MIDAS: Microcluster-Based Detector of Anomalies in Edge Streams

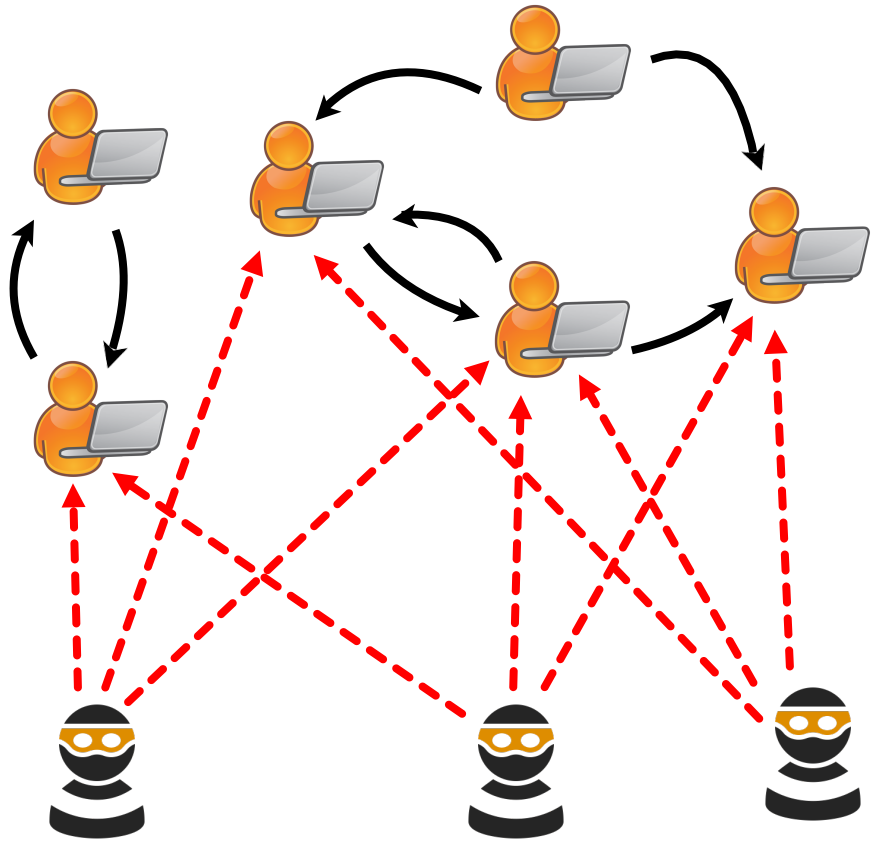
Siddharth Bhatia, Bryan Hooi, Minji Yoon, Kijung Shin, Christos Faloutsos


siddharth@comp.nus.edu.sg

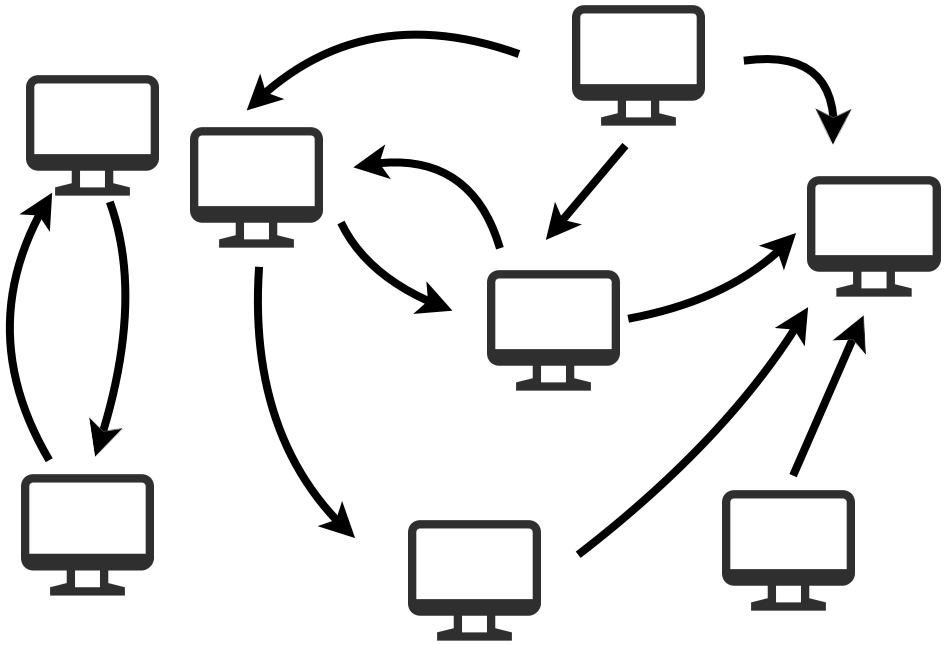

@siddharthb_


DoS ATTACK
DoS ATTACK
DoS ATTACK
DENIAL-OF-SERVICE
DoS ATTACK
DoS ATTACK
DoS ATTACK
DoS ATTACK
DENIAL-OF-SERVICE
DoS ATTACK
DENIAL-OF-SERVICE
DoS ATTACK

