. Devices behavior change with different frequencies.
- At device level?
  - Better, but still not deterministic as a device may communicate with many devices
- At connection level?
  - Stable, deterministic!

Command frequency distribution

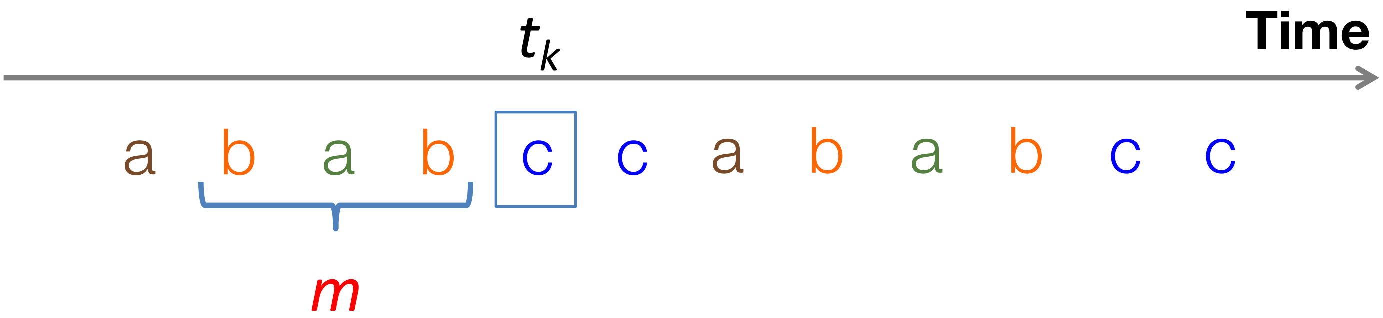


Command sequence



Can be modeled as sequence patterns

How to Model Sequence Patterns?



- What is the probability of seeing a certain command at time t_k given a history of commands of length m?

Learning Patterns of Commands and Data

- Learning the normal sequence of commands = Learning a Markov chain of order *m*

- Challenges
  - Packets can be missing
  - Patterns may vary

- Need for a probabilistic approach
  - Learn the conditional probability distribution (CPD)

$$Pr(\sigma_t | \sigma_{t-m} \cdots \sigma_{t-1})$$

Published at NDSS SENT13: Yoon and Ciocarlie

Learning Patterns Using Incremental PST

- Probabilistic Suffix Tree (PST)
  - A variable-order Markov model
  - Bounded depth (the maximum order), *L*

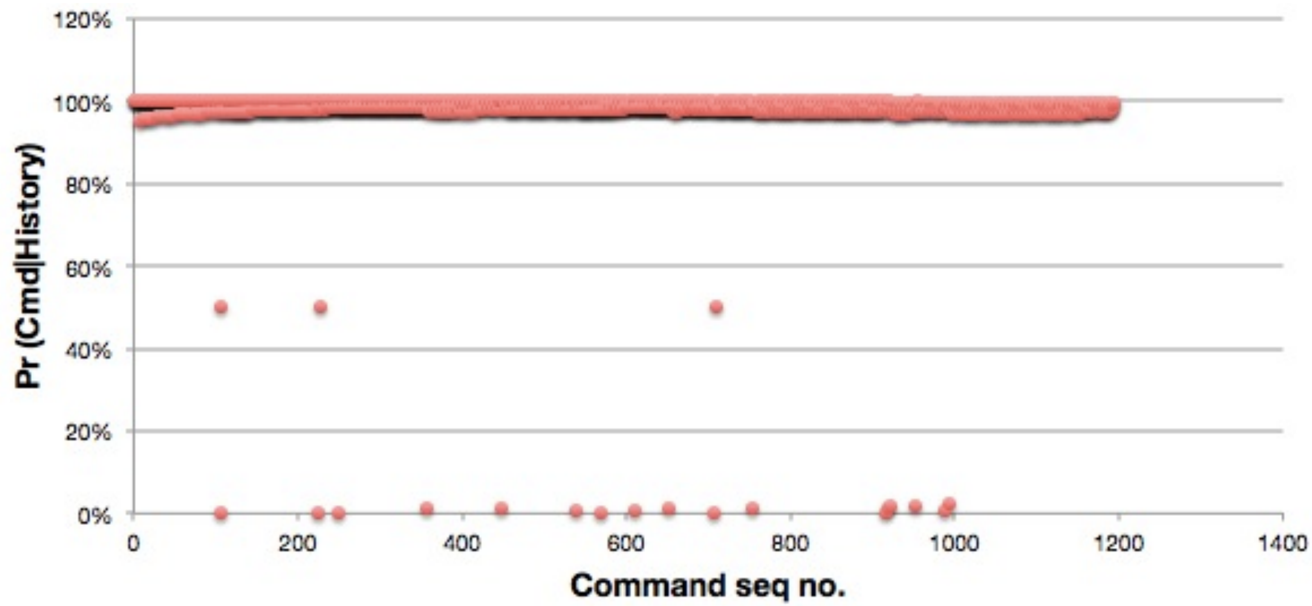
$$Pr(\sigma_t|\sigma_1\sigma_2\cdots\sigma_{t-1}) \sim Pr(\sigma_t|\sigma_{t-k}\cdots\sigma_{t-1}), k \leq L$$

  - Efficiently represents CPD using tree structure

- **Batch** learning is not applicable to network-level AD due to the flow of packets

- **Incremental** approach: update the tree whenever reading an element, **σ**
  - Keep recently-read elements
  - Update the counts for recent history of length $1..L$

Incremental PST Example



- A MODBUS connection
  - Base pattern: 1-2-1-2-4-4
  - Normal sequence
  - Most likelihoods are close to 1.0
  - Near zero values due to missing packets

False Positive Due to Missing Packets

Base pattern: 1 2 1 2 4 4 **L (MaxDepth) = 3**

1-2-1-2-4-4 - 1-2-**2**-4 - 1 **Time**

Pr(2|4-1-2) = 1.69%

- Missing one packet can cause multiple false positives
  - In this example, missing ‘1’ causes two false positives

Incremental PST with Prediction

- If $Pr(\sigma_t|\sigma_{t-L}\cdots\sigma_{t-1}) < \theta$
  - assume an element is missing and try to restore it!

- First, find what we should have seen.

