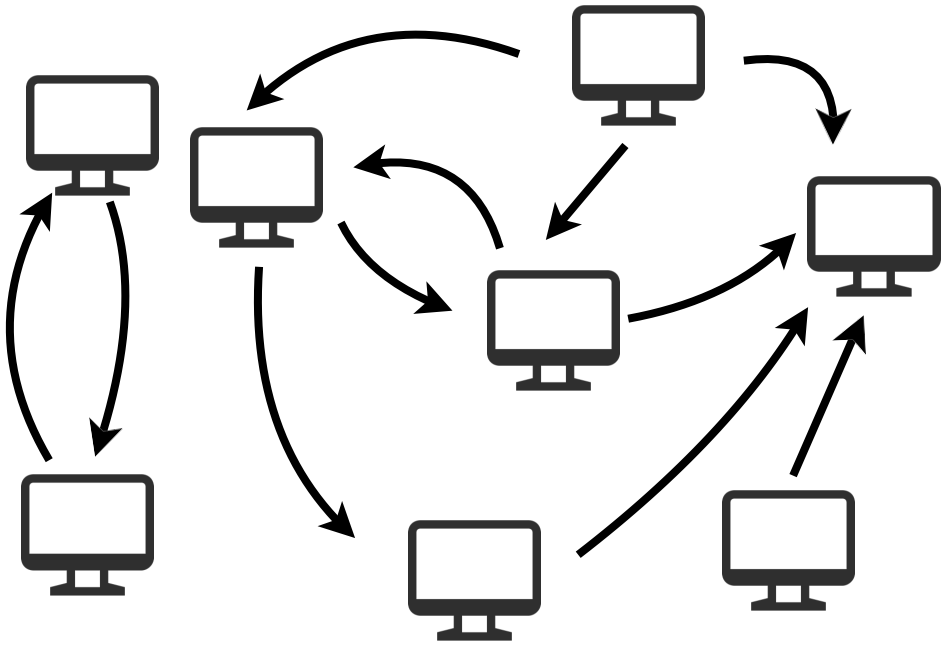
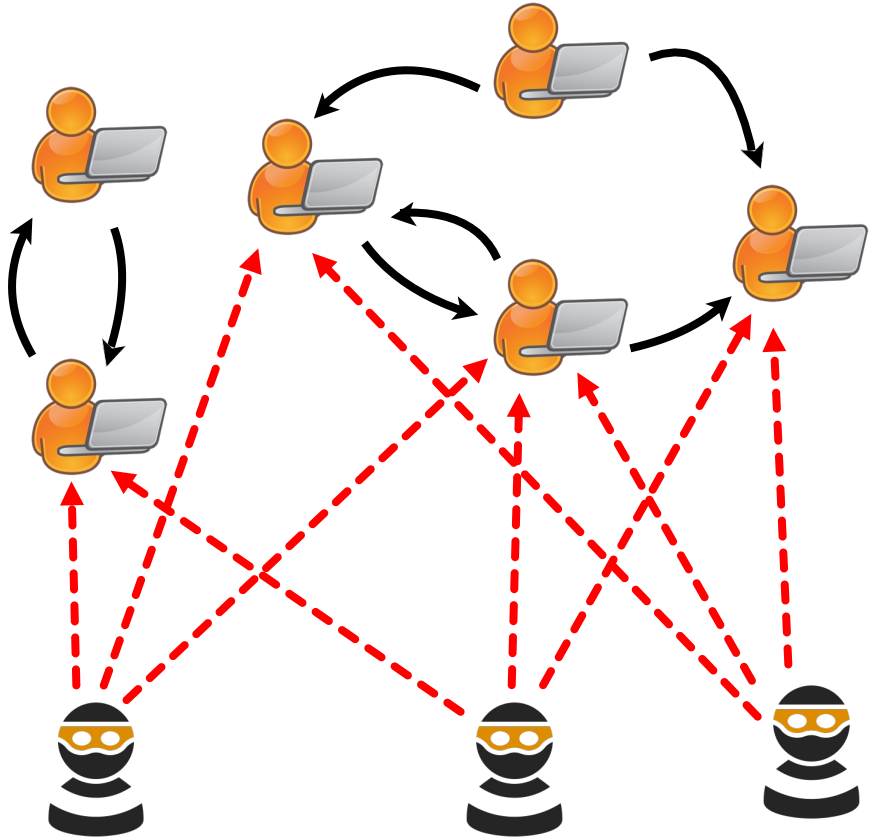
Streaming Anomaly Detection

SIDDHARTH BHATIA
siddharth@comp.nus.edu.sg

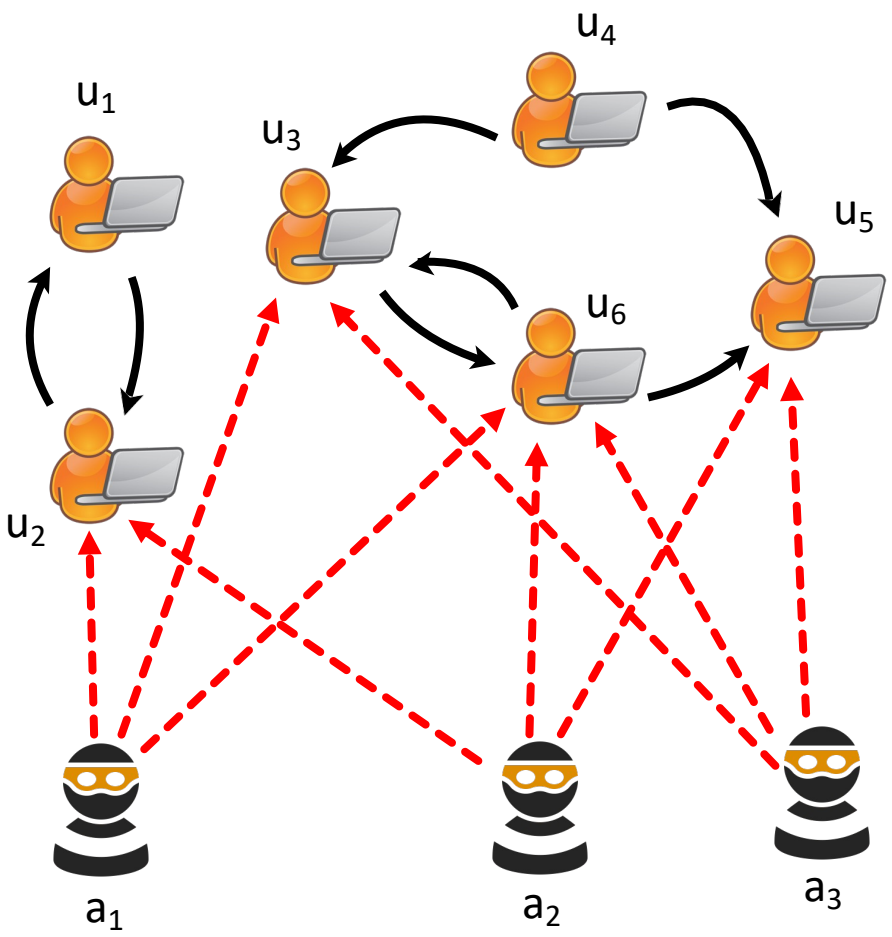
National University of Singapore

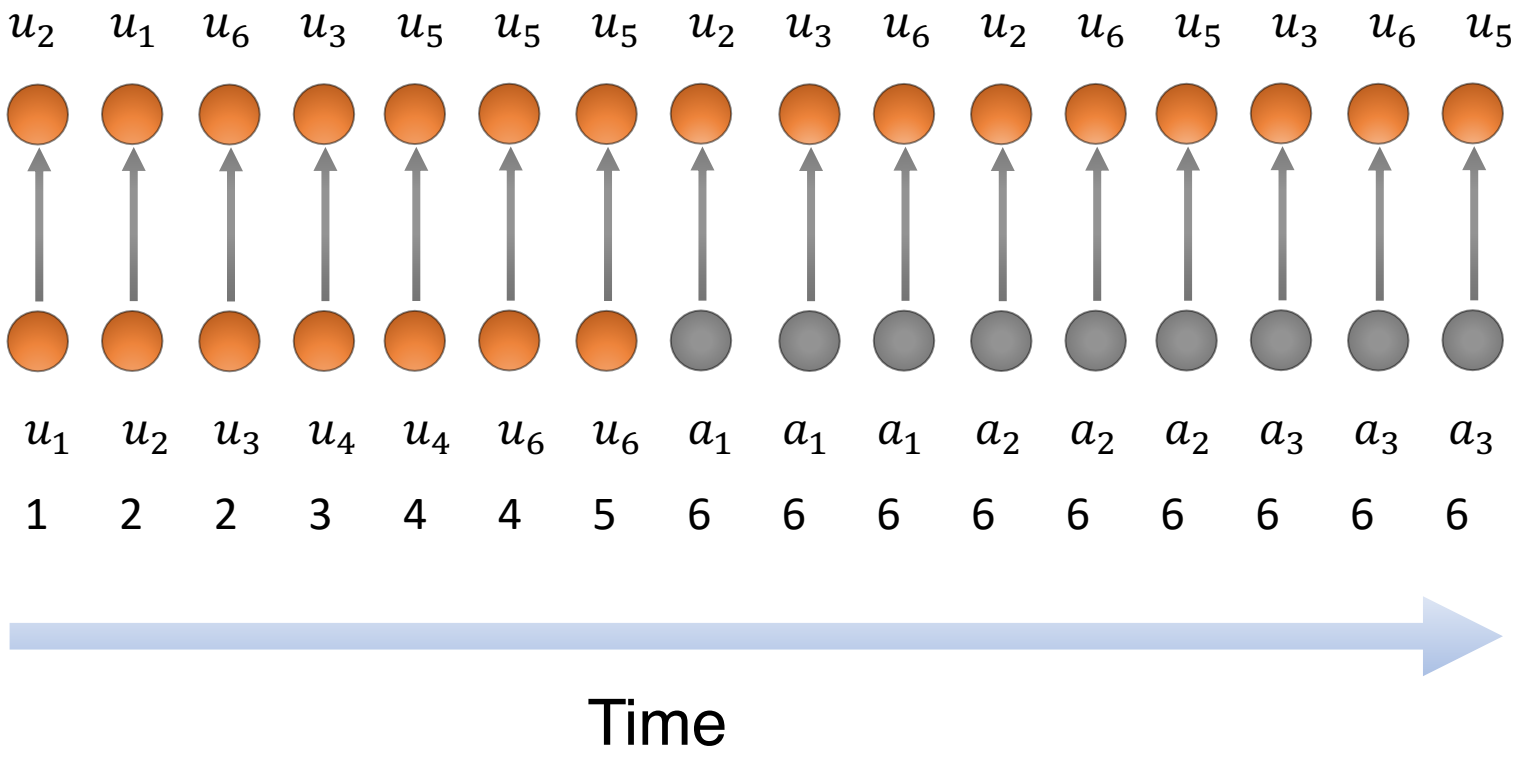
Motivation





Streaming Data





Research Topics

Setting	Anomaly Type	Method
Graph	Edges	MIDAS [AAAI20 & TKDD22]
Graph	Edges + Subgraphs	ANOEDGE/ANOGRAPH [Under Submission]
Multi-Aspect Data	Records	MSTREAM [WWW21]
Multi-Aspect Data	Records	MEMSTREAM [WWW22]

Roadmap

- Graphs
  - MIDAS
  - AnoEdge & AnoGraph
- Multi-Aspect Data
  - MStream
  - MemStream
- Conclusion

Roadmap

- Graphs
  - MIDAS
  - AnoEdge & AnoGraph
- Multi-Aspect Data
  - MStream
  - MemStream
- Conclusion

MIDAS

MIDAS: Microcluster-Based Detector of Anomalies in Edge Streams
Siddharth Bhatia, Bryan Hooi, Minji Yoon, Kijung Shin, Christos Faloutsos
AAAI, 2020

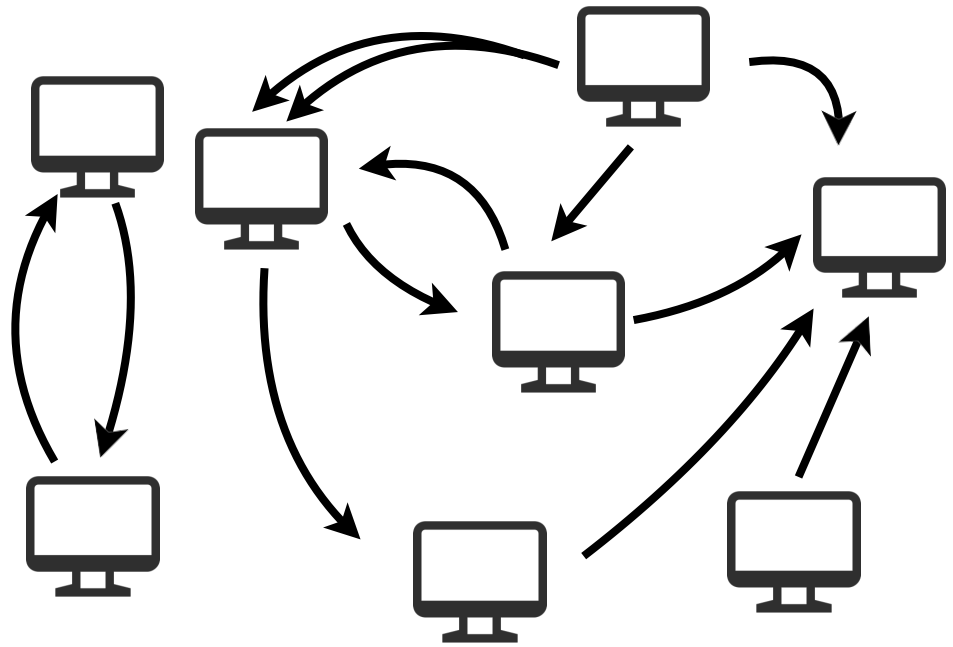
Real-Time Anomaly Detection in Edge Streams
Siddharth Bhatia, Rui Liu, Bryan Hooi, Minji Yoon, Kijung Shin, Christos Faloutsos
TKDD, 2022