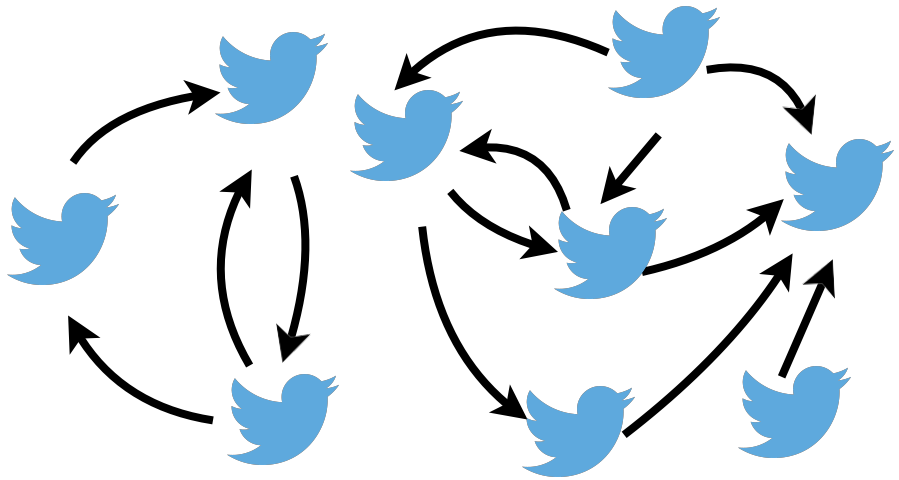
Intrusion Detection



Time Evolving Graphs

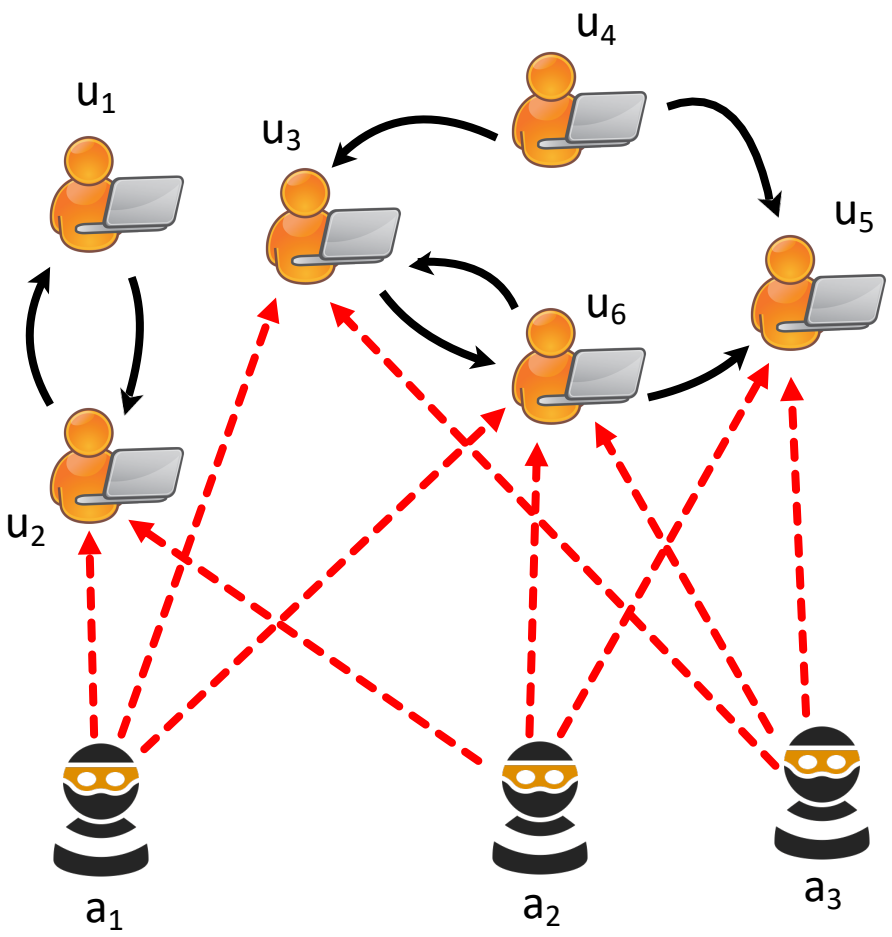


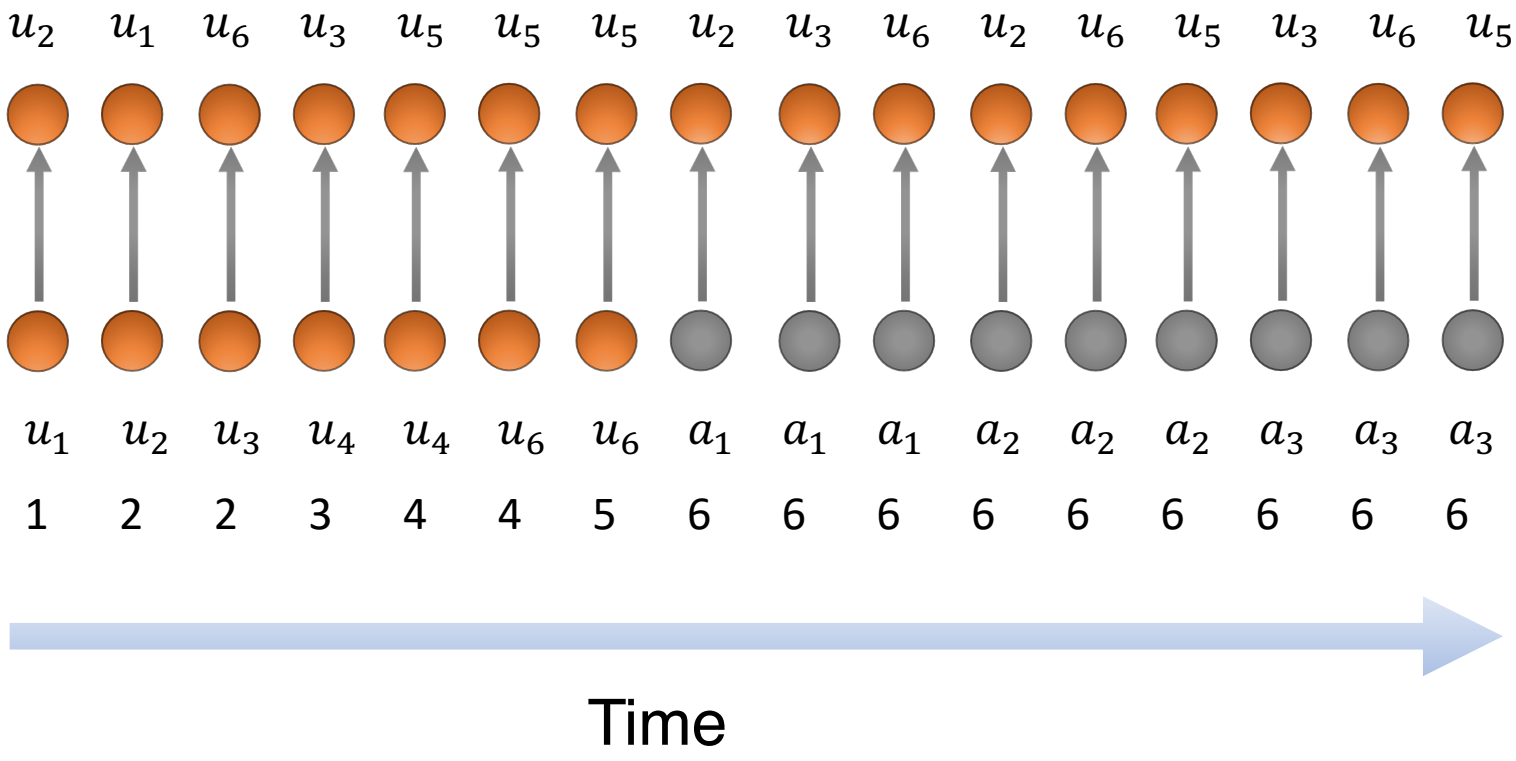
Computer Networks



Twitter / IM Networks

Edge Streams





Roadmap

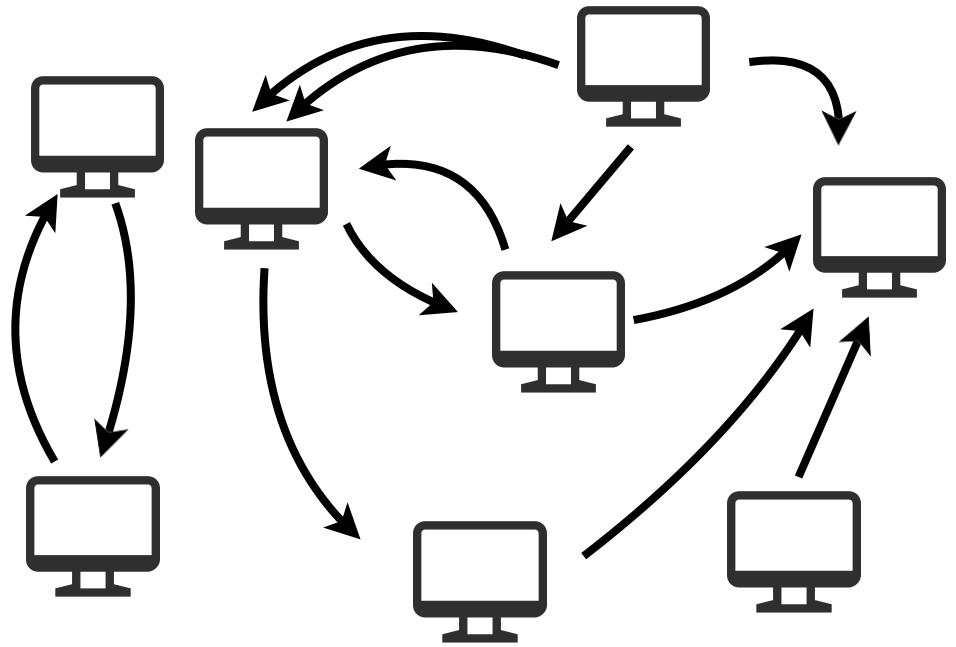
- **Problem**
- Algorithm
  - MIDAS
  - MIDAS-R
- Related Work
- Experiments
- Future Work

Problem

Input:
- Edge stream $\boldsymbol{E}$ from time evolving graph $\boldsymbol{G}$
- Directed, multigraph, discrete time

Output:
- Anomaly Score for each edge

Our Contributions:
- Microcluster Detection
- Guarantees on False Positive Probability
- Constant Memory
- Constant Update Time