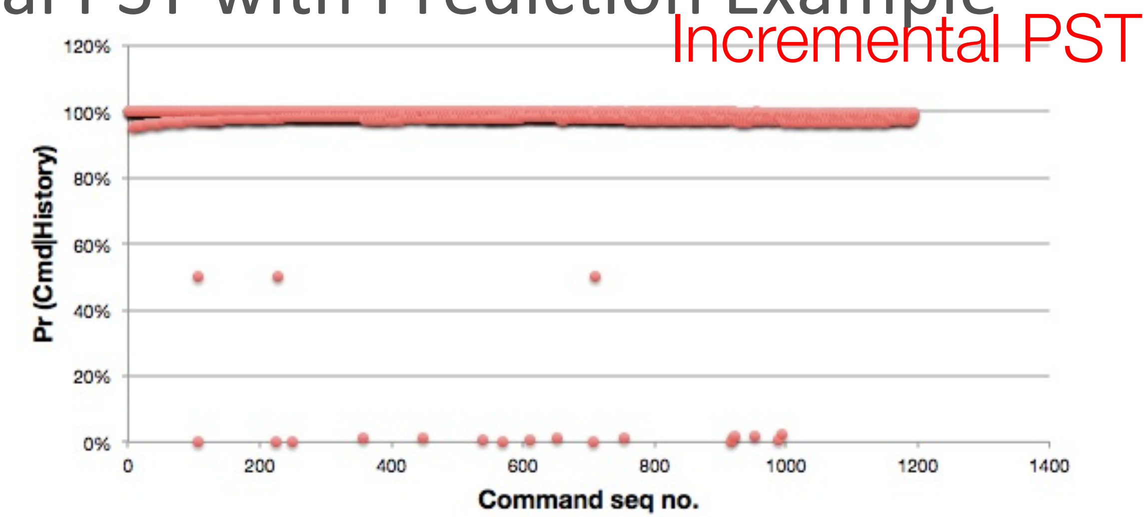
$$\sigma_{ML} = \arg\max_{\sigma} Pr(\sigma|\sigma_{t-L}\cdots\sigma_{t-1})$$

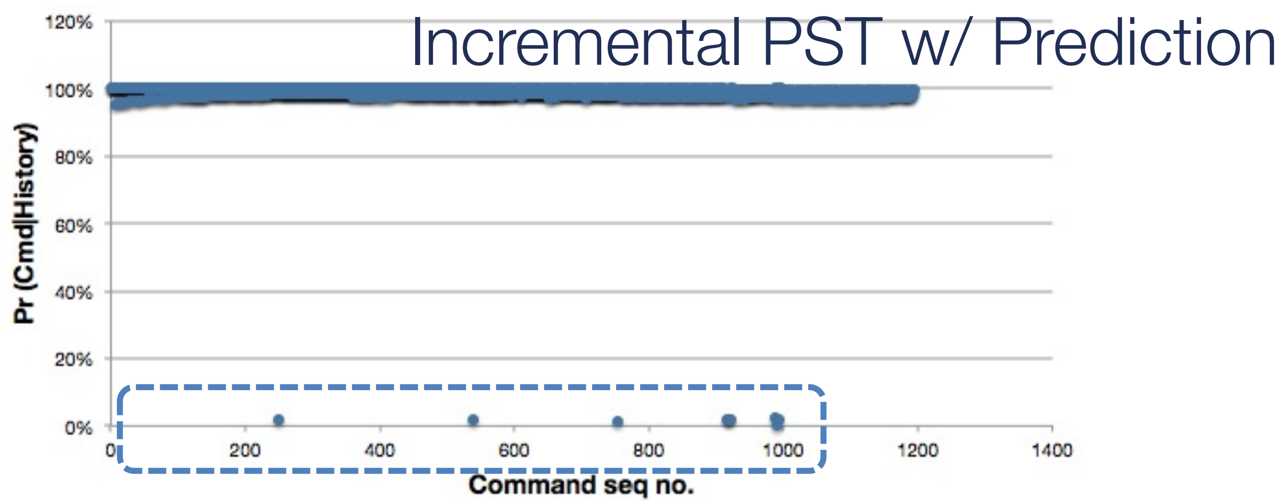
- Then, use it to calculate the new likelihood

$$\cancel{\sigma_{t-L}}\sigma_{t-L+1}\cdots\sigma_{t-1} \longrightarrow \underbrace{\sigma_{t-L+1}\cdots\sigma_{t-1}\sigma_{ML}}_{\text{Length = L}}$$

$$\boxed{\begin{aligned} &Pr(\sigma_t|\sigma_{t-L}\cdots\sigma_{t-1}) \\ &\quad\sim Pr(\sigma_{ML}|\sigma_{t-L}\cdots\sigma_{t-1})\cdot Pr(\sigma_t|\sigma_{t-L+1}\cdots\sigma_{t-1}\sigma_{ML}) \end{aligned}}$$