MIDAS

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



Streaming Data Structure: CMS

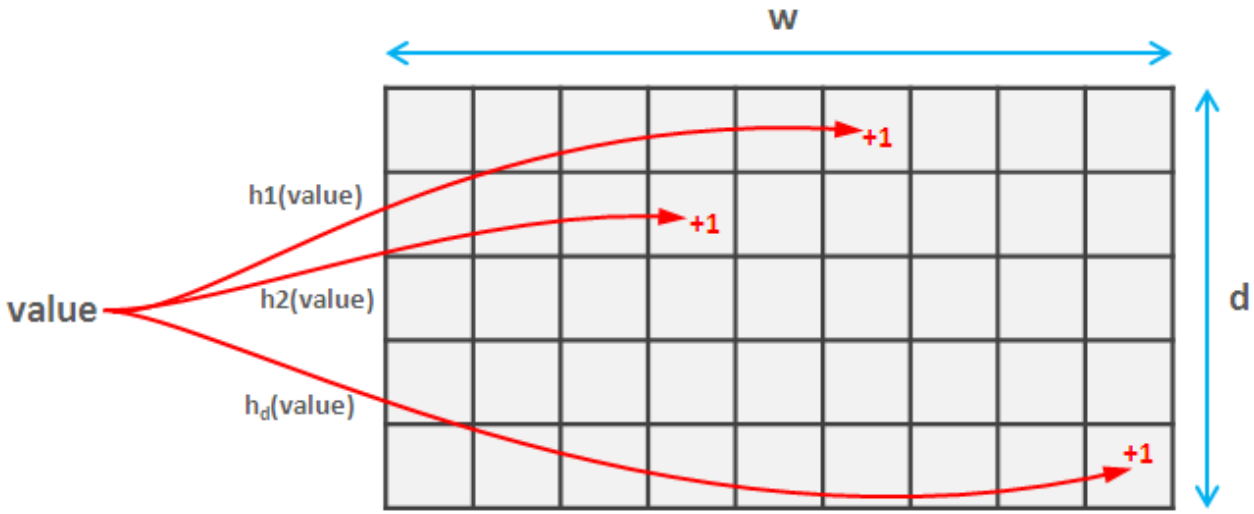
$\hat{a}_{uv} \leq auv + \upsilon N_t$ with probability at least $1 - \varepsilon$