Roadmap

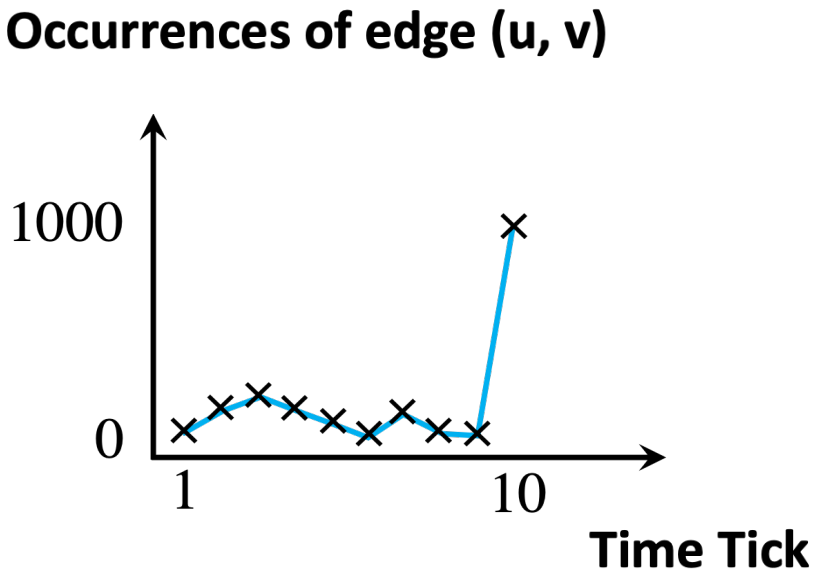
- Problem
- **Algorithm**
  - **MIDAS**
  - MIDAS-R
- Related Work
- Experiments
- Future Work

Detecting Microcluster

Background

Offline:
- Time series methods

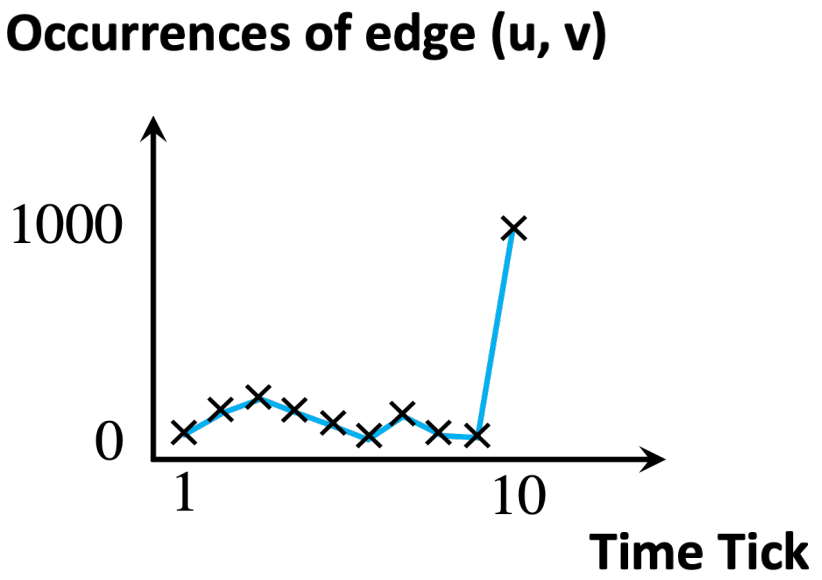
Online:
- Bounded memory
- Set of nodes is not fixed



Naive Solution

Gaussian distribution:

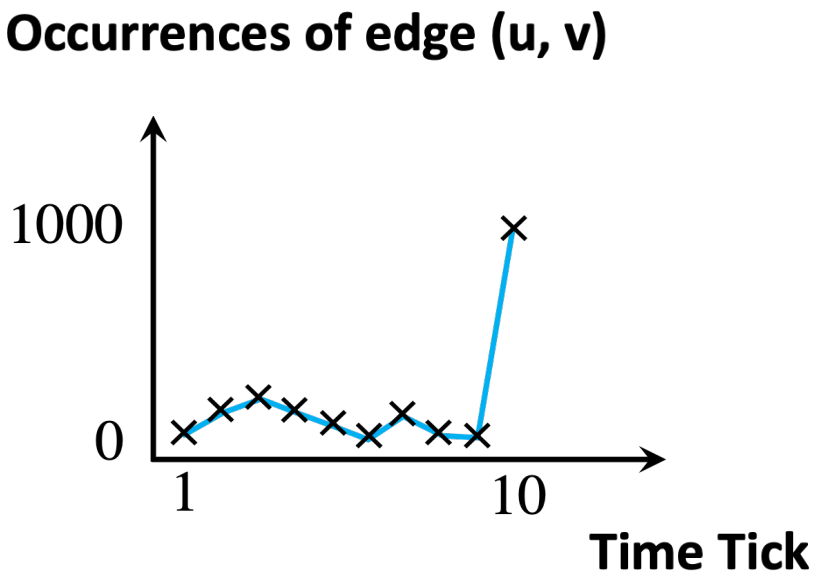
- Find mean & standard deviation
- Compute Gaussian likelihood
  (edge occurrences at t)
- if $likelihood < threshold$, declare anomaly



Our assumption

Mean Level at $t = 10$ equals Mean Level for $t < 10$

	Past (t<10)	**Current** (t=10)
Expected	900	100
Observed	0	1000

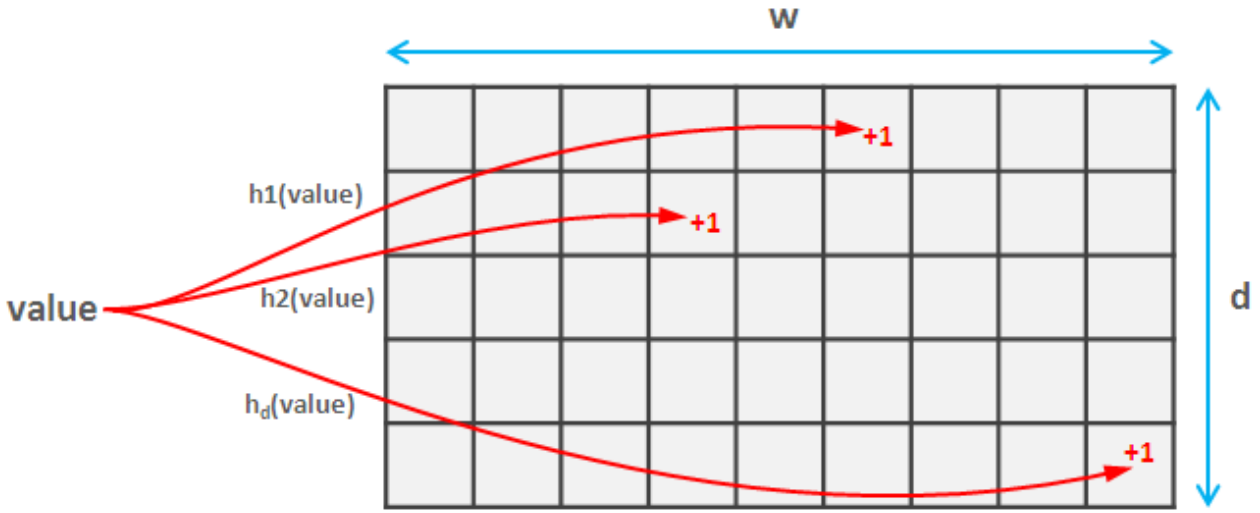


Streaming Data Structure: CMS

Details

s_{uv} : $u - v$ edges up to time t
a_{uv} : $u - v$ edges at current time t

$\hat{s}_{uv}$: Approximate total count
$\hat{a}_{uv}$: Approximate current count



$\hat{a}_{uv} \leq a_{uv} + \upsilon N_t$ with probability at least $1 - \varepsilon$

υ is the amount of error we can tolerate.
$1 - \varepsilon$ is the probability.
e.g. with 99% probability only up to 0.5% error

Anomaly Score: Chi-Squared Test

Details

$$X^2 = \frac{(\text{observed}_{(t=10)} - \text{expected}_{(t=10)})^2}{\text{expected}_{(t=10)}} + \frac{(\text{observed}_{(t<10)} - \text{expected}_{(t<10)})^2}{\text{expected}_{(t<10)}}$$

$$score((u, v, t)) = (\hat{a_{uv}} - \frac{\hat{s_{uv}}}{t})^2 \frac{t^2}{\hat{s_{uv}}(t-1)}$$

observed expected

Roadmap

- Problem
- **Algorithm**
  - MIDAS
  - **MIDAS-R**
- Related Work
- Experiments
- Future Work