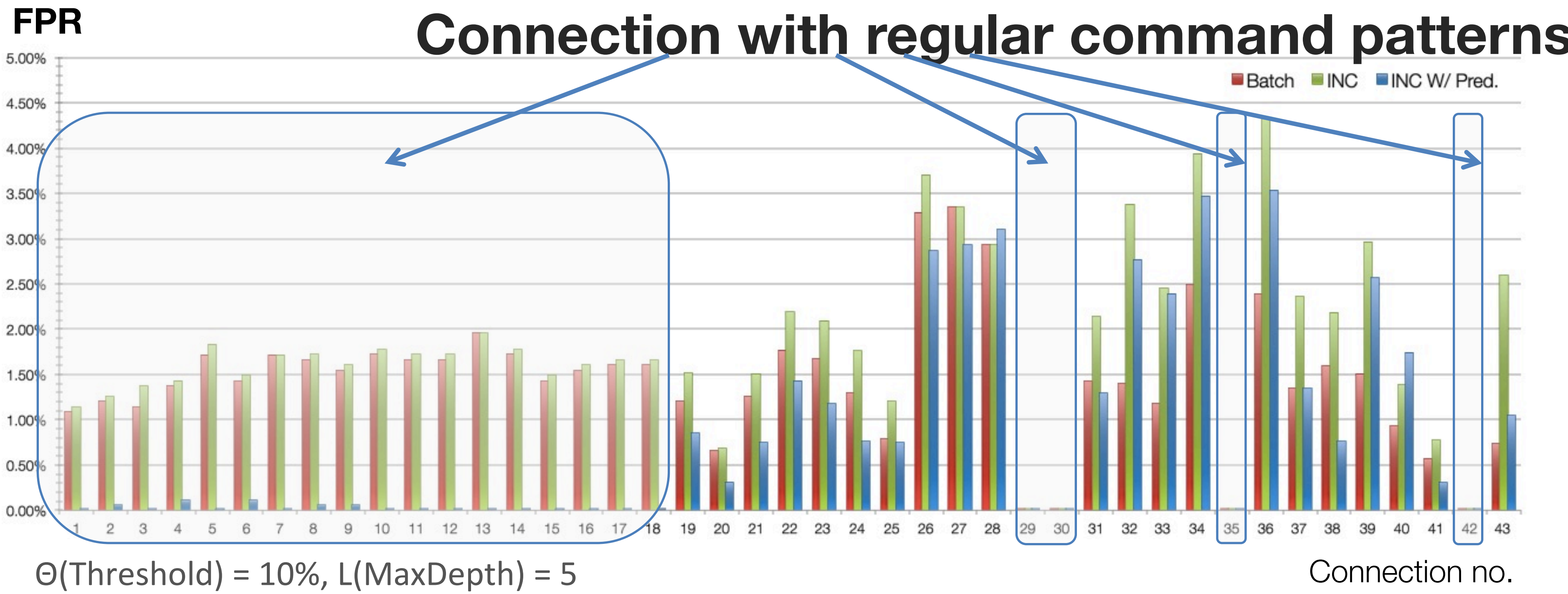
Incremental PST with Prediction Example





Significantly reduced FP rate, unless consecutive packets are missing.

False Positive Rates of Modbus Traffic



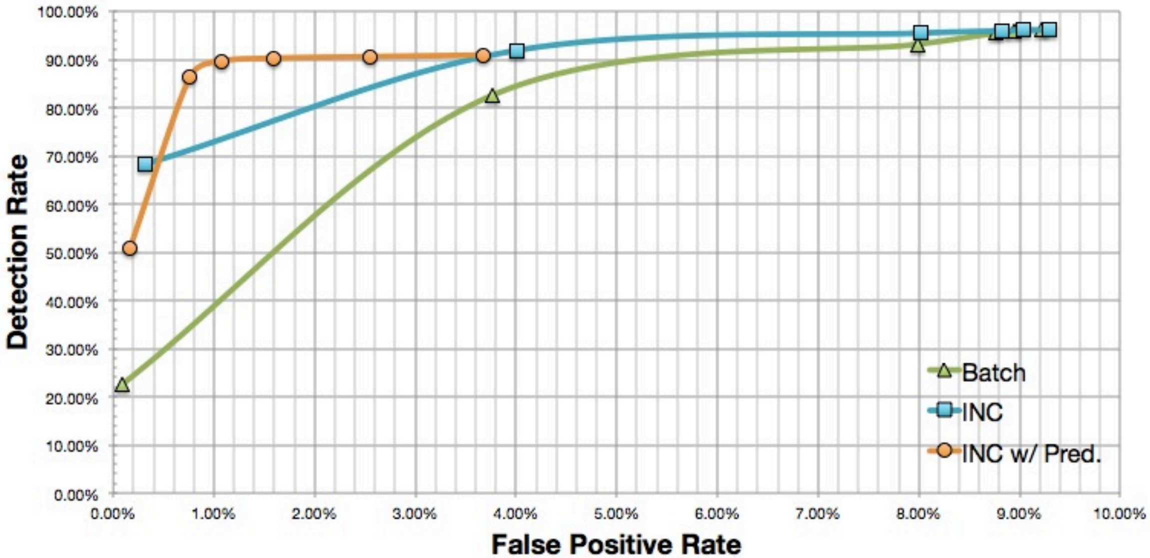
Θ(Threshold) = 10%, L(MaxDepth) = 5

- Real Modbus traffic
  - 2 masters, 25 slaves, 86 connections (43 pairs)
  - 4 cmd types
  - No attack/anomaly is known; some packets are missing

Evaluation – synthetic data

- Generate a random base pattern
- Then, generate a random sequence based on the pattern
  - With a missing probability, a command can be dropped
  - With an attack probability, a random short sequence is inserted
- Input parameters
  - Min, max of base pattern length
  - # of command types
  - Missing, attack probabilities