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

$w = [e/\varepsilon]$ and $d = [\ln 1/\delta]$

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



Anomaly Score: Chi-Squared Test

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

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

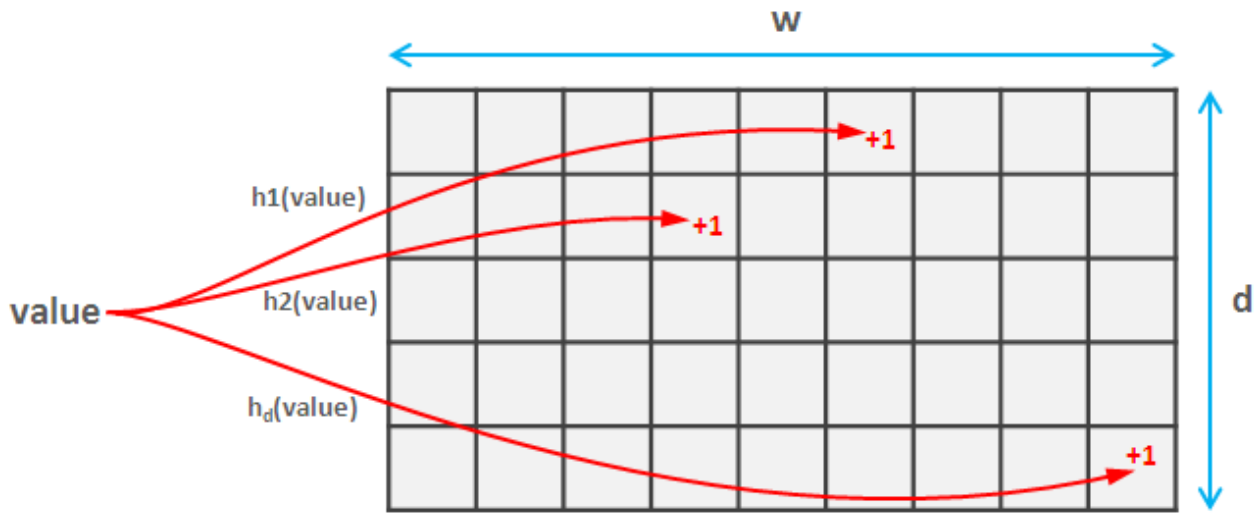
observed expected

Time and Memory Complexity

d: number of hash functions
w: number of buckets

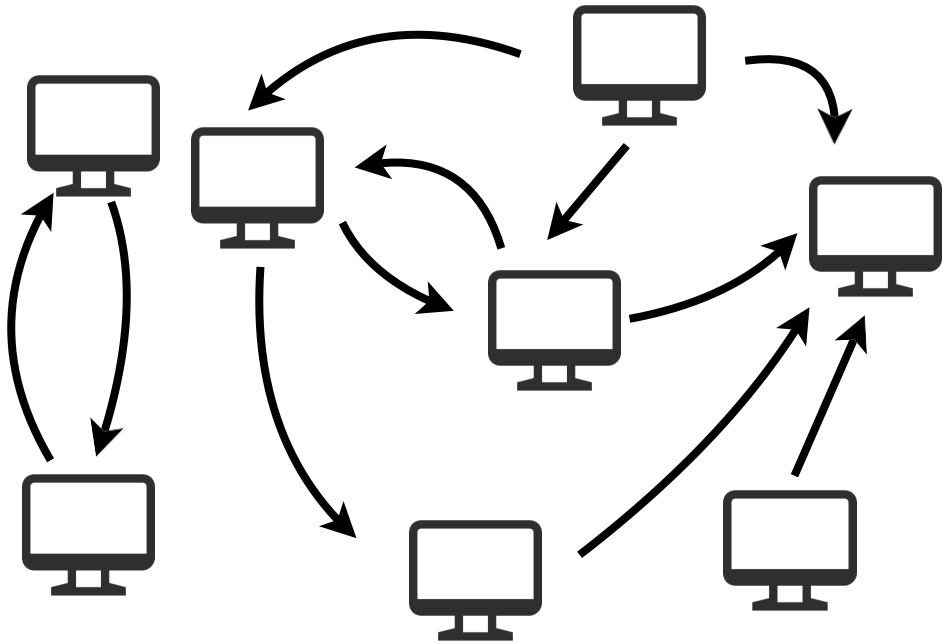
Space complexity:
- $O(wd)$

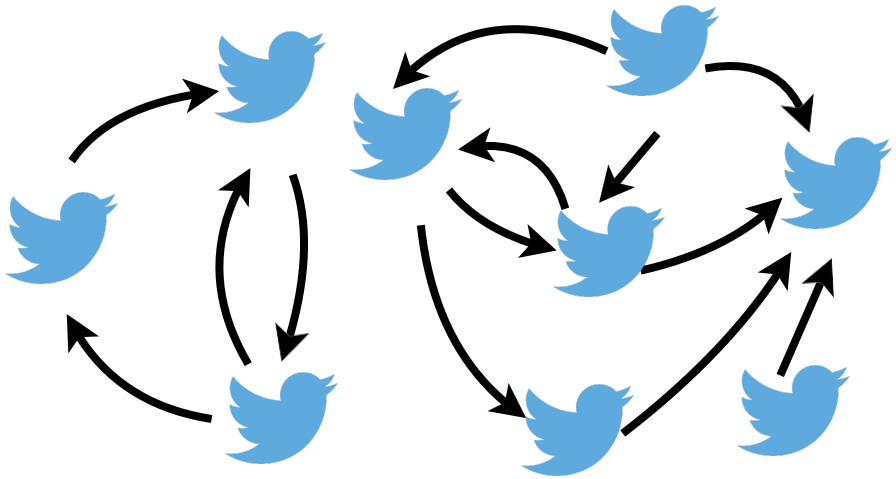
Time complexity:
- $O(d)$



EXPERIMENTS

Datasets

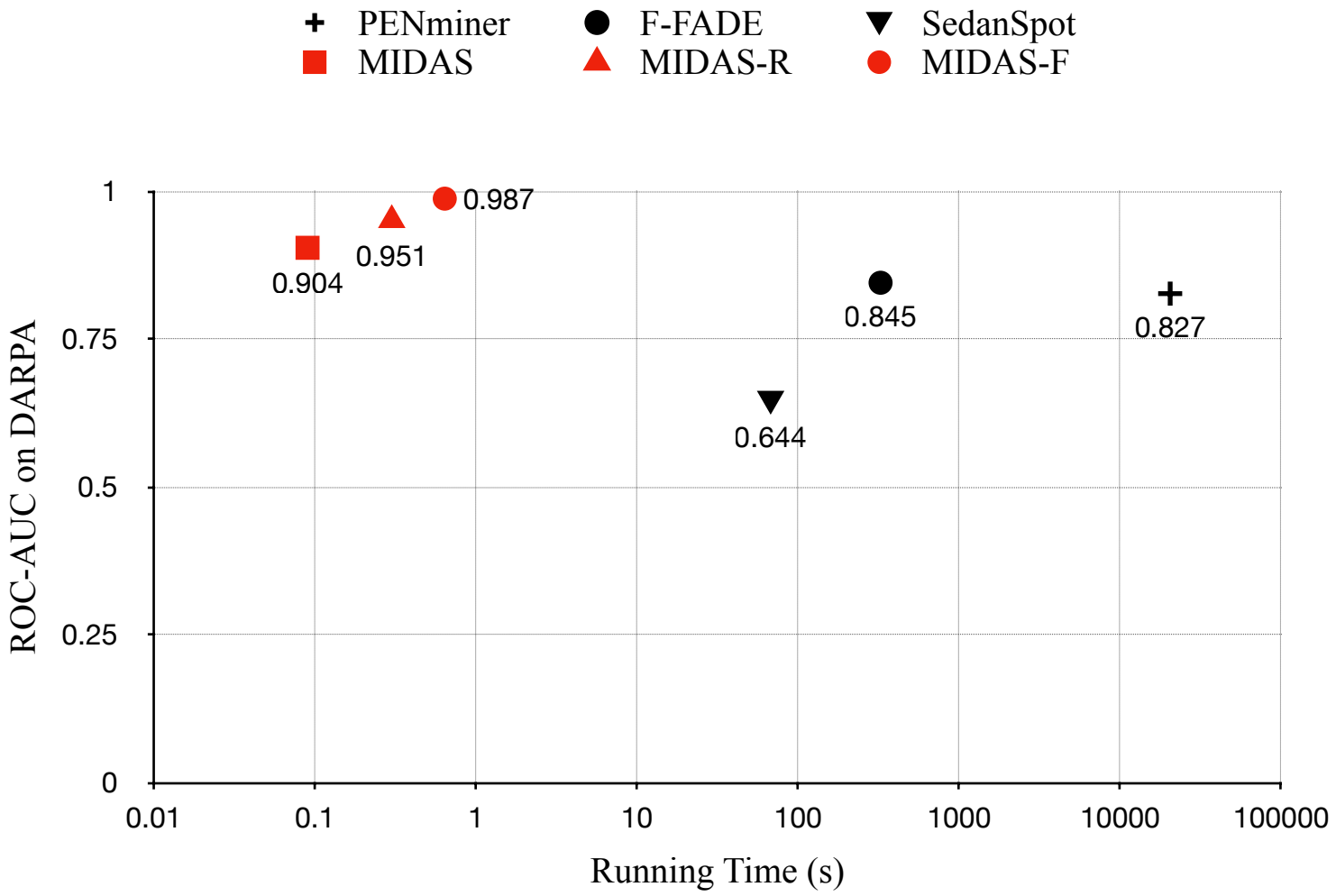




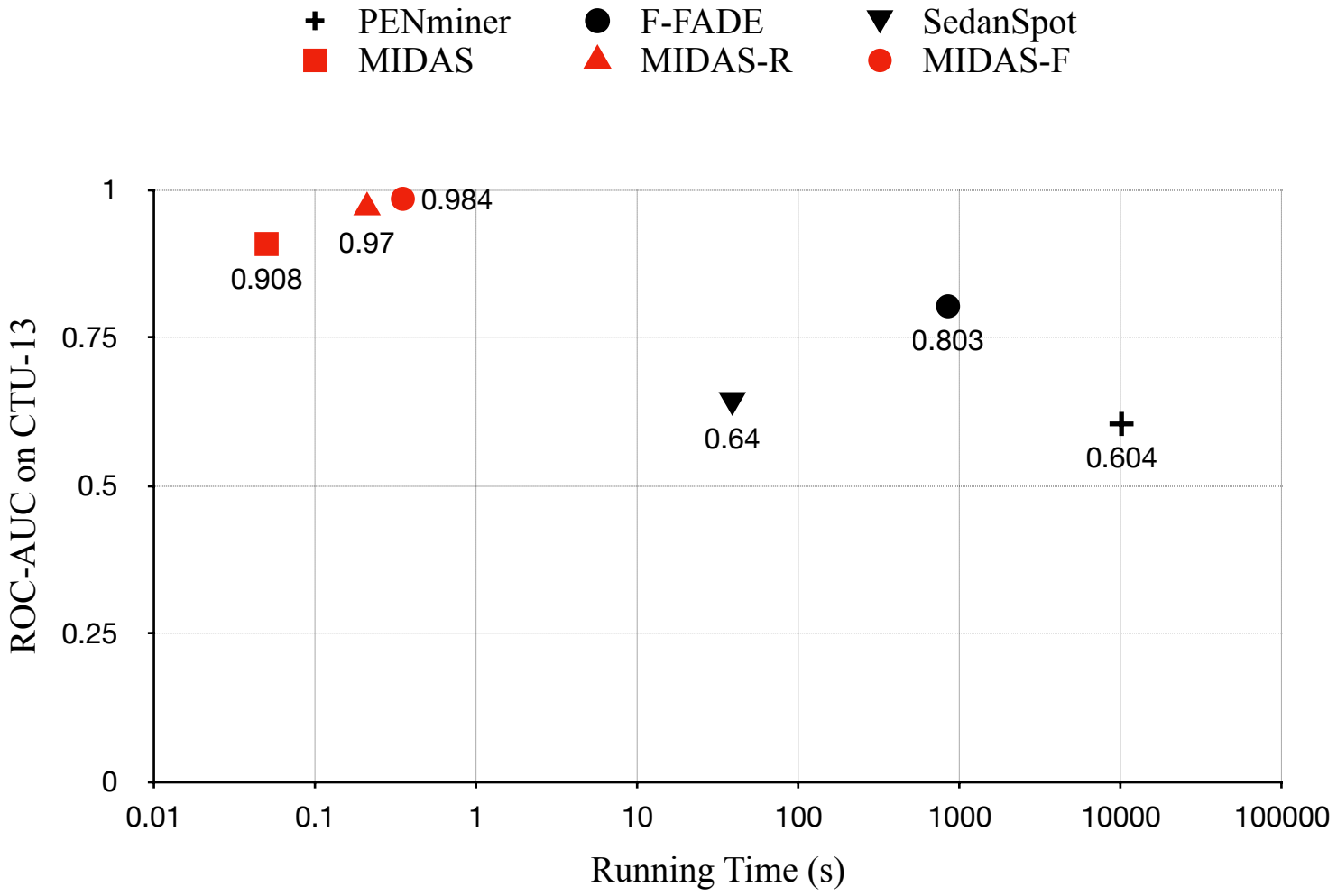
1. *DARPA*: 4.5M IP-IP communications, 46K timestamps
2. CTU-13: 2.5M IP-IP communications, 33K timestamps
3. UNSW-NB15: 2.5M IP-IP communications, 85K timestamps

4. *TwitterSecurity*: 2.6M tweets (May-August, 2014)
5. *TwitterWorldCup*: 1.7M tweets (June-July, 2014)

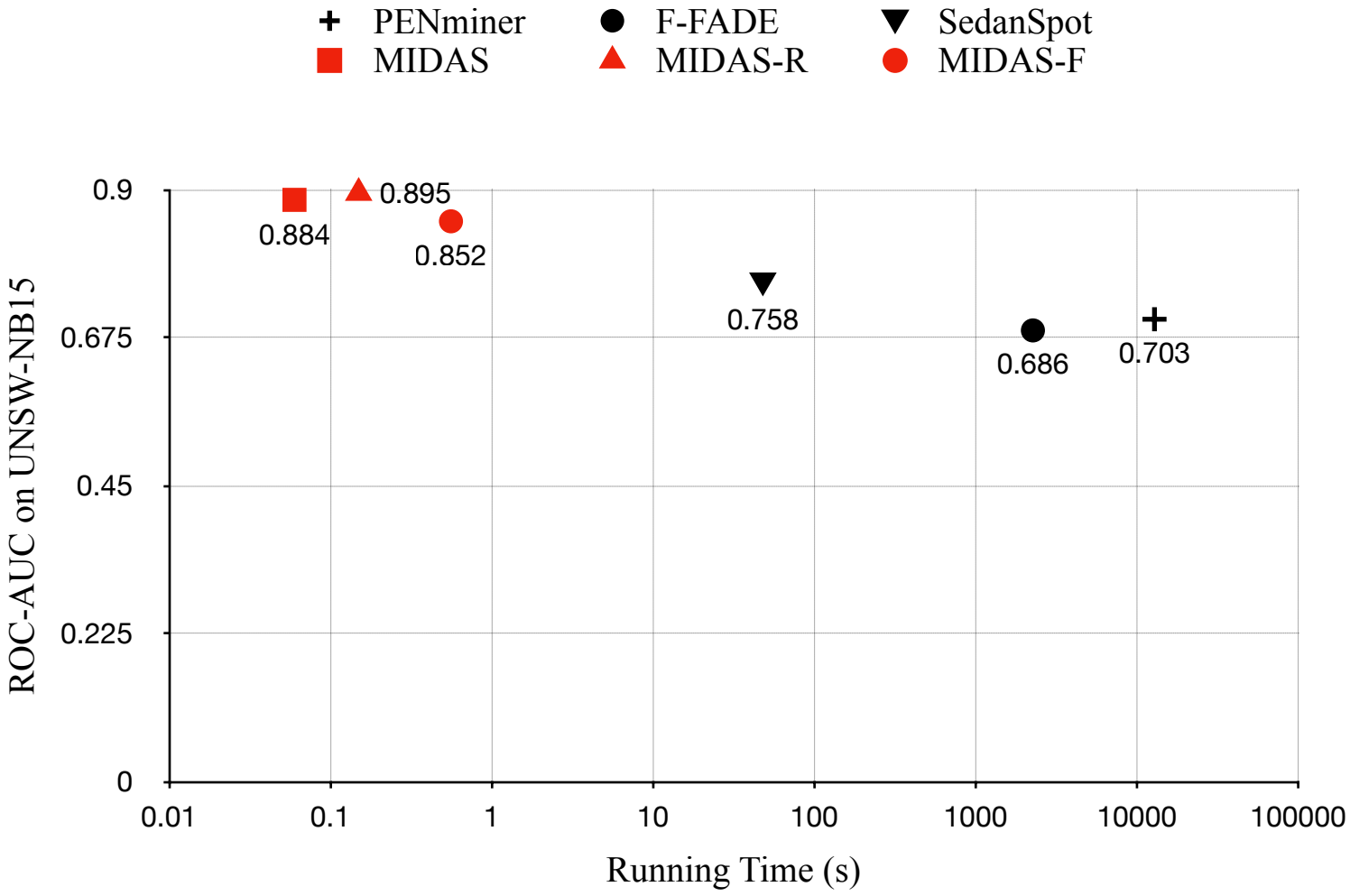
AUC vs time


PENminer
F-FADE
SedanSpot
MIDAS
MIDAS-R
MIDAS-F
ROC-AUC on DARPA
1
0.75
0.5
0.25
0
0.987
0.951
0.904
0.845
0.827
0.644
0.01
0.1
1
10
100
1000
10000
100000
Running Time (s)

AUC vs time


PENminer
F-FADE
SedanSpot
MIDAS
MIDAS-R
MIDAS-F
ROC-AUC on CTU-13
1
0.75
0.5
0.25
0
0.908
0.97
0.984
0.64
0.803
0.604
0.01
0.1
1
10
100
1000
10000
100000
Running Time (s)

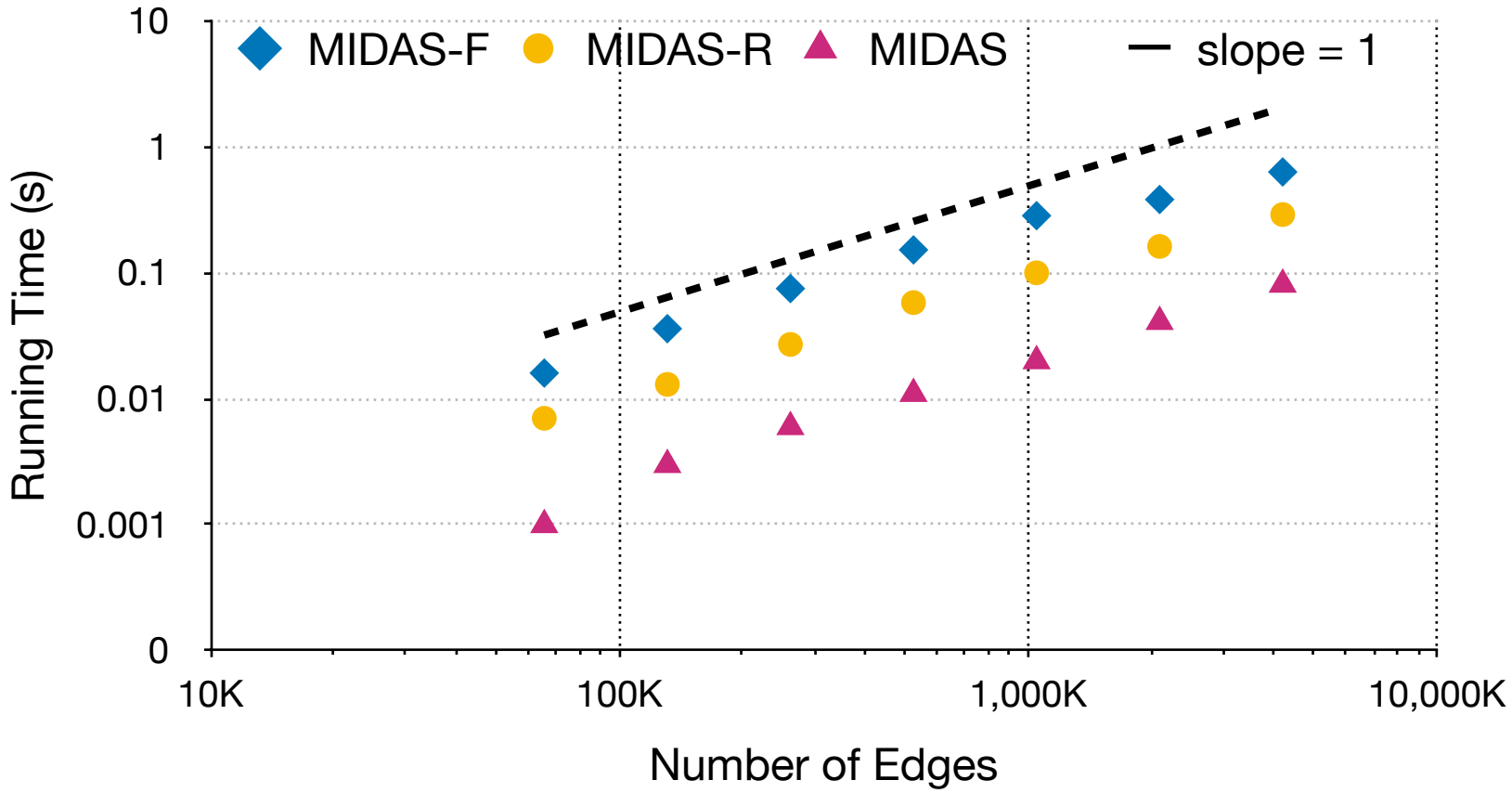
AUC vs time


PENminer
F-FADE
SedanSpot
MIDAS
MIDAS-R
MIDAS-F
0.9
0.675
0.45
0.225
0
ROC-AUC on UNSW-NB15
0.884
0.895
0.852
0.758
0.686
0.703
0.01
0.1
1
10
100
1000
10000
100000
Running Time (s)

Running Times

Dataset	PENminer	F-FADE	SedanSpot	Midas	Midas-R	Midas-F
DARPA	20423s	325.1s	67.54s	0.09s	0.30s	0.64s
CTU-13	10065s	844.2s	38.73s	0.05s	0.21s	0.35s
UNSW-NB15	12857s	2267s	48.03s	0.06s	0.15s	0.56s
TwitterWorldCup	3786s	141.7s	22.92s	0.03s	0.07s	0.08s
TwitterSecurity	5071s	40.34s	31.18s	0.05s	0.11s	0.11s