MIDAS-R: Incorporating Relations

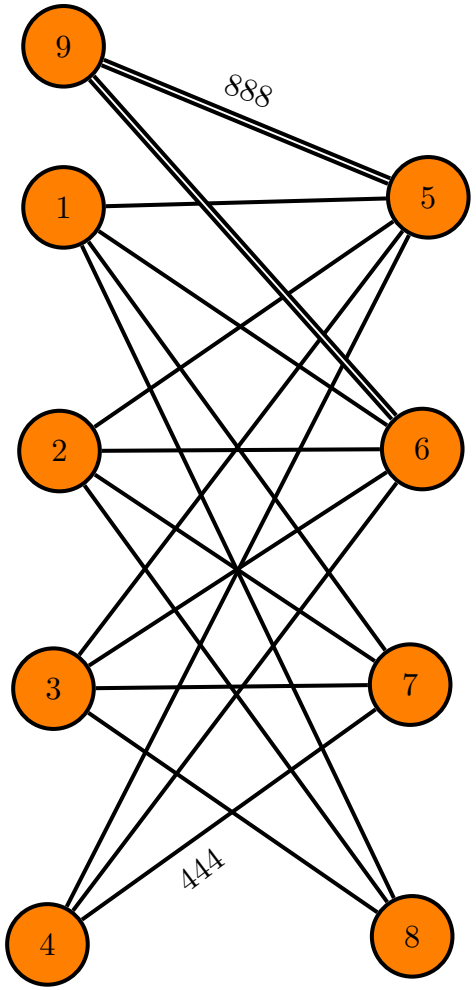
Details

Temporal Relations:

- Allows past edges to count toward the current time tick
- Diminishing weight
- At end of every time tick t, reduce counts by $\alpha \in [0,1]$

Spatial Relations:

- Catch large groups of spatially nearby edges
- s_u : edges adjacent to node u up to time t
- a_u : edges adjacent to node u at current time t
- $\max(score(u,v,t),\ score(u,t), score(v,t))$

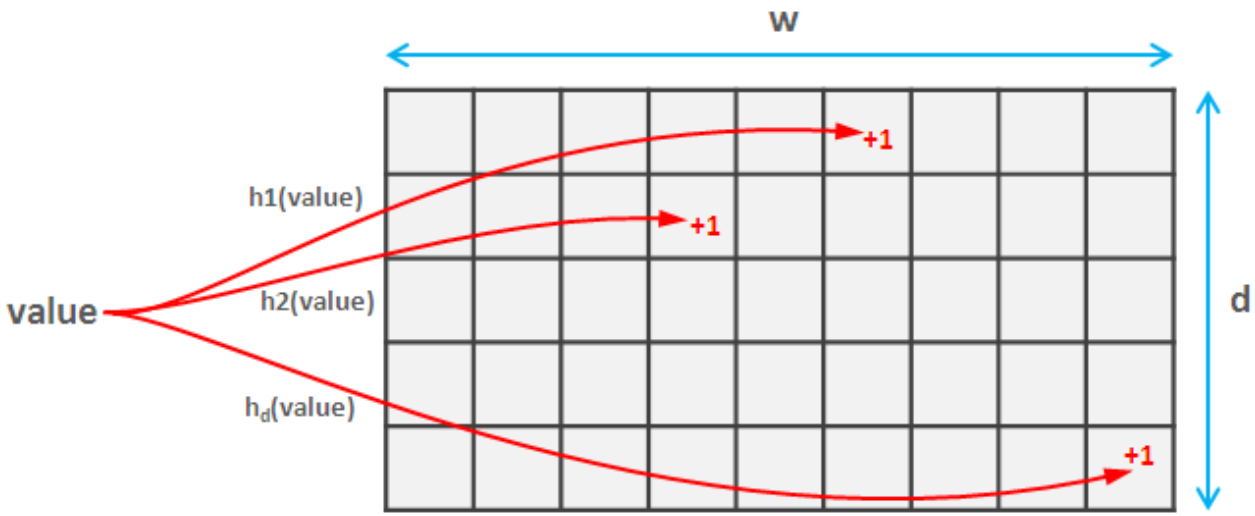


Time and Memory Complexity

d: number of hash functions
w: number of buckets

Space complexity:
- $O(wd)$

Time complexity:
- $O(d)$



Roadmap

- Problem
- Algorithm
  - MIDAS
  - MIDAS-R
- **Related Work**
- Experiments
- Future Work

Related Work

Anomaly Detection in Static Graphs:

- **Node detection:** CatchSync [2016], OddBall [2010]
- **Subgraph detection:** [Shin et al., 2018], FRAUDER [2017]
- **Edge detection:** NrMF [2011], Autopart [2004]

Anomaly Detection in Graph Streams:

- **Node detection:** DTA [2006]
- **Subgraph detection:** CatchSync [2016], CopyCatch [2013]
- **Edge detection:** AnomRank [2019], SpotLight [2018]

Anomaly Detection in Edge Streams:

- **Node detection:** [Yu et al., 2013]
- **Subgraph detection:** DenseAlert [2017]
- **Edge detection:** **SedanSpot** [2018], **RHSS** [2016]

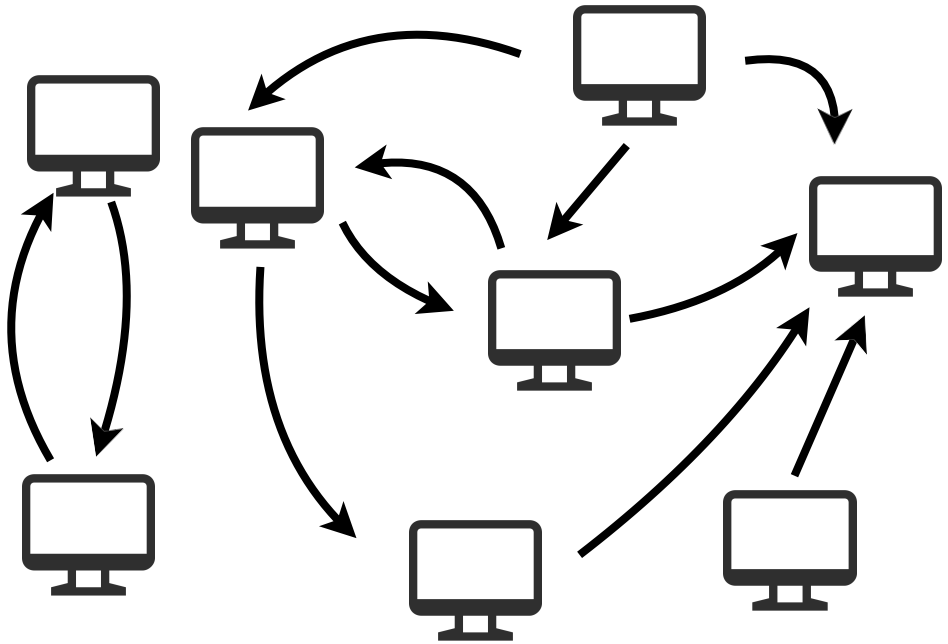
Comparison

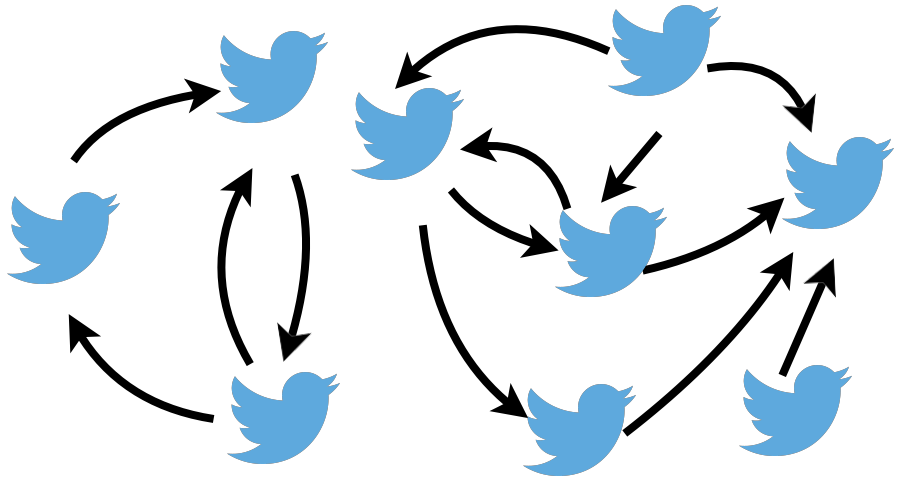
	SedanSpot(2018)	RHSS(2016)	Midas
Microcluster Detection			✔
Guarantee on False Positive Probability			✔
Constant Memory	✓	✓	✔
Constant Update Time	✓	✓	✔