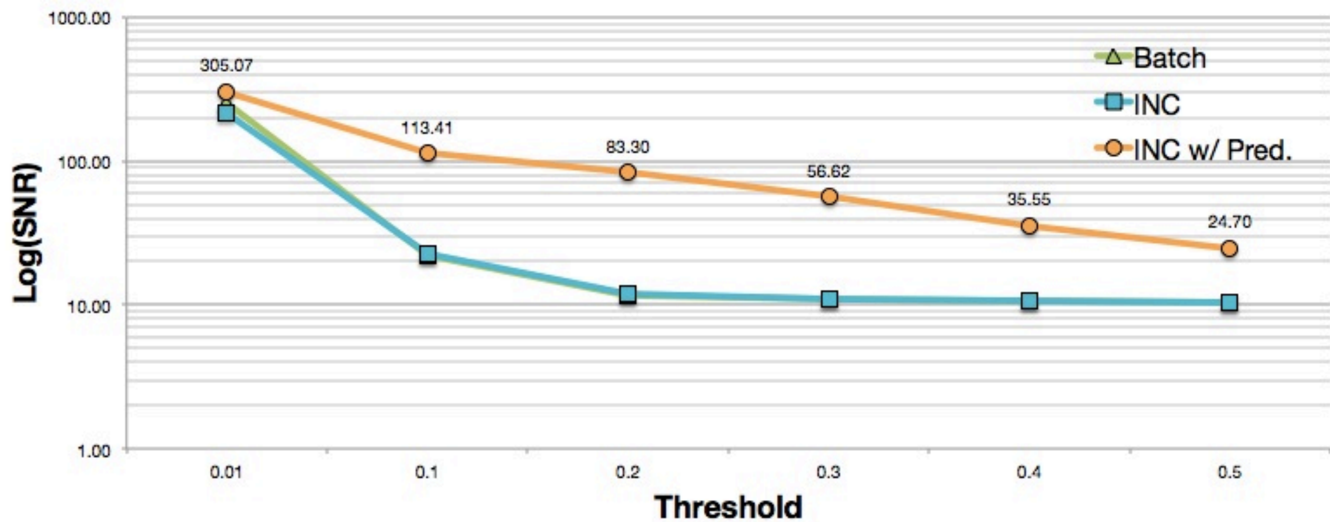
Evaluation



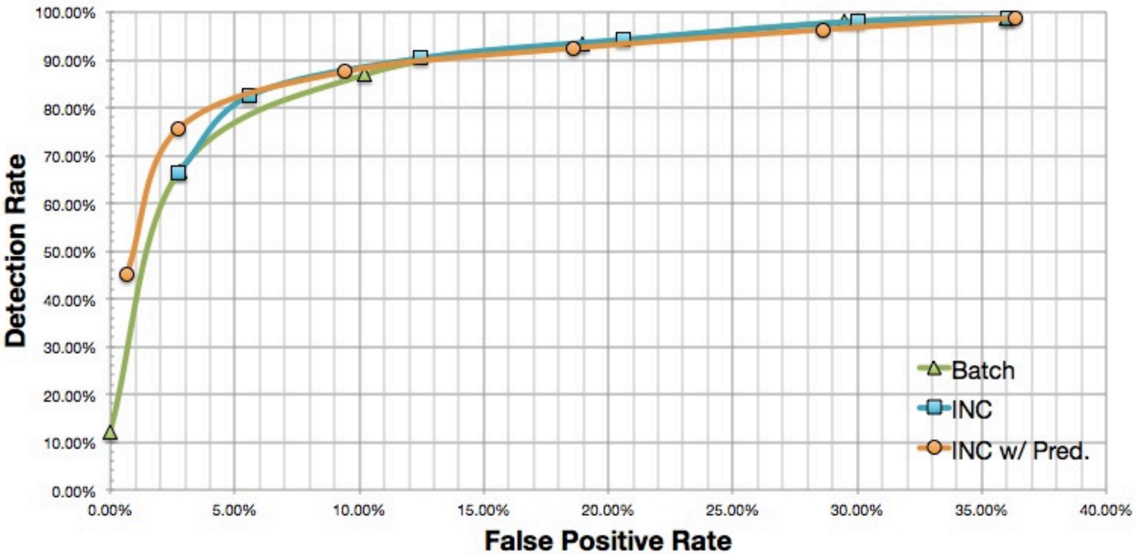
ROC curve

Miss prob = 10%
MaxDepth(L) = 5

Signal-Noise Ratio



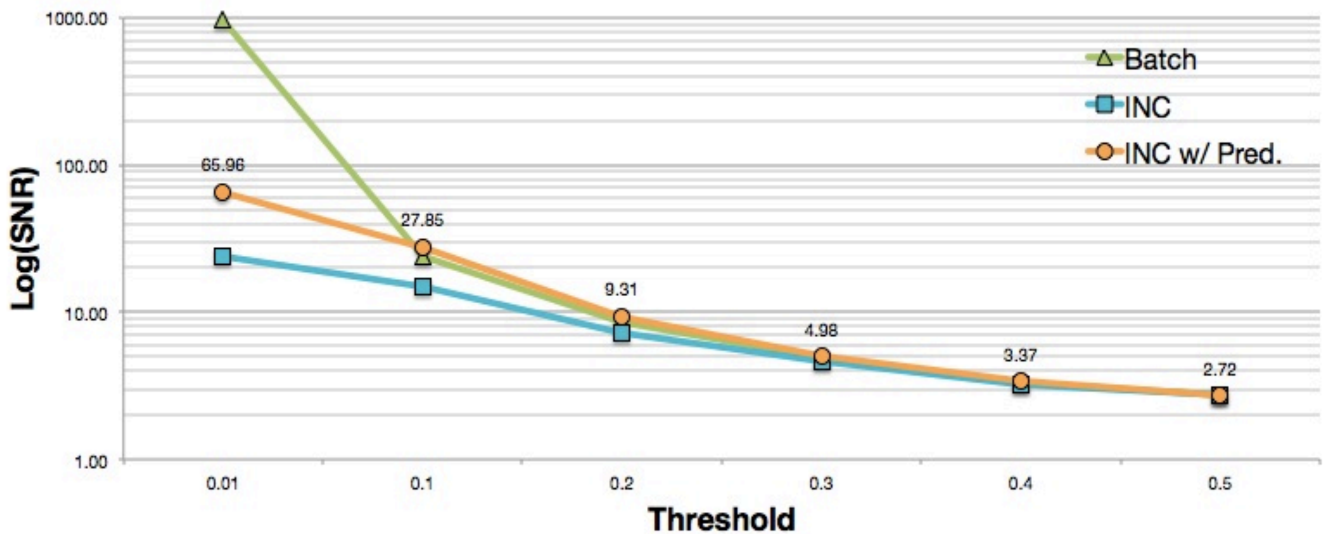
Evaluation



ROC curve

Miss prob = 50%
MaxDepth(L) = 5

Signal-Noise Ratio



Communication pattern monitoring for ICS

- A new probabilistic-suffix-tree-based approach for ICS anomaly detection, which extracts the normal patterns of command and data sequences from ICS communications

- A false positive rate reduction mechanism, instrumental for ICS environments

- An implementation of the proposed approached applied to both real and simulated datasets

Outline

- Hands-free accurate anomaly detection
- Communication pattern monitoring for industrial control systems
- Anomaly detection in operational cellular networks

SRI International

Anomaly Detection in Operational Cellular Networks

Problem: SON Coordination/Verification

- Networks can suffer degradations if actions that change network-element configurations are not coordinated
- Mitigation: SON verification
  - must occur fast in order to correlate the detection results and diagnose the system
- Key problem: modeling network/subnetwork state