Scalability



Roadmap

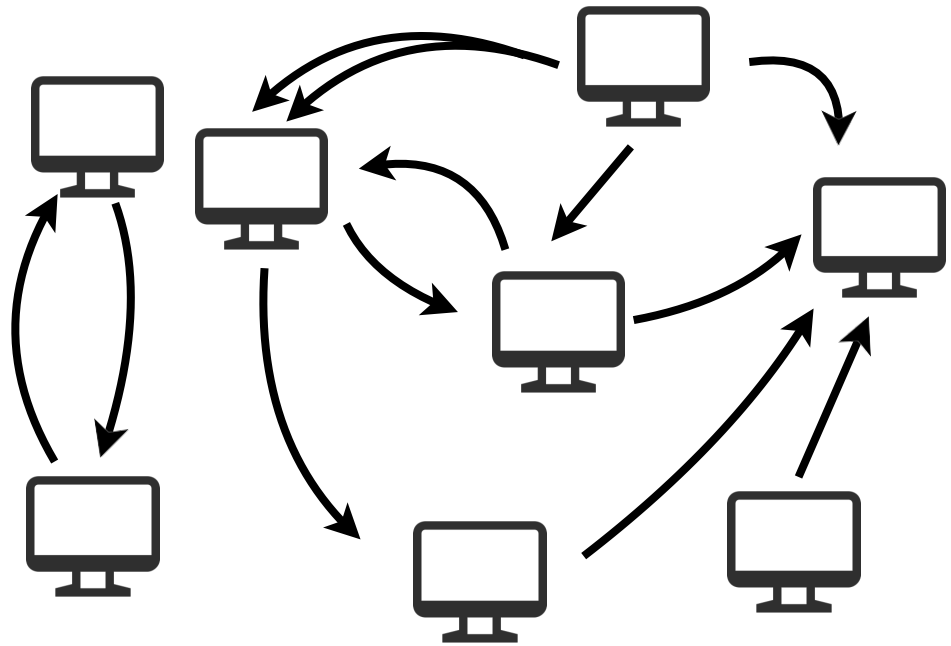
- Graphs
  - MIDAS
  - AnoEdge & AnoGraph
- Multi-Aspect Data
  - MStream
  - MemStream
- Conclusion

AnoEdge & AnoGraph

Higher-Order Sketch-Based Anomaly Detection in Dynamic Graphs
Siddharth Bhatia, Mohit Wadhwa, Kenji Kawaguchi, Neil Shah, Philip S. Yu, Bryan Hooi
[Under Submission]

AnoEdge & AnoGraph

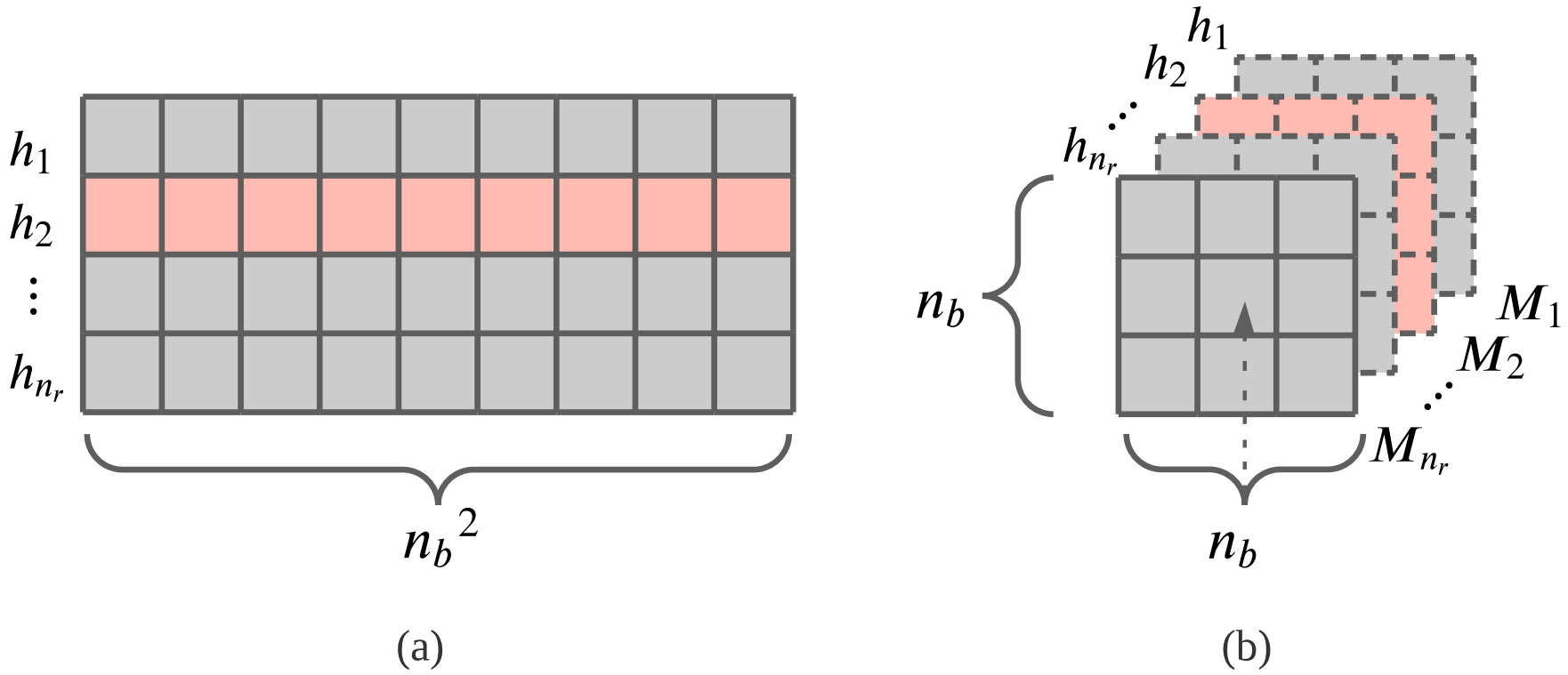
Input:
- Edge stream $\boldsymbol{E}$ from time evolving graph $\boldsymbol{G}$
- Directed, multigraph, discrete time



Output:
- Anomaly Score for each edge
- Anomaly Score for each subgraph

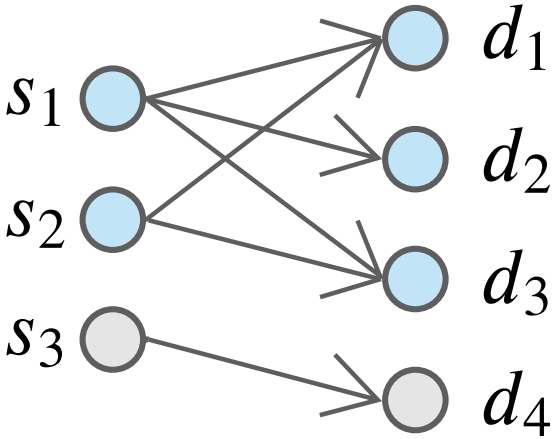
Our Contributions:
- Higher-Order sketch
- Streaming Anomaly Detection
- Incorporates dense subgraph search to detect graph anomalies in constant memory/time

CMS → Higher-order Sketch



(a) Original CMS with $n_b{}^2$ buckets for each hash function
(b) Higher-order CMS with n_b x n_b buckets for each hash function

Dense Subgraph → Dense Submatrix



(a)

	c_1	c_2	c_3	c_4
r_1	1	1	1	0
r_2	1	0	1	0
r_3	0	0	0	1
r_4	0	0	0	0

(b)

AnoEdge

- Detect edge anomalies by checking whether the received edge when mapped to a sketch matrix element is part of a dense submatrix.
- **AnoEdge-G** finds a Global dense submatrix and performs well in practice.
- **AnoEdge-L** maintains and updates a Local dense submatrix around the matrix element and therefore has better time complexity.

AnoGraph

- Detect graph anomalies by first mapping the graph to a higher-order sketch, and then checking for a dense submatrix.
- First streaming algorithms that make use of dense subgraph search to detect graph anomalies in constant memory and time.
- **AnoGraph** greedily finds a dense submatrix with a 2-approximation guarantee on the density measure.
- **AnoGraph-K** greedily find a dense submatrix around K strategically picked matrix elements

EXPERIMENTS

Datasets

Dataset	$\|V\|$	$\|E\|$	$\|T\|$
DARPA	25,525	4,554,344	46,567
ISCX-IDS2012	30,917	1,097,070	165,043
CIC-IDS2018	33,176	7,948,748	38,478
CIC-DDoS2019	1,290	20,364,525	12,224