Roadmap

- Problem
- Algorithm
  - MIDAS
  - MIDAS-R
- Related Work
- **Experiments**
- Future Work

Datasets

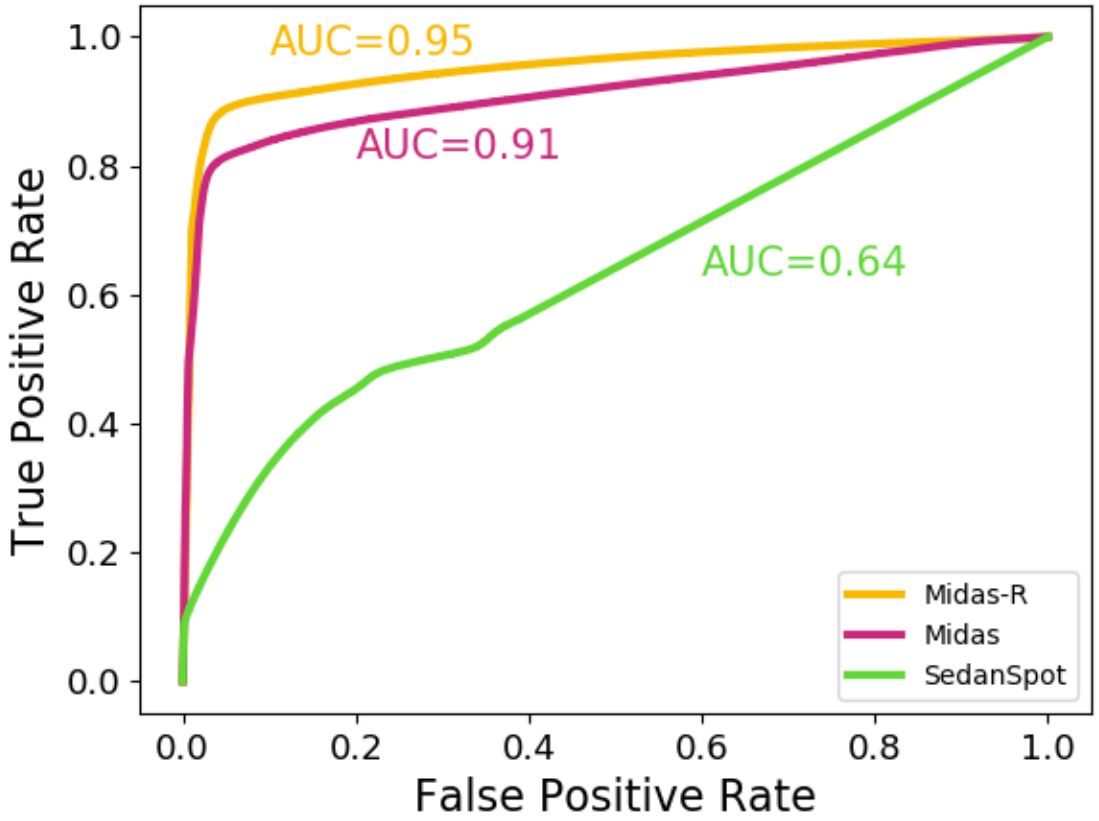




1. *DARPA*: 4.5M IP-IP communications, 87.7K minutes

2. TwitterSecurity: 2.6M tweets (May-August, 2014)
3. TwitterWorldCup: 1.7M tweets (June-July, 2014)

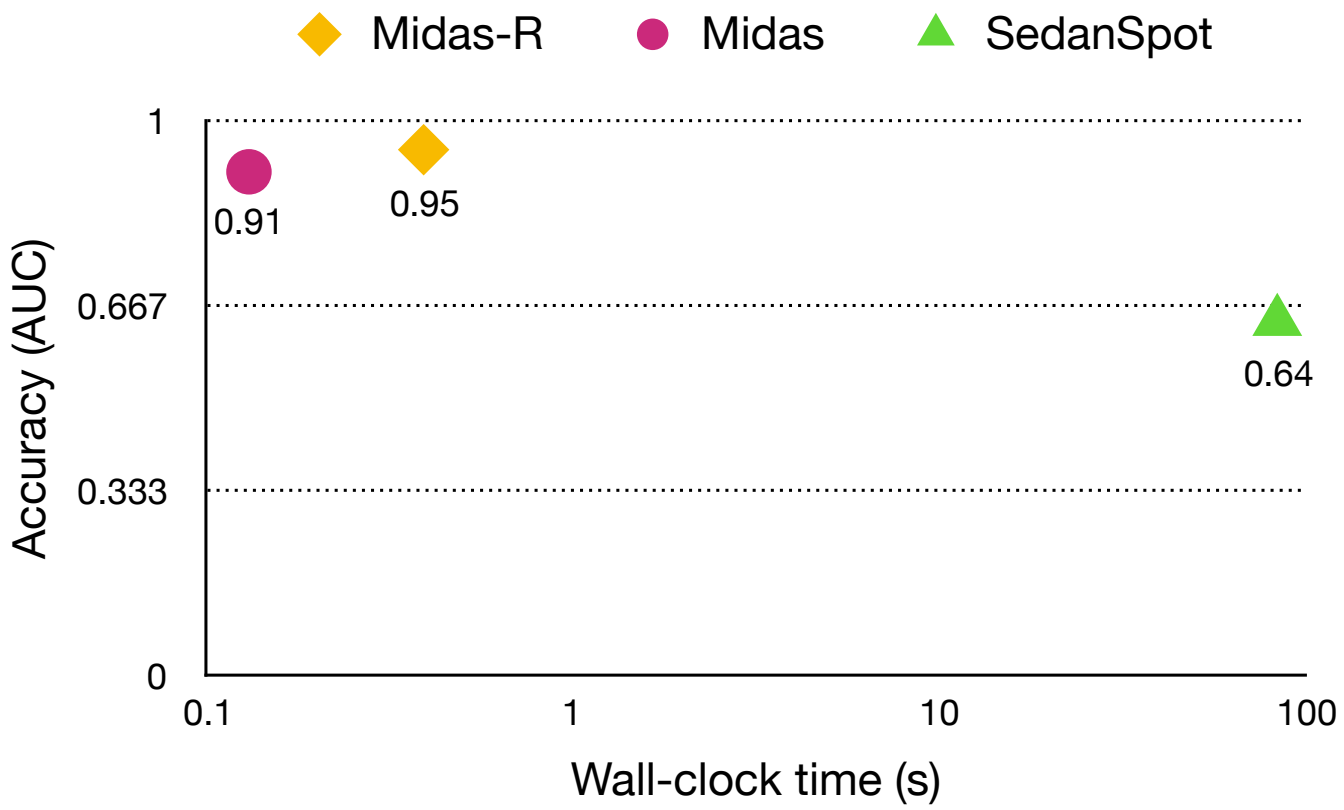
Accuracy


AUC=0.95
AUC=0.91
AUC=0.64
Midas-R
Midas
SedanSpot
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