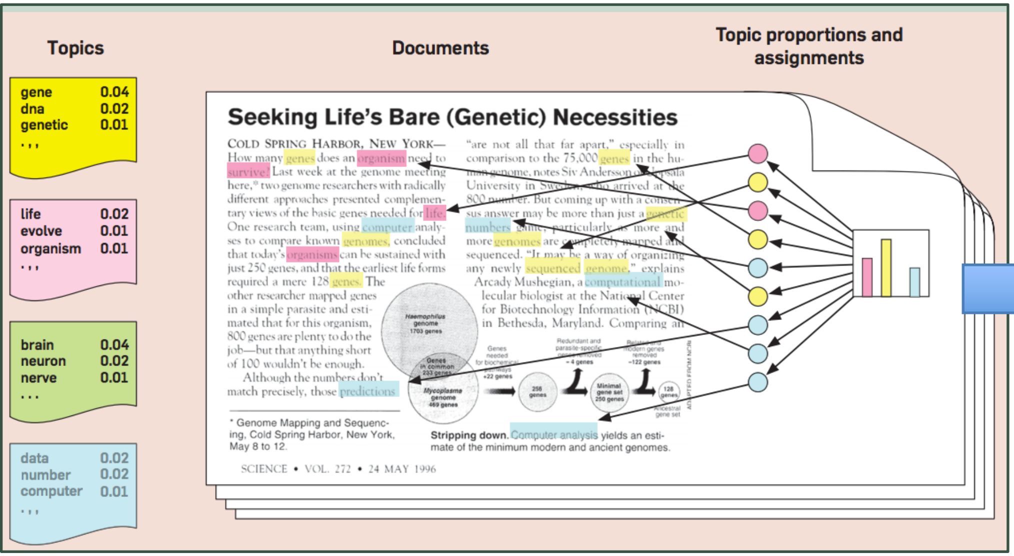
Published at MONAMI14, IWSON14: Ciocarlie et al.

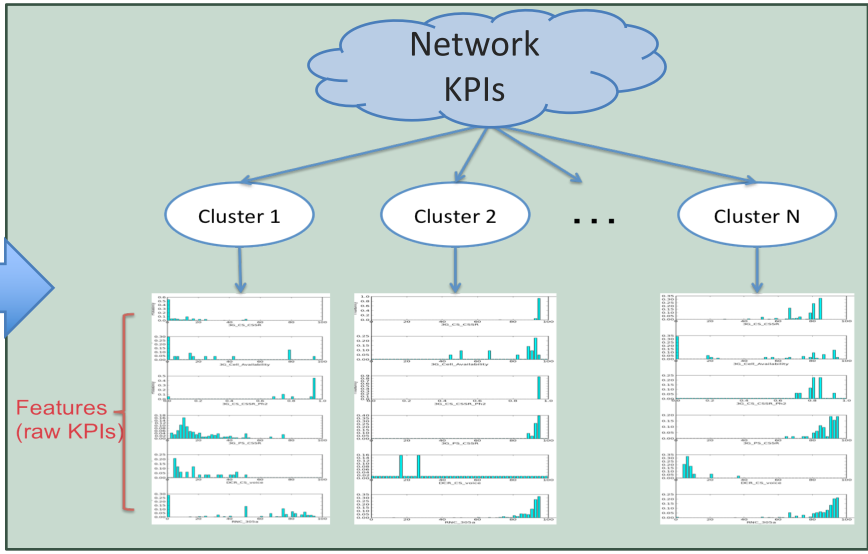
Clustering Module → Interpretation Module → Detection Module

Clustering Module
uses Probabilistic Topic Modeling

- Discover and annotate large archives of documents with thematic information

- Discover "topics"/states in a cellular network
- Determine the number of clusters automatically using a Hierarchical Dirichlet Process (HDP) approach