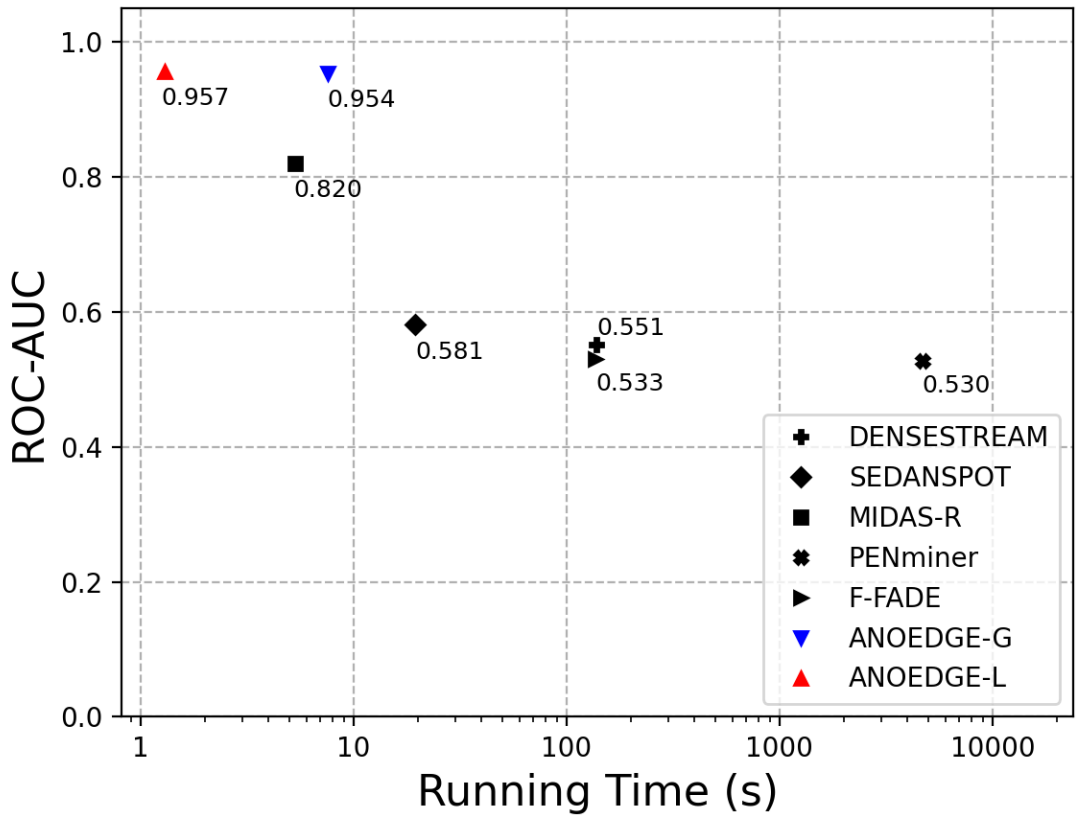
Anomalous Edges Baselines

DenseStream	SedanSpot	MIDAS-R	PENminer	F-FADE

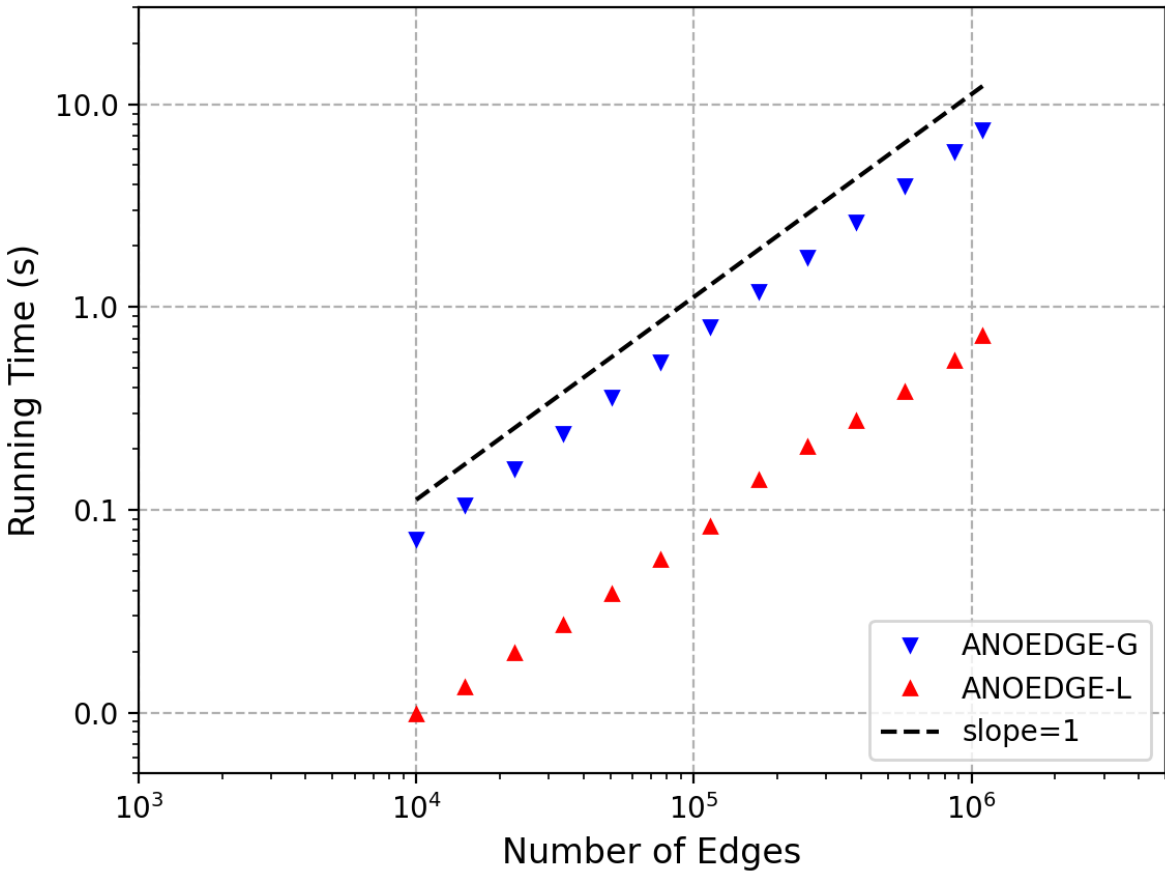
Anomalous Edges: AUC and Running Time

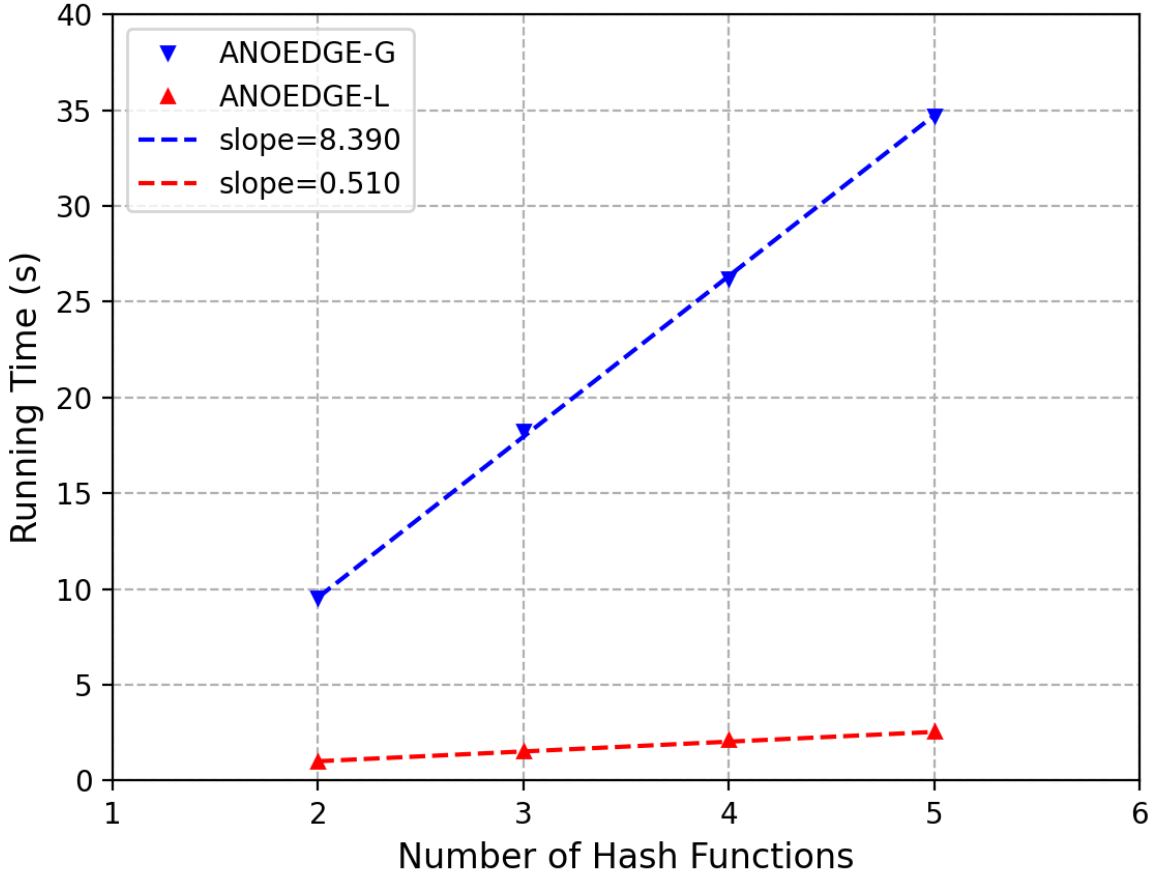
Dataset	DenseStream	SedanSpot	MIDAS-R	PENminer	F-FADE	AnoEdge-G	AnoEdge-L
DARPA	0.532 57.7s	0.647 ± 0.006 129.1s	0.953 ± 0.002 1.4s	0.872 5.21 hrs	0.919 ± 0.005 317.8s	**0.970 ± 0.001** 28.7s	0.964 ± 0.001 6.1s
ISCX-IDS2012	0.551 138.6s	0.581 ± 0.001 19.5s	0.820 ± 0.050 5.3s	0.530 1.3 hrs	0.533 ± 0.020 137.4s	0.954 ± 0.000 7.8s	**0.957 ± 0.003** 0.7s
CIC-IDS2018	0.756 3.3 hours	0.325 ± 0.037 209.6s	0.919 ± 0.019 1.1s	0.821 10 hrs	0.607 ± 0.001 279.7s	**0.963 ± 0.014** 58.4s	0.927 ± 0.035 10.2s
CIC-DDoS2019	0.263 265.6s	0.567 ± 0.004 697.6s	0.983 ± 0.003 2.2s	— > 24 hrs	0.717 ± 0.041 18.7s	0.997 ± 0.001 123.3s	**0.998 ± 0.001** 17.8s

Anomalous Edges: AUC vs Time



Anomalous Edges: Scalability





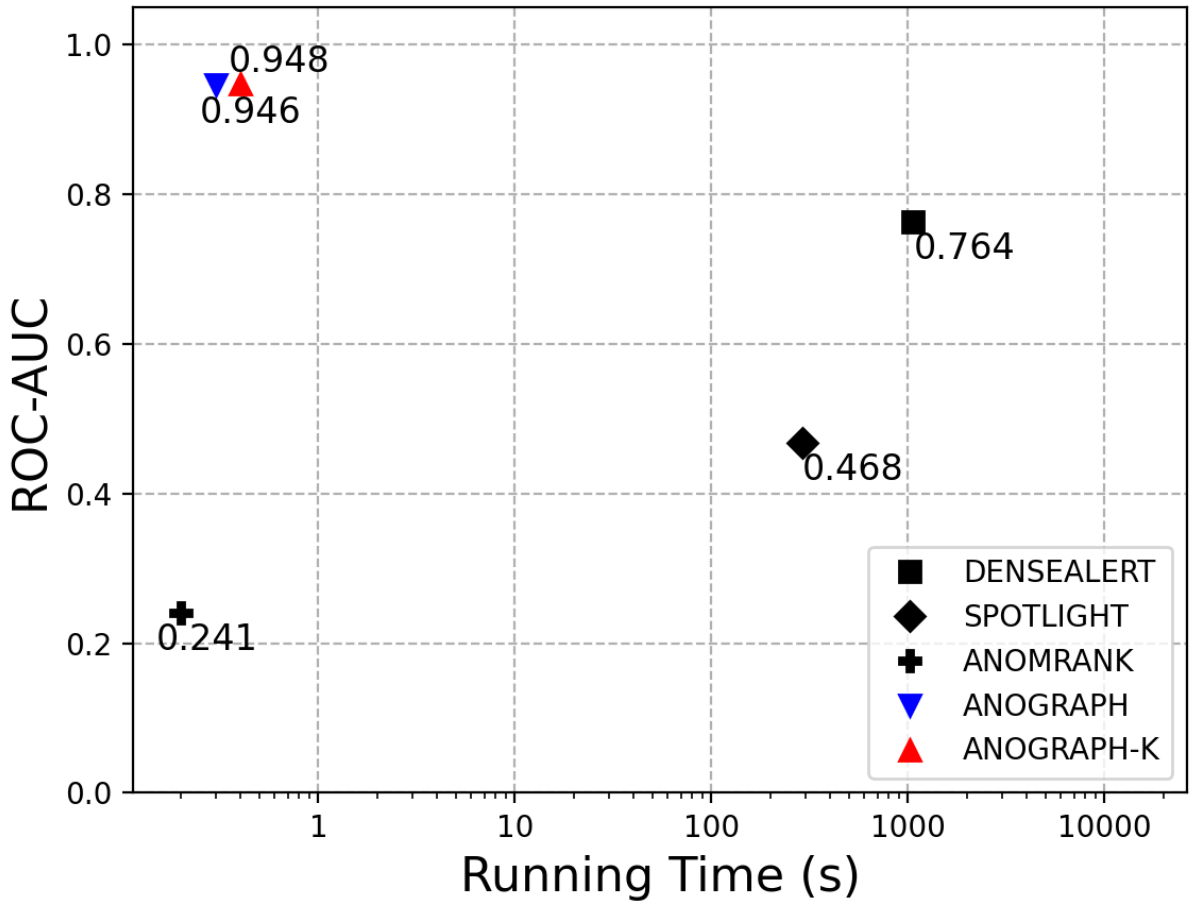
Anomalous Graphs Baselines

DenseAlert	SpotLight	AnomRank

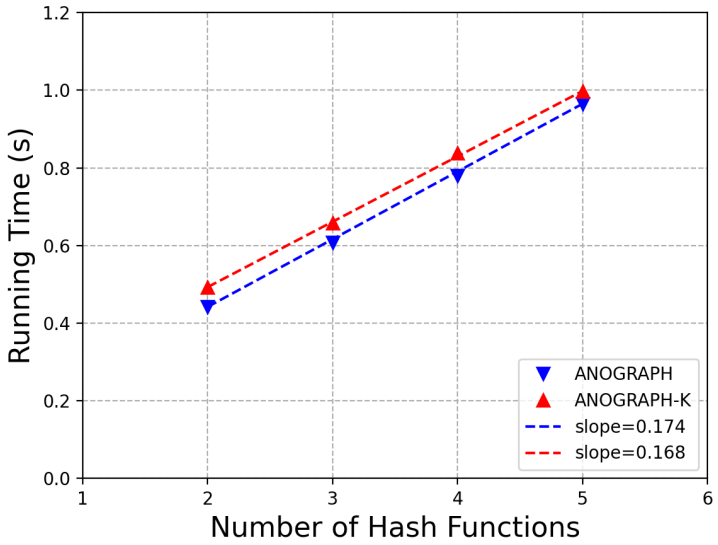
Anomalous Graphs: AUC and Running Time

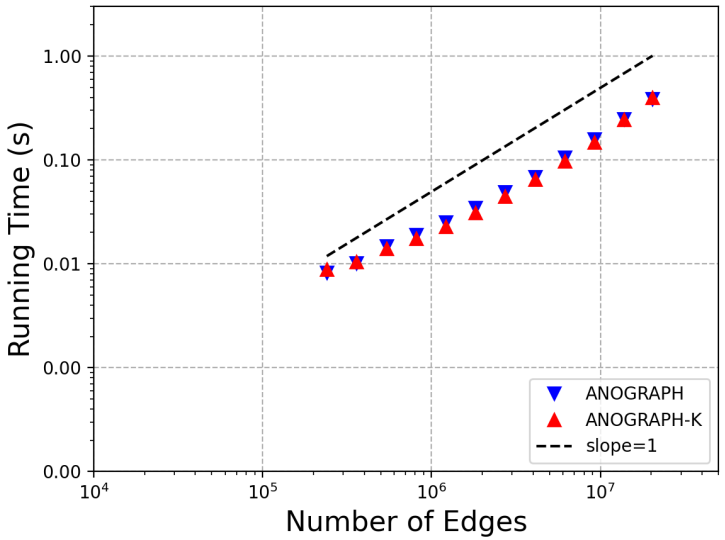
Dataset	DenseAlert	SpotLight	AnomRank	**AnoGraph**	**AnoGraph-K**
DARPA	0.833 49.3s	0.728 ± 0.016 88.5s	0.754 3.7s	0.835 ± 0.002 0.3s	**0.839 ± 0.002** 0.3s
ISCX-IDS2012	0.906 6.4s	0.872 ± 0.019 21.1s	0.194 5.2s	**0.950 ± 0.001** 0.5s	**0.950 ± 0.001** 0.5s
CIC-IDS2018	0.950 67.9s	0.835 ± 0.022 149.0s	0.783 7.0s	**0.957 ± 0.000** 0.2s	**0.957 ± 0.000** 0.3s
CIC-DDoS2019	0.764 1065.0s	0.468 ± 0.048 289.7s	0.241 0.2s	0.946 ± 0.002 0.4s	**0.948 ± 0.002** 0.4s