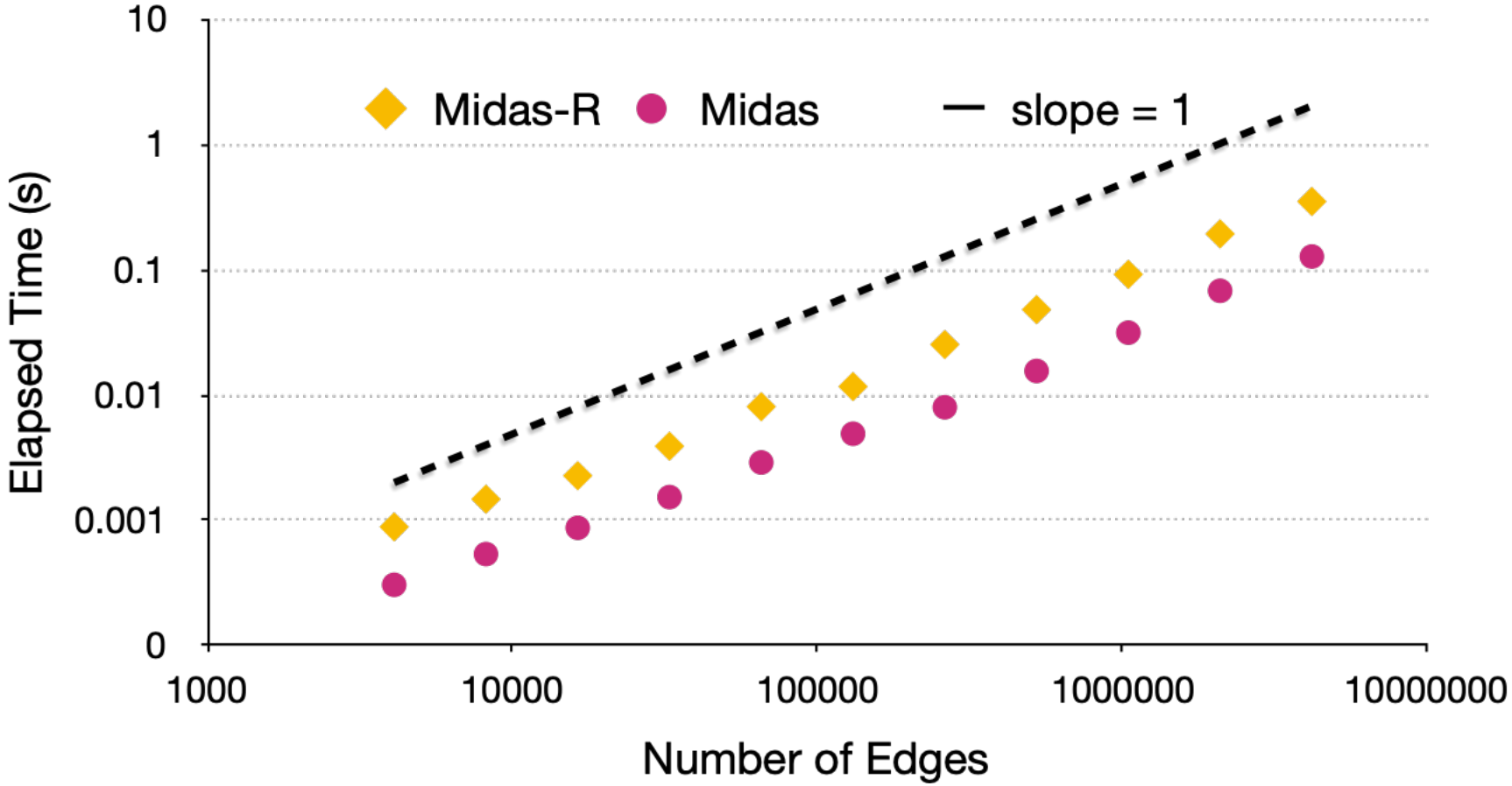
Running Times

	SedanSpot	Midas	Midas-R
TwitterWorldCup	27.58s	0.06s	0.17s
TwitterSecurity	40.71s	0.08s	0.23s
DARPA	83.66s	0.13s	0.39s