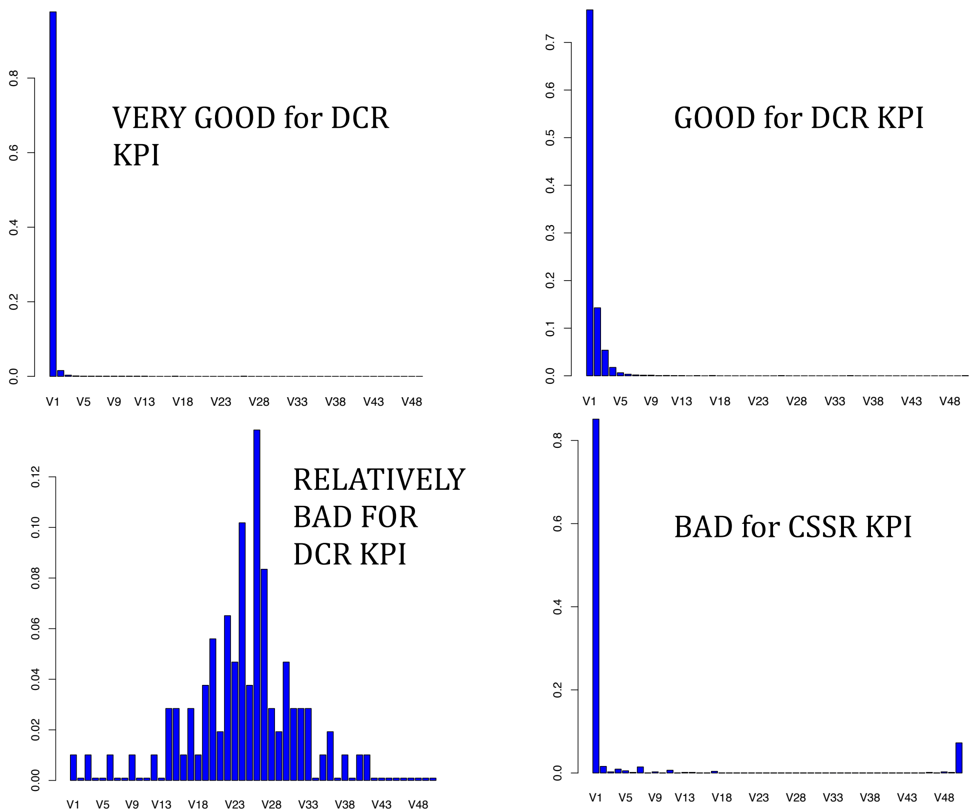


Communications of the ACM, April 2012, Vol 55, No.4, p.78

Cluster Interpretation Module

uses KPI Characteristics

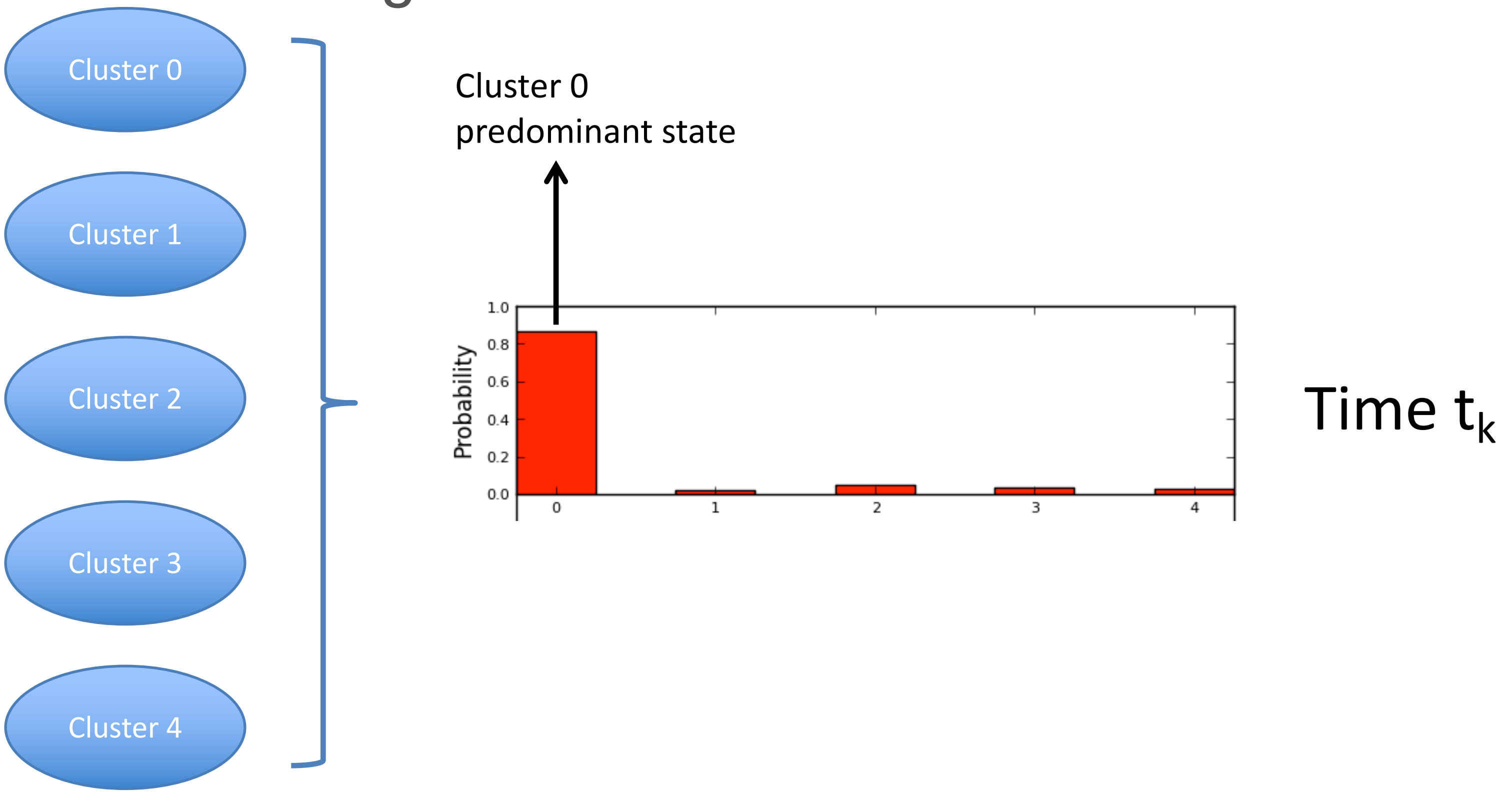
- Automatically classifies each cluster as either normal or abnormal based on KPI characteristics
  - KPIs that should not increase (e.g., drop call rate) or decrease (e.g., call success rate) beyond a certain threshold



Detection Module

uses Topic Modeling

- At every timestamp t_k a set of **cluster mixture weights** is generated indicating the state of the network



Diagnosis of Network-Level Anomalies Using Markov Logic Networks (MLNs)

- MLNs combine first-order logic and probabilistic models in a single representation (Richardson and Domingos, 2006)
- MLNs are first-order knowledge bases with a weight attached to each rule
  - Weights can be learned over time as examples arise
  - Contradictions OK; missing data OK

1.5	$\forall x \; Smokes(x) \Rightarrow Cancer(x)$
1.1	$\forall x, y \; Friends(x,y) \Rightarrow \left(Smokes(x) \Leftrightarrow Smokes(y)\right)$

- MLNs compute the "most likely explanation" for an event given the data

- SRI has a very efficient state-of-the-art MLN solver called the Probabilistic Consistency Engine (PCE)

Combine MLN and Topic Modeling

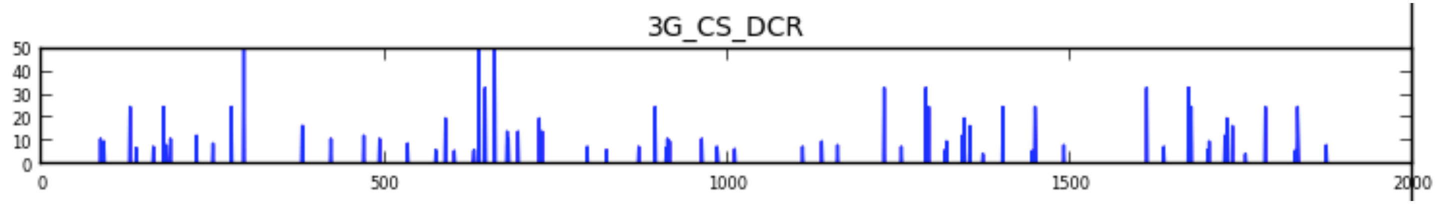
- Use network state information as extracted by topic modeling in the MLN inference
  - Use Principal Component Analysis (PCA) to identify groups of cells that exhibit similar behavior
  - Reason over groups of cells to reduce the number of entities

- The MLN provides the most likely explanation for the state of the network
  - Reasoning over configuration management (CM) and external factor information

Real Dataset

- 3G dataset for January-March 2013, 1583 timestamps
  - ~ 9000 cells (~ 4000 valid)
  - 11 non-periodic KPIs (3G_CS_CSSR, 3G_CS_DCR, 3G_Cell_Availability, 3G_CS_CSSR_Ph1, 3G_CS_CSSR_Ph2, 3G_CS_CSSR_Ph3, 3G_PS_CSSR, 3G_PS_DCR, DCR_CS_voice, Retainability_PS_Rel99, RNC_305a)