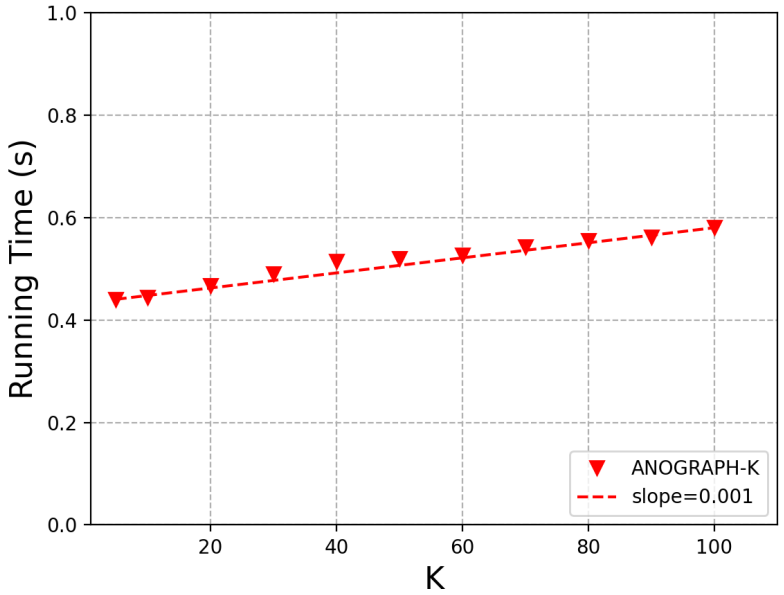
Anomalous Graphs: AUC vs Time



Anomalous Graphs: Scalability







Roadmap

- Graphs
  - MIDAS
  - AnoEdge & AnoGraph
- Multi-Aspect Data
  - MStream
  - MemStream
- Conclusion

Roadmap

- Graphs
  - MIDAS
  - AnoEdge & AnoGraph
- Multi-Aspect Data
  - MStream
  - MemStream
- Conclusion

MStream

MStream: Fast Anomaly Detection in Multi-Aspect Streams [Best Paper Finalist]
Siddharth Bhatia, Arjit Jain, Pan Li, Ritesh Kumar, Bryan Hooi
WWW, 2021

MStream

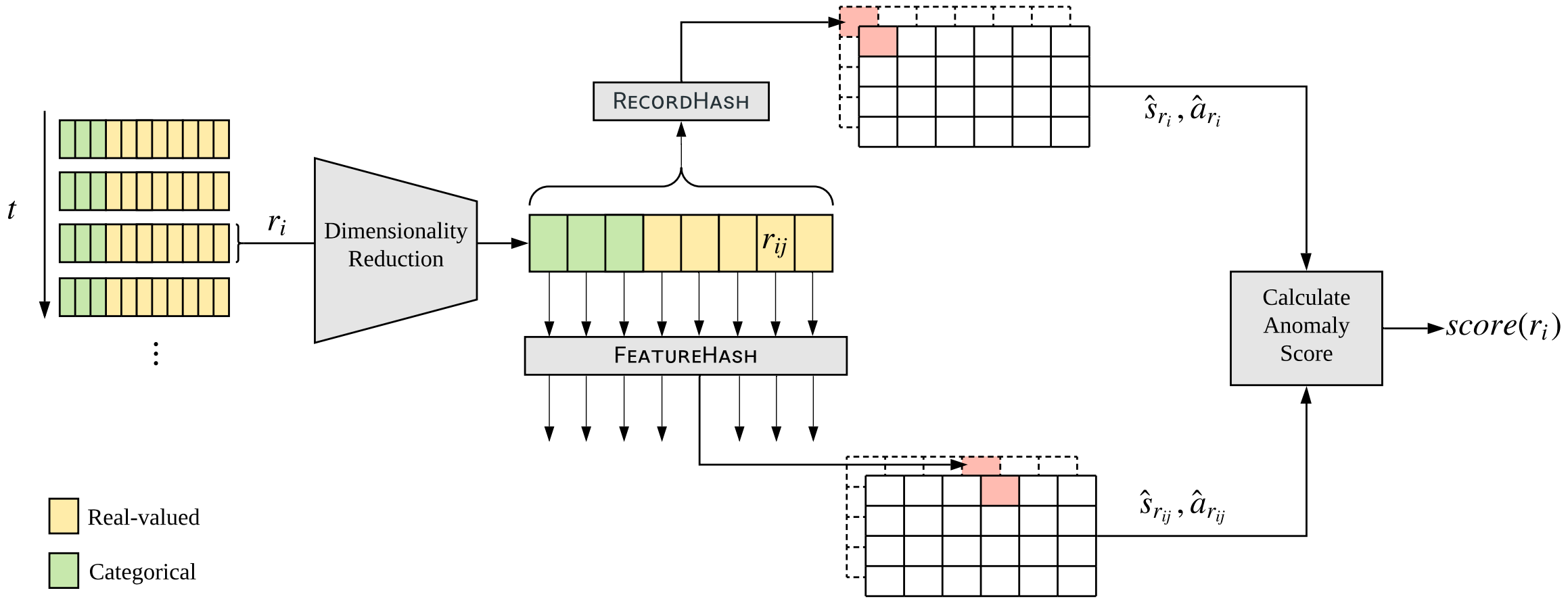
Input:
- Record stream $\boldsymbol{R}$
- Each having $\boldsymbol{d}$ dimensions

Output:
- Anomaly Score for each Record

Our Contributions:
- Multi-Aspect Group Anomaly Detection
- Streaming Approach
- Capture Correlation Between Features

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	···

Overview


t
r_i
Dimensionality Reduction
RecordHash
r_{ij}
FeatureHash
$\hat{s}_{r_i}, \hat{a}_{r_i}$
$\hat{s}_{r_{ij}}, \hat{a}_{r_{ij}}$
Calculate Anomaly Score
$score(r_i)$
Real-valued
Categorical

Feature Hash

Input: r_{ij} (Feature j of record r_i)
Output: Bucket index in $\{0, \ldots, b-1\}$ to map r_{ij} into

if r_{ij} is categorical
    output HASH(r_{ij}) // Linear Hash
else if r_{ij} is real-valued
    ▷ **Log-Transform**
        $\tilde{r}_{ij} = \log(1 + r_{ij})$
    ▷ **Normalize**
        $\tilde{r}_{ij} \leftarrow \frac{\tilde{r}_{ij} - min_j}{max_j - min_j}$ // Streaming Min-Max
    output $\lfloor \tilde{r}_{ij} \cdot b \rfloor \pmod b$ // Bucketization into b buckets

Record Hash

Input: Record r_i
Output: Bucket index in $\{0, \ldots, b-1\}$ to map r_i into

▷ **Divide r_i into its categorical part, r_i^{cat}, and its numerical part, r_i^{num}**
▷ **Hashing r_i^{cat}**
    $bucket_{cat} = (\sum_{j \in C} \text{HASH}(r_{ij})) \pmod b$ // Linear Hash
▷ **Hashing r_i^{num}**
    for $id \leftarrow 1$ to k
        if $\langle r_i^{num}, \mathbf{a_{id}} \rangle > 0$
            $bitset[id] = 1$
        else
            $bitset[id] = 0$
    $bucket_{num} = \text{INT}(bitset)$ // Convert bitset to integer
output $(bucket_{cat} + bucket_{num}) \pmod b$

Dimensionality Reduction

- Incorporates Correlations
- Faster processing

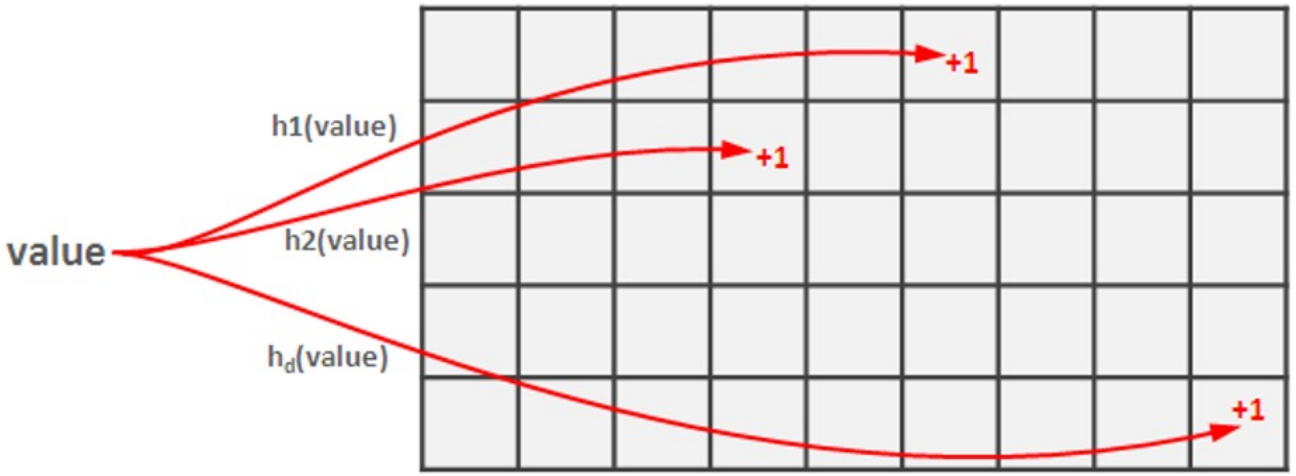
1. Principal Component Analysis
2. Information Bottleneck
3. Autoencoder

Time and Memory Complexity

w: number of hash functions
b: number of buckets
d: number of dimensions/features

Space complexity:
- $O(wbd)$

Time complexity:
- $O(wd)$



EXPERIMENTS

Datasets

1. *KDDCUP99*: 1.21M records (20% anomalies), 42 features
2. CICIDS-DoS: 1.05M records (5% anomalies), 80 features
3. UNSW-NB15: 2.5M records (13% anomalies), 49 features
4. CICIDS-DDoS: 7.9M records (7% anomalies), 83 features

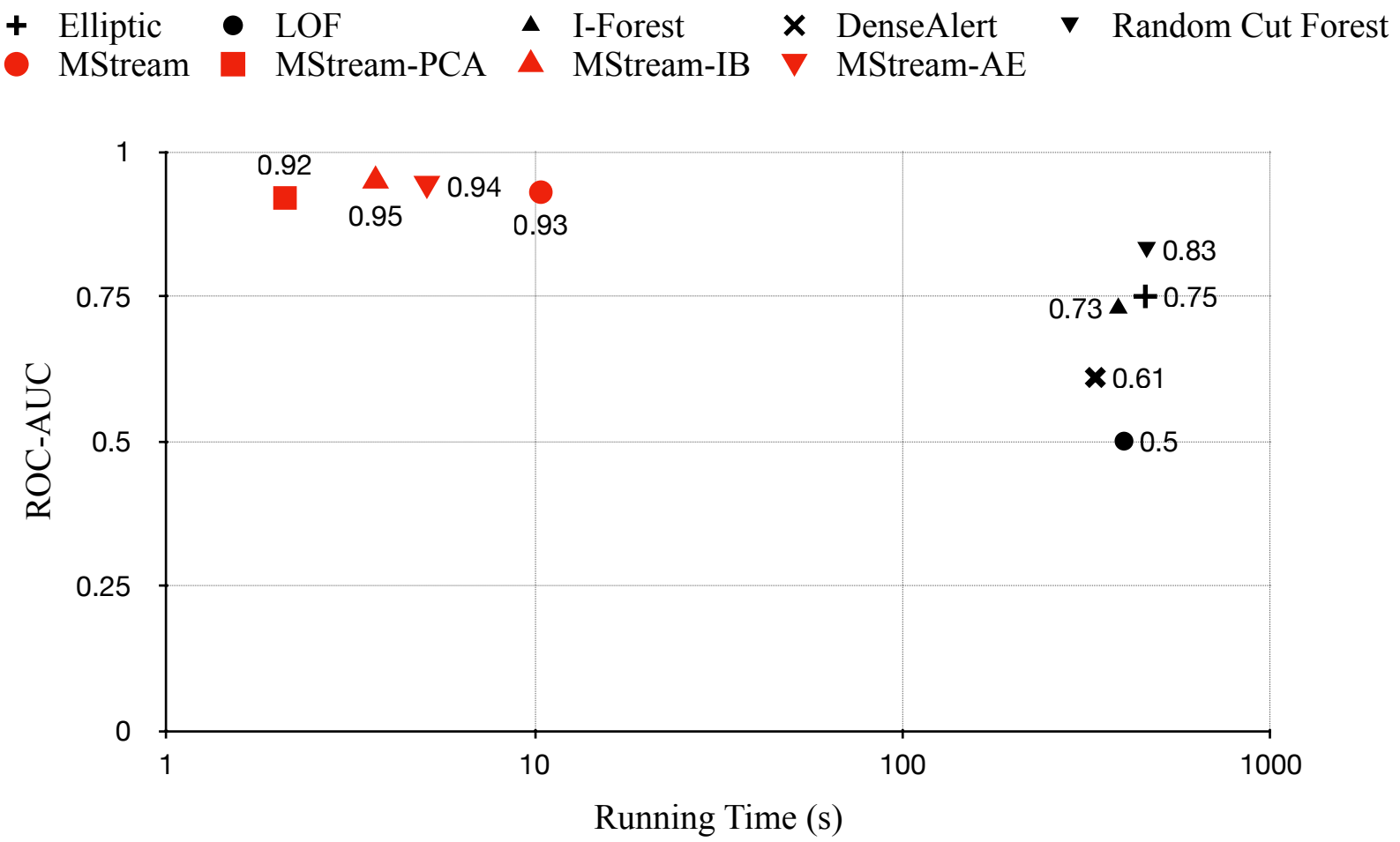
AUC

	Elliptic	LOF	I-Forest	DAlert	RCF	MSTREAM	MSTREAM-PCA	MSTREAM-IB	MSTREAM-AE
KDD	0.34 ± 0.025	0.34	0.81 ± 0.018	0.92	0.63	0.91 ± 0.016	0.92 ± 0.000	**0.96** ± 0.002	**0.96** ± 0.005
DoS	0.75 ± 0.021	0.50	0.73 ± 0.008	0.61	0.83	0.93 ± 0.001	0.92 ± 0.001	**0.95** ± 0.003	0.94 ± 0.001
UNSW	0.25 ± 0.003	0.49	0.84 ± 0.023	0.80	0.45	0.86 ± 0.001	0.81 ± 0.001	0.82 ± 0.001	**0.90** ± 0.001
DDoS	0.57 ± 0.106	0.46	0.56 ± 0.021	−−	0.63	0.91 ± 0.000	**0.94** ± 0.000	0.82 ± 0.000	0.93 ± 0.000