Accuracy vs Time



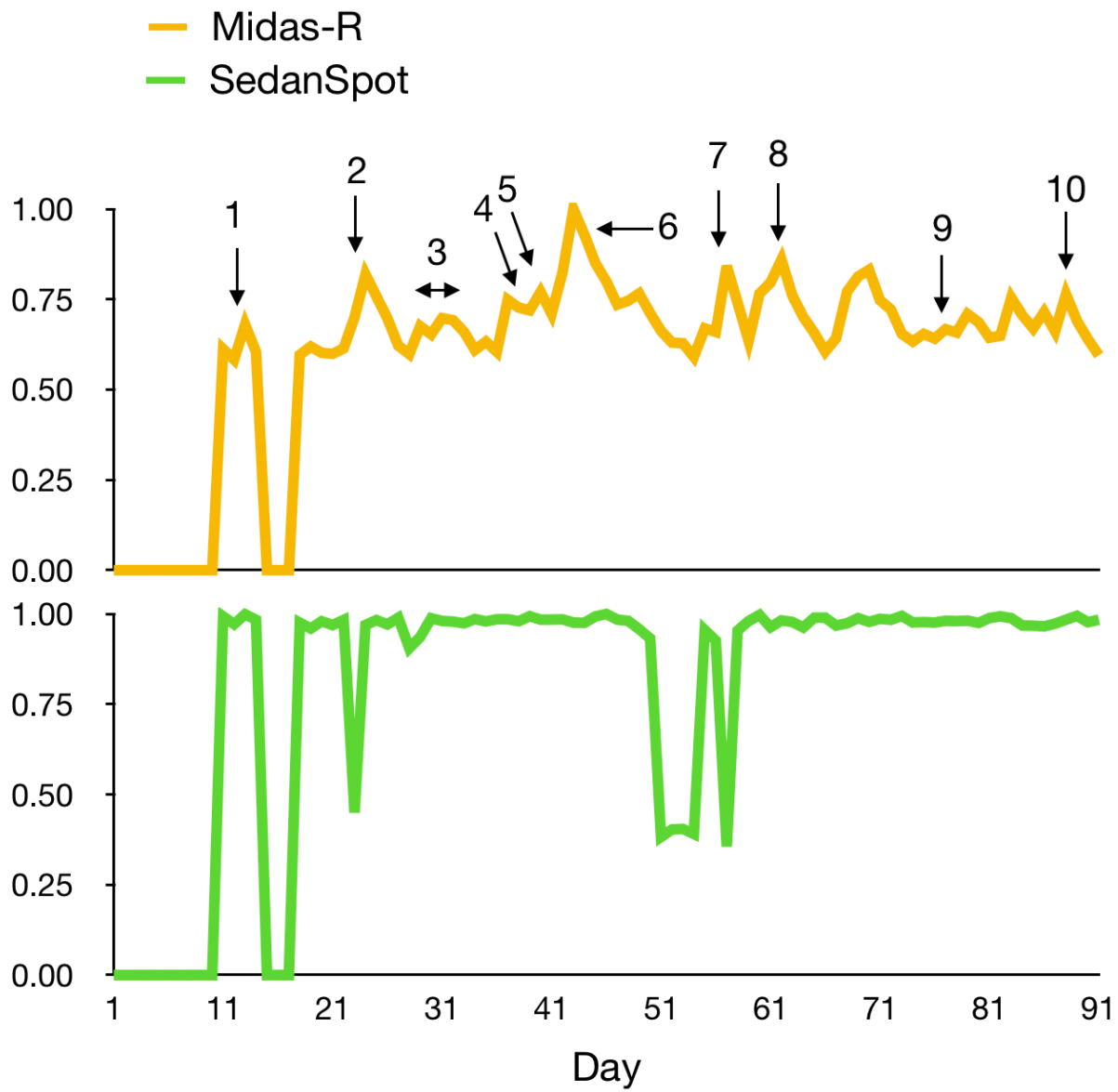
Scalability



Real-World Effectiveness

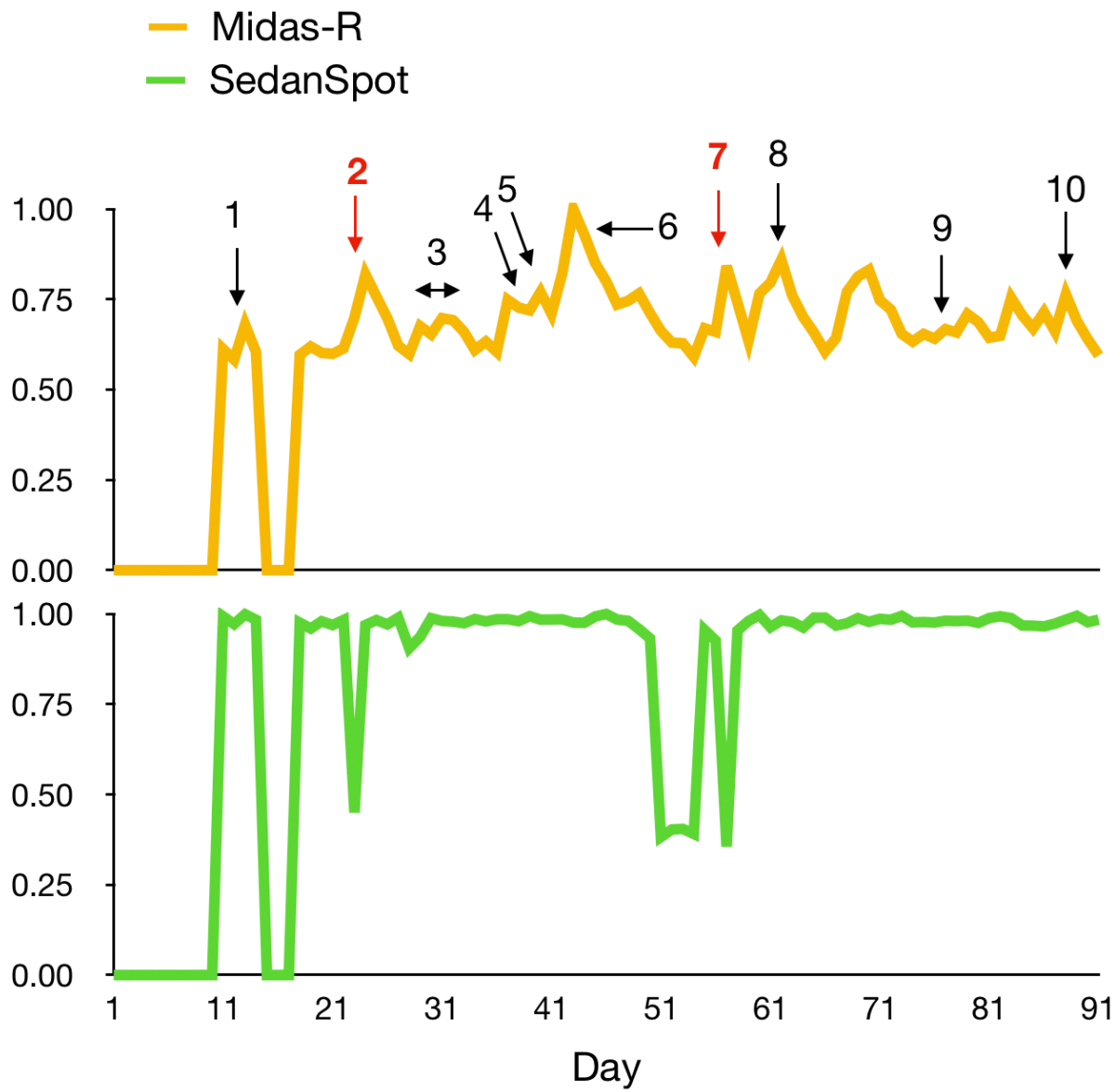
1. 13-05-2014. Turkey Mine Accident
2. 24-05-2014. Raid
3. 30-05-2014. Attack/Ambush
   03-06-14. Suicide bombing
4. 09-06-14. Suicide/Truck bombings
5. 10-06-2014. Iraqi Militants Seize
   11-06-2014. Kidnapping
6. 15-06-14. Mpeketoni attack
7. 26-06-14. Suicide Bombing
8. 03-07-14. Gaza conflicts
9. 18-07-14. Airplane shot
10. 30-07-14. Ebola Virus Outbreak

Real-World Effectiveness



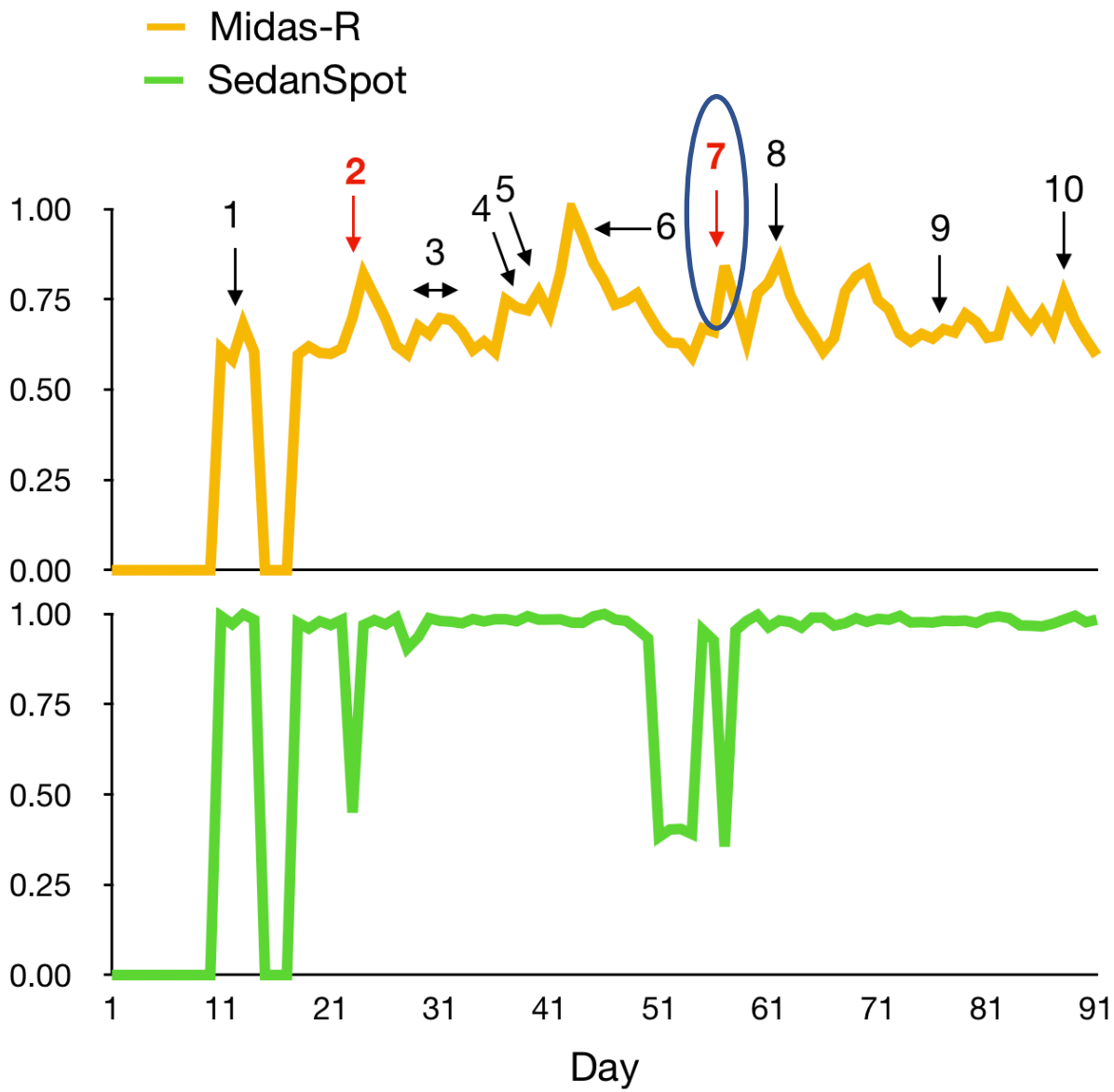
1. 13-05-2014. Turkey Mine Accident
2. 24-05-2014. Raid
3. 30-05-2014. Attack/Ambush
   03-06-14. Suicide bombing
4. 09-06-14. Suicide/Truck bombings
5. 10-06-2014. Iraqi Militants Seize
   11-06-2014. Kidnapping
6. 15-06-14. Mpeketoni attack
7. 26-06-14. Suicide Bombing
8. 03-07-14. Gaza conflicts
9. 18-07-14. Airplane shot
10. 30-07-14. Ebola Virus Outbreak