Example:
3G_CS_DCR

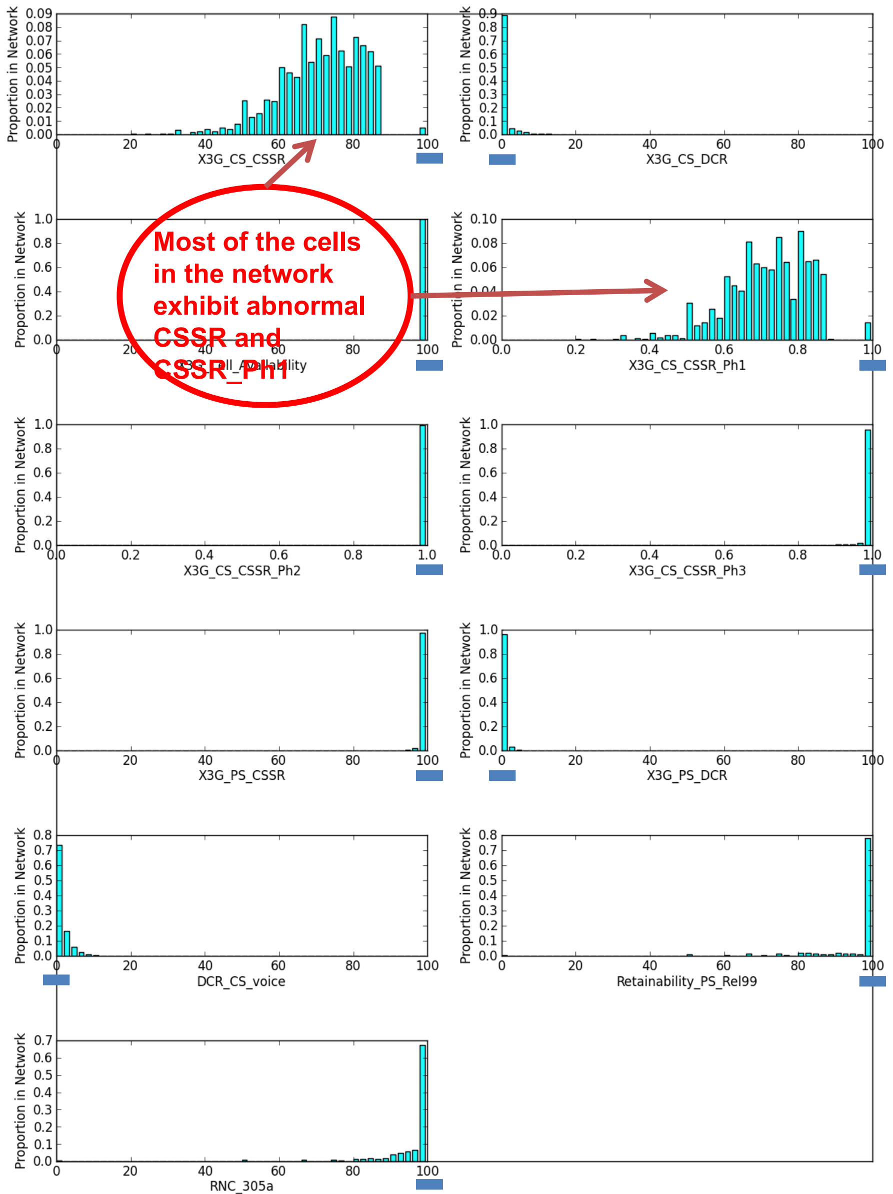


- Shortcomings
  - Many data points are missing
  - No ground truth information associated with the provided dataset
  - Hourly KPI and daily CM data

Hierarchical Dirichlet Process

- 32 topic modeling states learned from the process
  - 15 normal
  - 17 abnormal

- Example abnormal state: #8
  - Corresponds to an anomaly condition in mid-Feb
  - Shows abnormal condition with 3G_CS_CSSR and 3G_CS_CSSR_Ph1



MLN Inference

Input

```
sort Group_t;
sort Precip_t;

const G1, G2, G3, ... G486: Group_t;

…
# All anomalies
assert anomaly(G258);
assert anomaly(G259);
assert anomaly(G265);
assert anomaly(G316);
assert anomaly(G325);
assert anomaly(G344);
assert anomaly(G365);
assert anomaly(G386);
assert anomaly(G401);
assert anomaly(G438);
```

Anomalies from Topic Modeling

```
….
add wcel_angle_changed(G10) 36.0;
add wcel_angle_changed(G364) 2.0;
add wcel_angle_changed(G9) 40.5;
add wcel_angle_changed(G31) 21.0;
add wcel_angle_changed(G290) 1.0;
add wcel_angle_changed(G229) 3.0;
add wcel_angle_changed(G45) 8.0;
add wcel_angle_changed(G68) 6.5;
add wcel_angle_changed(G294) 5.0;
add wcel_angle_changed(G83) 8.5;
add wcel_angle_changed(G72) 13.0;
```

CM data (e.g. wcel_angle change)

```
…..

 add [G] (precip(G, LIGHT-SNOW) and anomaly(G)) implies weather_event(G) 0.1;
 add [G] (precip(G, SNOW) and anomaly(G)) implies weather_event(G) 0.5;
 add [G] (precip(G, HEAVY-SNOW) and anomaly(G)) implies weather_event(G) 2.0;
 add [G] (wcel_angle_changed(G) and anomaly(G)) implies cm_event(G) 4.0;
 add [G] cm_event(G) implies not weather_event(G) 1.0;
 add [G] (anomaly(G) and (not weather_event(G)) and (not cm_event(G))) implies
hw_event(G) 5.0;
```