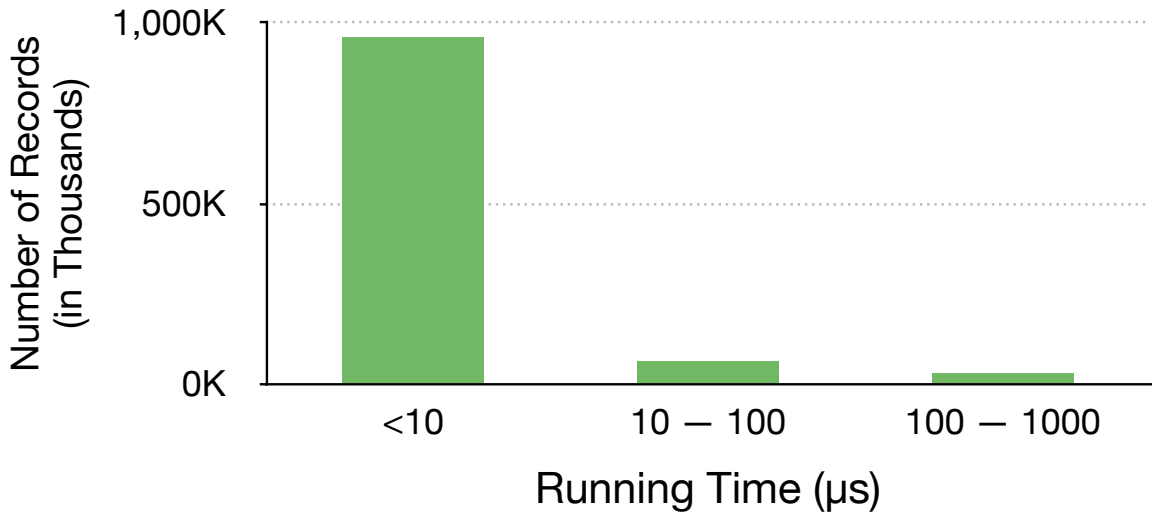
Running Time

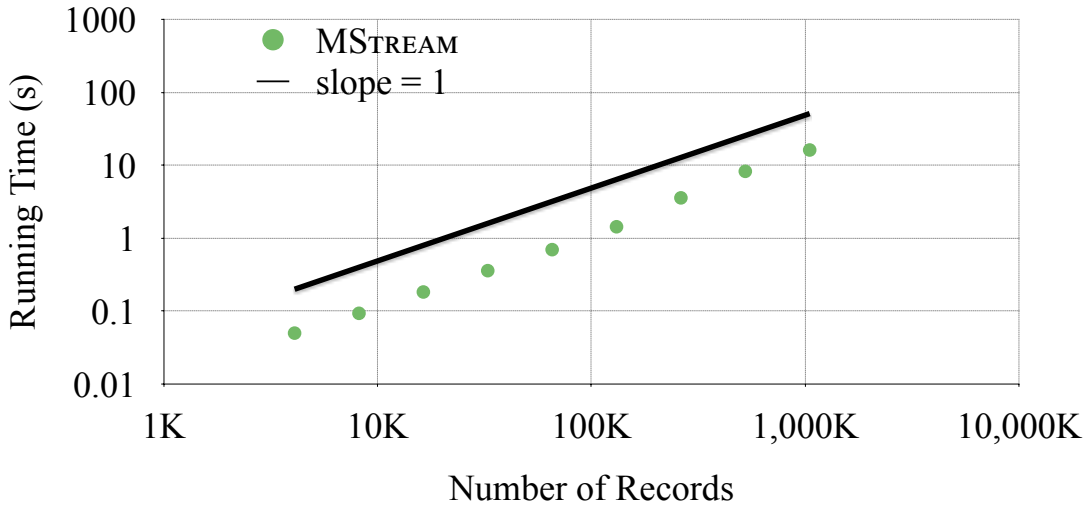
	Elliptic	LOF	I-Forest	DAlert	RCF	MSTREAM	MSTREAM-PCA	MSTREAM-IB	MSTREAM-AE
KDD	216.3	1478.8	230.4	341.8	181.6	4.3	2.5	3.1	3.1
DoS	455.8	398.8	384.8	333.4	459.4	10.4	2.1	3.7	5.1
UNSW	654.6	2091.1	627.4	329.6	683.8	12.8	6.6	8	8
DDoS	3371.4	15577*s*	3295.8	−−	4168.8	61.6	16.9	25.6	27.7

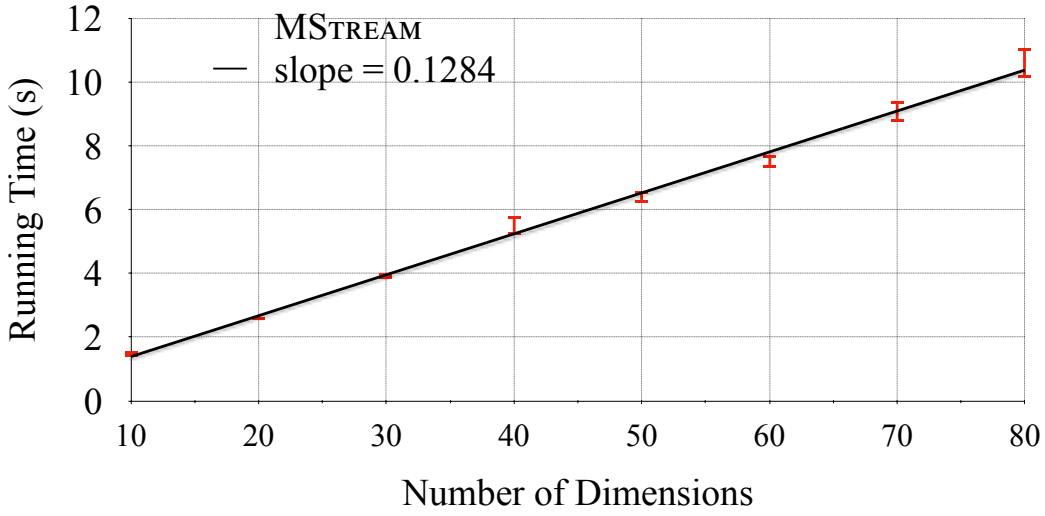
AUC vs Time

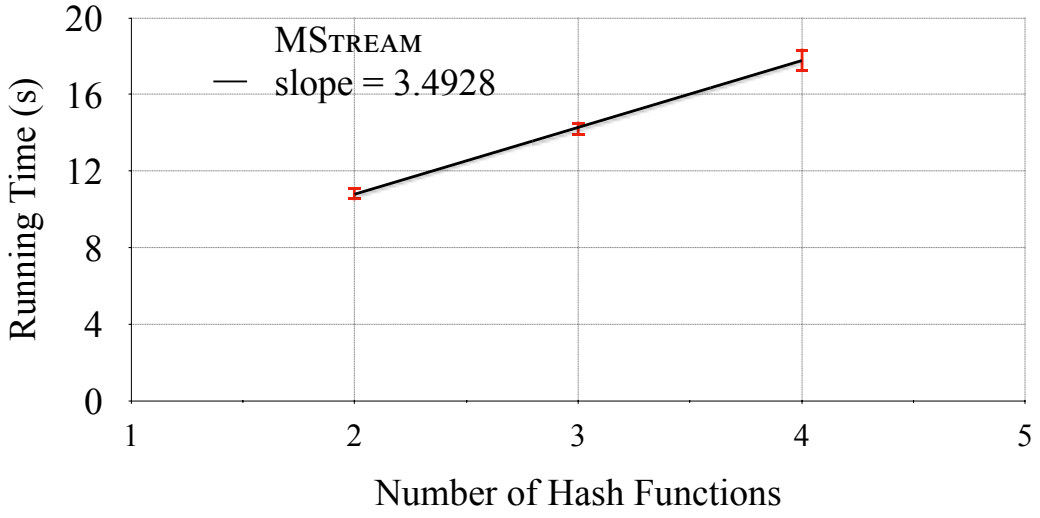

Elliptic
LOF
I-Forest
DenseAlert
Random Cut Forest
MStream
MStream-PCA
MStream-IB
MStream-AE
ROC-AUC
1
0.75
0.5
0.25
0
0.92
0.95
0.94
0.93
0.83
0.73
0.75
0.61
0.5
1
10
100
1000
Running Time (s)

Scalability









Roadmap

- Graphs
  - MIDAS
  - AnoEdge & AnoGraph
- Multi-Aspect Data
  - MStream
  - MemStream
- Conclusion

MemStream

MemStream: Memory-Based Streaming Anomaly Detection
Siddharth Bhatia, Arjit Jain, Shivin Srivastava, Kenji Kawaguchi, Bryan Hooi
WWW, 2022

MemStream

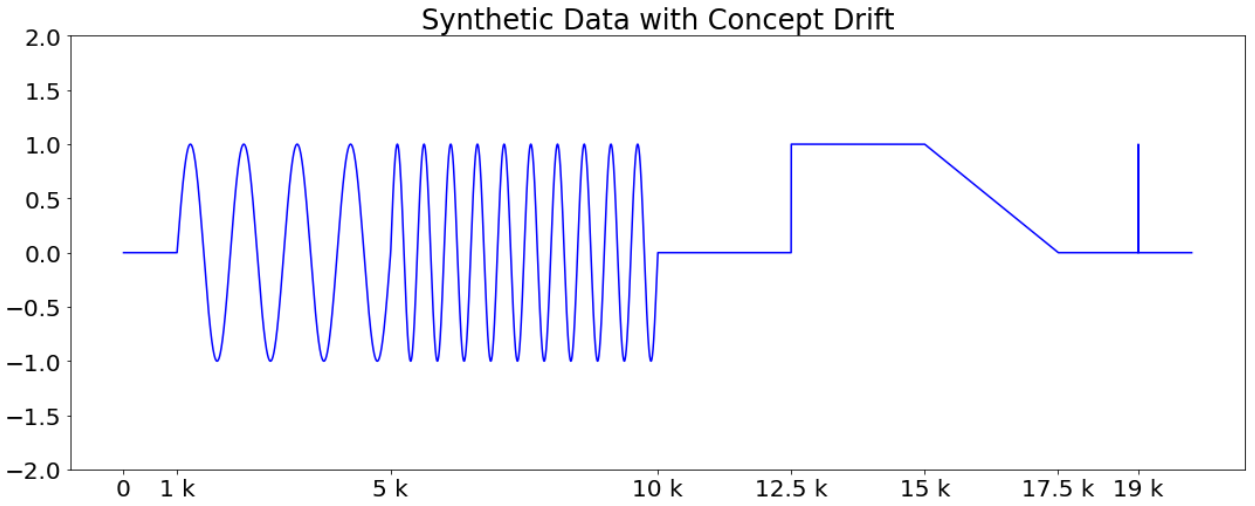
Input:
- Record stream $\boldsymbol{R}$ having concept drift
- Each having $\boldsymbol{d}$ dimensions

Output:
- Anomaly Score for each Record

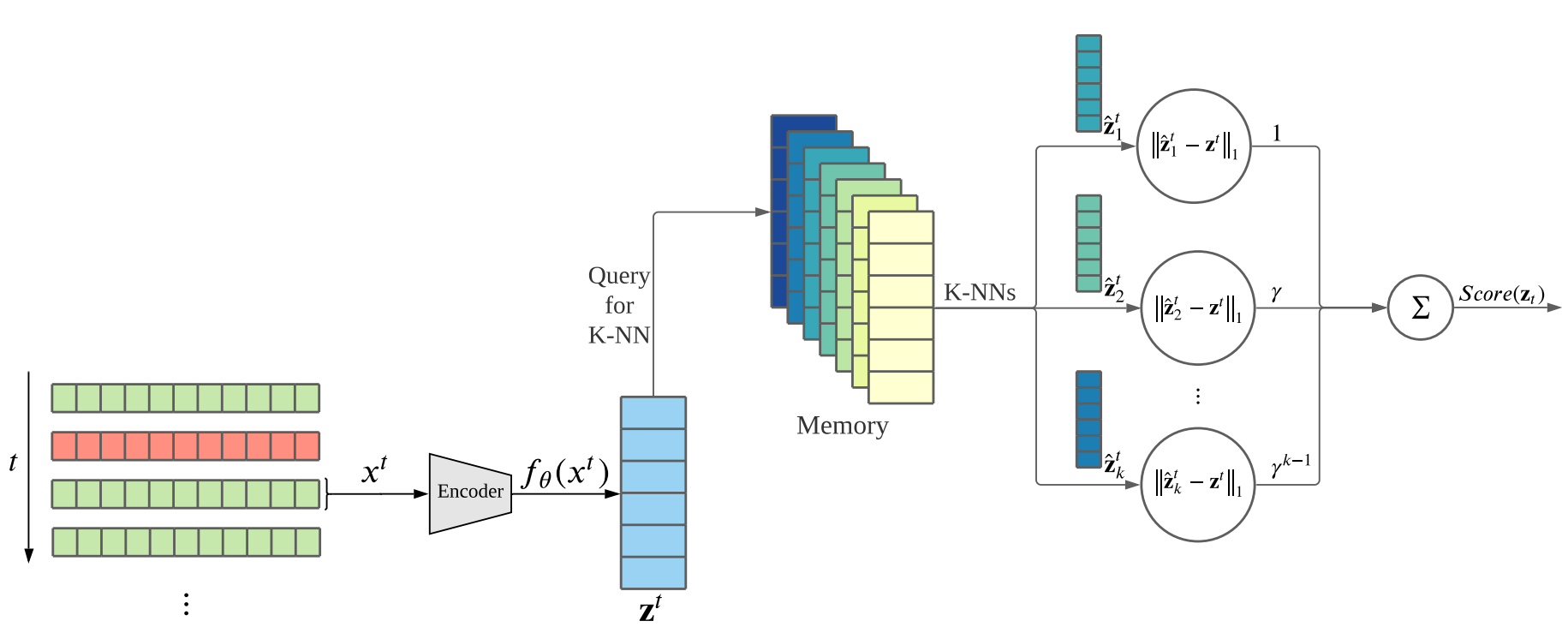
Our Contributions:
- Resilient to Concept Drift
- Theoretical Guarantees
- Quick Retraining
- Robustness to Memory Poisoning