Real-World Effectiveness



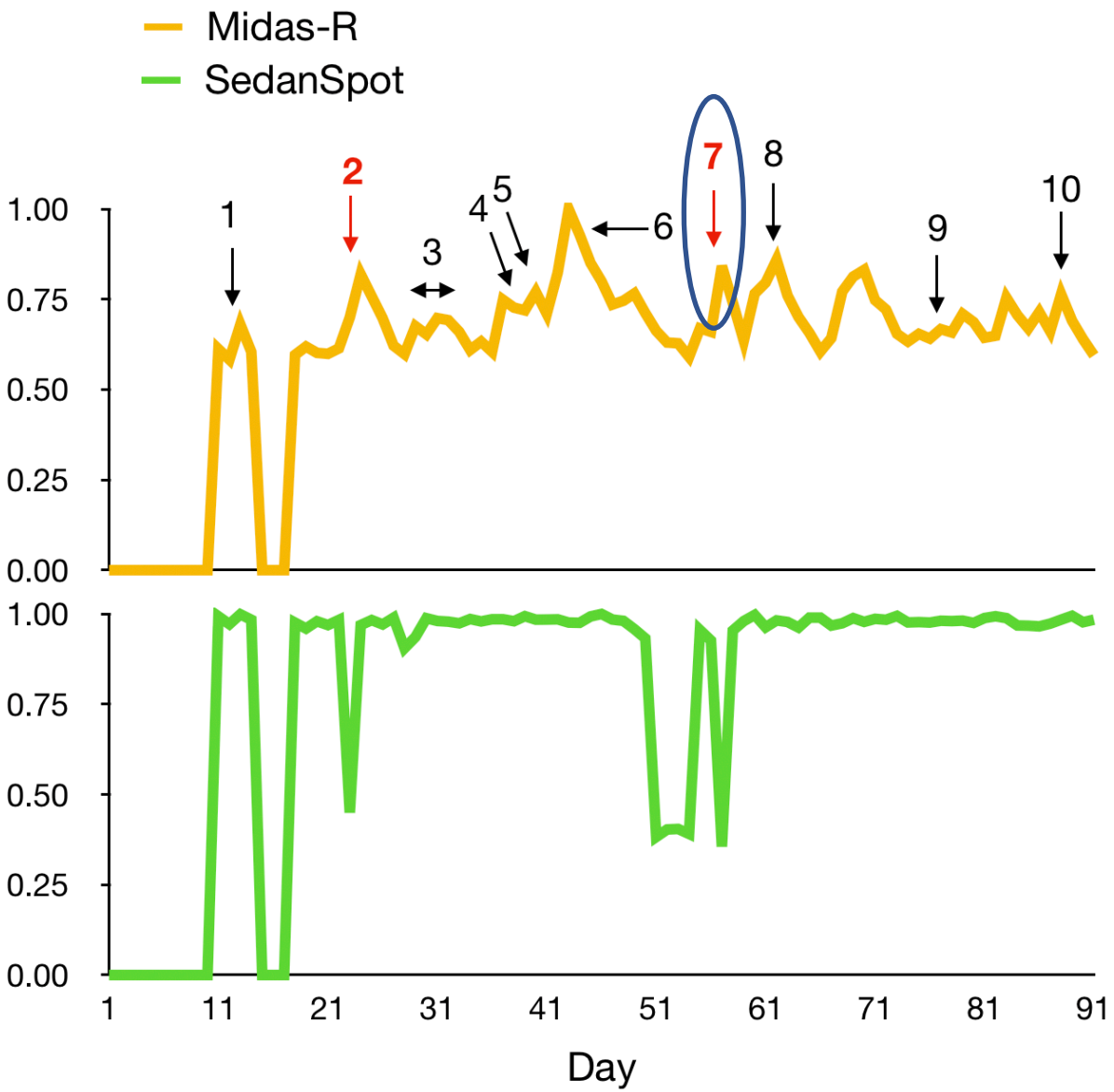
1. 13-05-2014. Turkey Mine Accident
2. **24-05-2014. Raid**
3. 30-05-2014. Attack/Ambush
   03-06-14. Suicide bombing
4. 09-06-14. Suicide/Truck bombings
5. 10-06-2014. Iraqi Militants Seize
   11-06-2014. Kidnapping
6. 15-06-14. Mpeketoni attack
7. **26-06-14. Suicide Bombing**
8. 03-07-14. Gaza conflicts
9. 18-07-14. Airplane shot
10. 30-07-14. Ebola Virus Outbreak

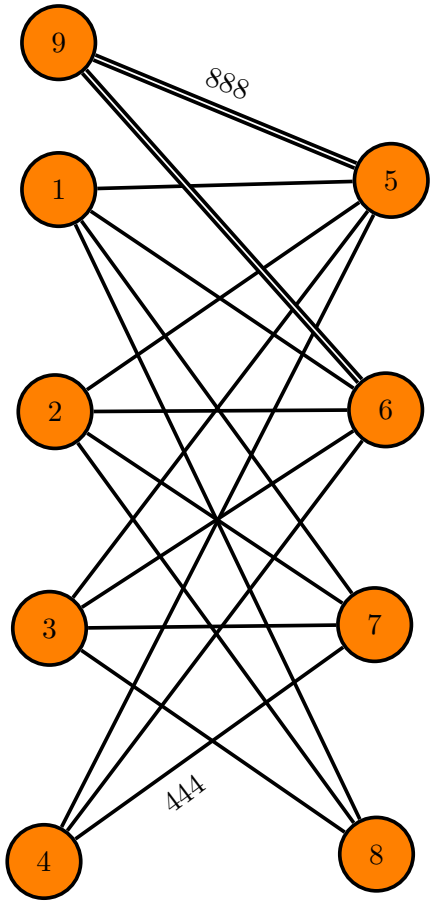
Real-World Effectiveness



1. 13-05-2014. Turkey Mine Accident
2. **24-05-2014. Raid**
3. 30-05-2014. Attack/Ambush
   03-06-14. Suicide bombing
4. 09-06-14. Suicide/Truck bombings
5. 10-06-2014. Iraqi Militants Seize
   11-06-2014. Kidnapping
6. 15-06-14. Mpeketoni attack
7. **26-06-14. Suicide Bombing**
8. 03-07-14. Gaza conflicts
9. 18-07-14. Airplane shot
10. 30-07-14. Ebola Virus Outbreak

Real-World Effectiveness





Roadmap

- Problem
- Algorithm
  - MIDAS
  - MIDAS-R
- Related Work
- Experiments
- **Future Work**

MSTREAM

Challenges:

- High number of dimensions
- Real-valued features
- Correlation between features
- Constant memory & time both w.r.t. stream length and dimensionality

Time	Source IP	Dest. IP	Pkt. Size	...
1	194.027.251.021	194.027.251.021	100	···
2	172.016.113.105	207.230.054.203	80	···
4	194.027.251.021	192.168.001.001	1000	···
4	194.027.251.021	192.168.001.001	995	···
4	194.027.251.021	192.168.001.001	1000	···
5	194.027.251.021	192.168.001.001	990	···
5	194.027.251.021	194.027.251.021	1000	···
5	194.027.251.021	194.027.251.021	995	···
6	194.027.251.021	194.027.251.021	100	···
7	172.016.113.105	207.230.054.203	80	···

How to Contribute?

1. MIDAS-F: Filtering MIDAS
2. Parallel/Distributed/GPU version
3. Hardware implementation over FPGA
4. Knowledge Graphs (NLP)
5. Periodic setting
6. Erlang, F# implementations

Conclusion

1. Streaming Microcluster Detection:
   - Constant time and memory

2. Theoretical Guarantees:
   - False Positive Probability

3. Effectiveness:
   - MIDAS is 42%-48% more accurate
   - MIDAS processes the data 162×-644× faster

https://github.com/Stream-AD/MIDAS/

Siddharth Bhatia, Bryan Hooi, Minji Yoon, Kijung Shin and Christos Faloutsos. “MIDAS: Microcluster-Based Detector of Anomalies in Edge Streams.” AAAI Conference on Artificial Intelligence (AAAI), 2020.
https://arxiv.org/abs/1911.04464