Rules and weights

Output

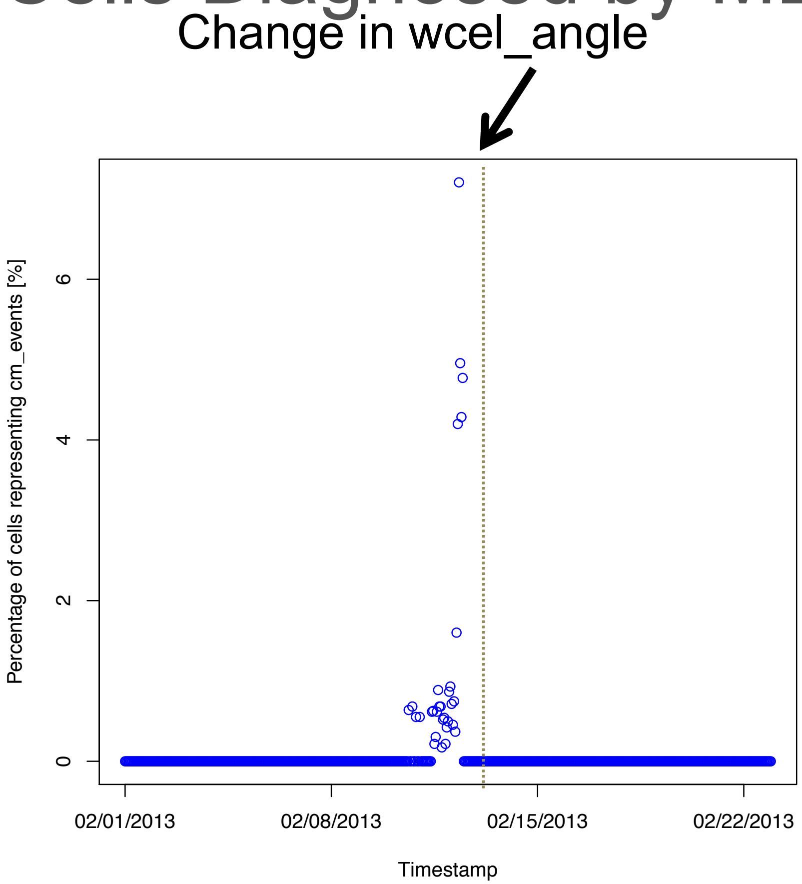
```
8 results:
[x <- G325] 0.968: (cm_event(G325))
[x <- G265] 0.955: (cm_event(G265))
[x <- G316] 0.940: (cm_event(G316))
[x <- G365] 0.937: (cm_event(G365))
[x <- G344] 0.876: (cm_event(G344))
[x <- G386] 0.828: (cm_event(G386))
[x <- G259] 0.768: (cm_event(G259))
[x <- G258] 0.749: (cm_event(G258))
```

Groups of cells affected by CM changes

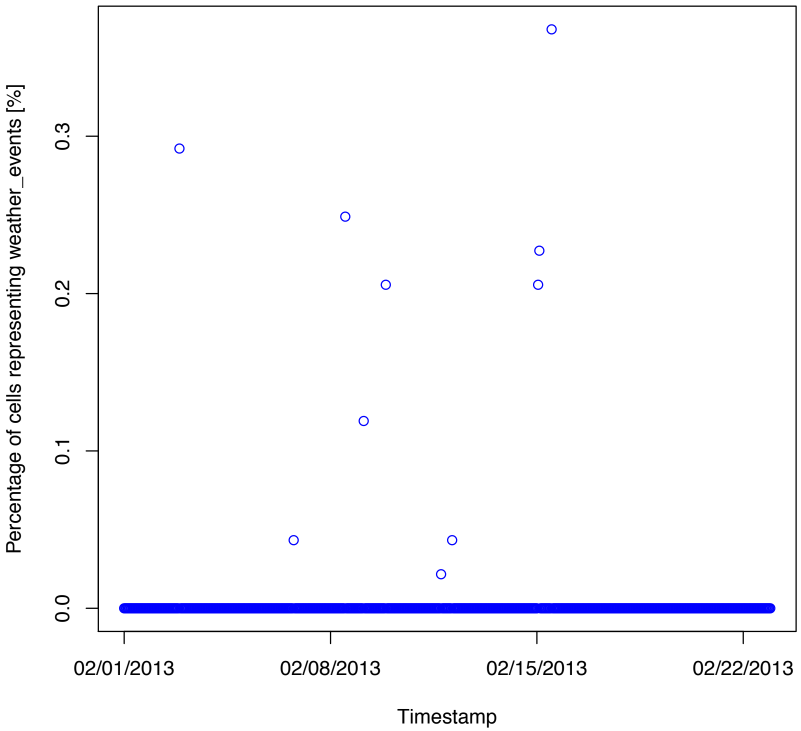
```
0 results:

0 results:

476 results:
[x <- G119] 1.000: (normal(G119))
[x <- G2] 1.000: (normal(G2))
[x <- G3] 1.000: (normal(G3))
[x <- G4] 1.000: (normal(G4))
[x <- G376] 1.000: (normal(G376))
[x <- G377] 1.000: (normal(G377))
[x <- G378] 1.000: (normal(G378))
[x <- G379] 1.000: (normal(G379))
[x <- G380] 1.000: (normal(G380))
[x <- G381] 1.000: (normal(G381))
[x <- G382] 1.000: (normal(G382))
[x <- G383] 1.000: (normal(G383))
[x <- G385] 1.000: (normal(G385))
[x <- G387] 1.000: (normal(G387))
[x <- G388] 1.000: (normal(G388))
[x <- G389] 1.000: (normal(G389))
[x <- G392] 1.000: (normal(G392))
[x <- G393] 1.000: (normal(G393))
```

Normal groups of cells

Cells Diagnosed by MLN



Percentage of cells diagnosed as anomalous due to CM changes



Percentage of cells diagnosed as anomalous due to weather events

Operational Cellular Networks

- SON verification
  - Tested on KPI, CM and weather data from a real operational cell network
  - Topic modeling detects anomalies at a large scale
  - MLN performs diagnosis within groups of cells

Recap

- New methods for detecting intrusions, performance degradation, and other anomalous behaviors
  - capture the normal behavior of a system
  - detect departures from normality and attribute causes
- Industrial control systems
  - probabilistic-suffix-tree-based approach to extract normal patterns of command and data sequences
- Mobile broadband networks
  - model cell behavior based on key performance indicators to identify partial and complete degradations
  - model the state of the network within a larger scope to verify configuration management parameters changes

SRI International

Questions?

Headquarters
333 Ravenswood Avenue
Menlo Park, CA 94025
+1.650.859.2000

Princeton, NJ
201 Washington Road
Princeton, NJ 08540
+1.609.734.2553

Additional U.S. and
international locations

www.sri.com