Time	Feature 1	Feature 2	Feature 3	...
1	8.39	1.44	4.16	⋯
2	6.72	4.55	3.49	⋯
3	3.49	2.10	1.56	⋯
4	4.28	0.64	1.22	⋯
5	5.54	2.40	6.55	⋯
6	183.75	132.03	9.86	⋯
7	146.47	128.49	16.52	⋯
8	197.96	97.16	15.05	⋯
9	192.50	89.95	12.46	⋯
10	158.32	10.37	15.76	⋯

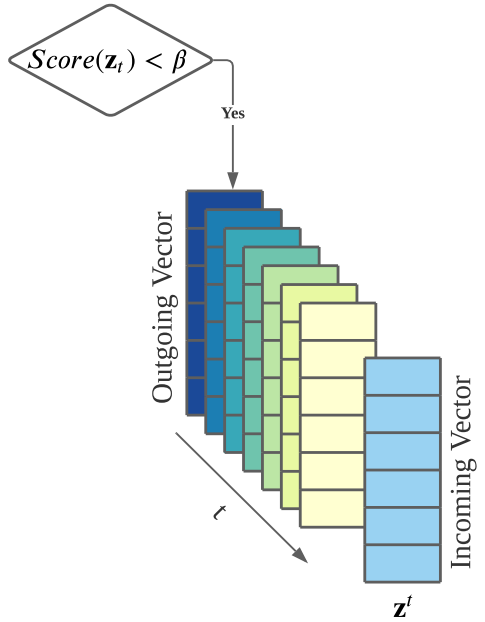
Concept Drift



MemStream



(i) Calculating Discounted Score



(ii) Updating Memory

MemStream: Algorithm

Input: Stream of data records
Output: Anomaly scores for each record

▷ **Initialization**
Feature Extractor, f_θ, trained using small subset of data $\mathcal{D}$
Memory, M, initialized as $f_\theta(\mathcal{D})$
while *new sample* $\mathbf{x}^t$ *is received:* **do**
  ▷ **Extract features:**
  $\mathbf{z}^t = f_\theta(\mathbf{x}^t)$
  ▷ **Query memory:**
  $\{\hat{\mathbf{z}}_1^t, \hat{\mathbf{z}}_2^t \ldots \hat{\mathbf{z}}_K^t\} = K$-nearest neighbours of $\mathbf{z}^t$ in M
  ▷ **Calculate distance:**
  $R(\mathbf{z}^t, \hat{\mathbf{z}}_i^t) = ||\mathbf{z}^t - \hat{\mathbf{z}}_i^t||_1$ **for all** $i \in 1..K$
  ▷ **Assign discounted score:**
  $$Score(\mathbf{z}^t) = \frac{\sum_{i=1}^{K} \gamma^{i-1} R(\mathbf{z}^t, \hat{\mathbf{z}}_i^t)}{\sum_{i=1}^{K} \gamma^{i-1}}$$
  ▷ **Update Memory:**
  if $Score(\mathbf{z}^t) < \beta$ **then**
    Replace earliest added element in M with $\mathbf{z}^t$
  ▷ **Anomaly Score:**
  output $Score(\mathbf{z}^t)$

EXPERIMENTS

Baselines and Datasets

Method	KDD99	NSL	UNSW	DoS	Syn.	Ion.	Cardio	Sat.	Sat.-2	Mamm.	Pima	Cover
STORM (CIKM'07)												
HS-Tree (IJCAI'11)												
iForestASD (ICONS'13)												
RS-Hash (ICDM'16)												
RCF (ICML'16)												
LODA (ML'16)												
Kitsune (NDSS'18)												
DILOF (KDD'18)												
xSTREAM (KDD'18)												
MSTREAM (WWW'21)												
Ex. IF (TKDE'21)												

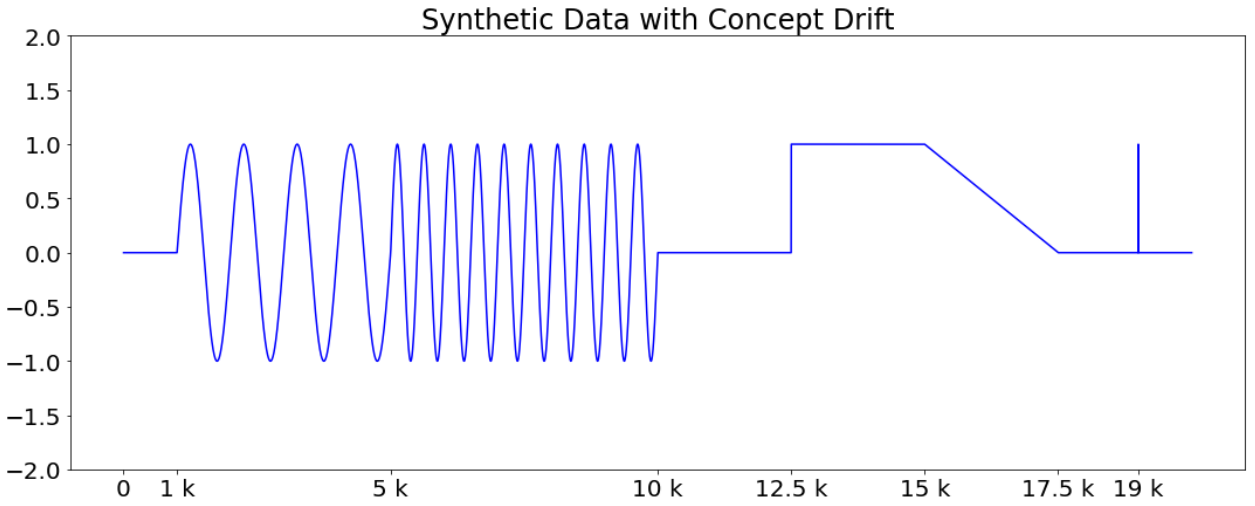
AUC

Method	KDD99	NSL	UNSW	DoS	Syn.	Ion.	Cardio	Sat.	Sat.-2	Mamm.	Pima	Cover
STORM (CIKM'07)	0.914	0.504	0.810	0.511	0.910	0.637	0.507	0.662	0.514	0.650	0.528	0.778
HS-Tree (IJCAI'11)	0.912	0.845	0.769	0.707	0.800	0.764	0.673	0.519	0.929	0.832	0.667	0.731
iForestASD (ICONS'13)	0.575	0.500	0.557	0.529	0.501	0.694	0.515	0.504	0.554	0.574	0.525	0.603
RS-Hash (ICDM'16)	0.859	0.701	0.778	0.527	0.921	0.772	0.532	0.675	0.685	0.773	0.562	0.640
RCF (ICML'16)	0.791	0.745	0.512	0.514	0.774	0.675	0.617	0.552	0.738	0.755	0.571	0.586
LODA (ML'16)	0.500	0.500	– – –	0.500	0.506	0.503	0.501	0.500	0.500	0.500	0.502	0.500
Kitsune (NDSS'18)	0.525	0.659	0.794	0.907	– – –	0.514	0.966	0.665	0.973	0.592	0.511	0.888
DILOF (KDD'18)	0.535	0.821	0.737	0.613	0.703	**0.928**	0.570	0.561	0.563	0.733	0.543	0.688
xSTREAM (KDD'18)	0.957	0.552	0.804	0.800	0.539	0.847	0.918	0.677	**0.996**	0.856	0.663	0.894
MSTREAM (WWW'21)	0.844	0.544	0.860	0.930	0.505	0.670	**0.986**	0.563	0.958	0.567	0.529	0.874
Ex. IF (TKDE'21)	0.874	0.767	0.541	0.734	– – –	0.872	0.921	0.716	0.995	0.867	0.672	0.902
MEMSTREAM	**0.980**	**0.978**	**0.972**	**0.938**	**0.955**	0.821	0.884	**0.727**	0.991	**0.894**	**0.742**	**0.952**

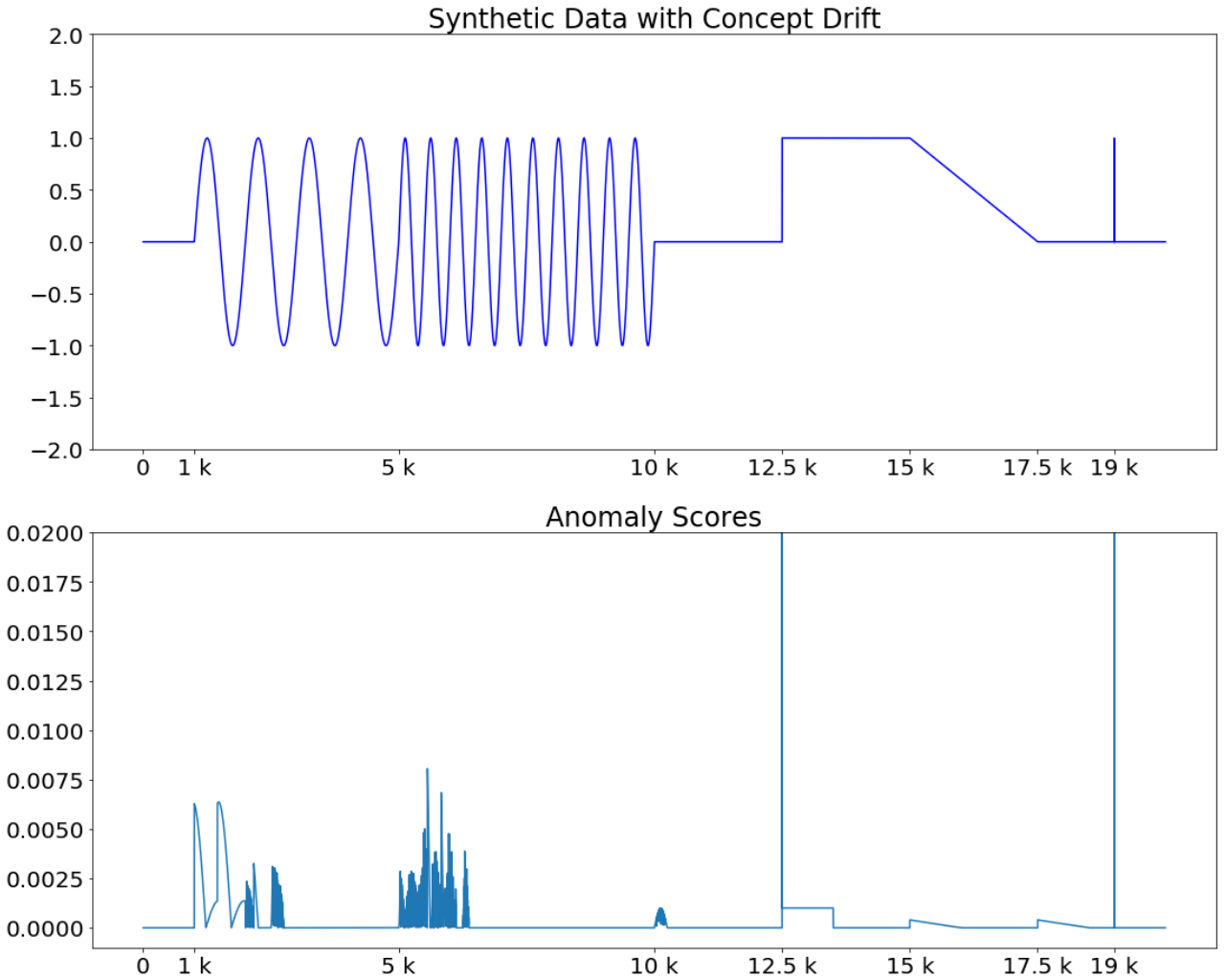
AUC-PR and Running Time

Method	AUC-PR	Time (s)
STORM	0.681 ± 0.000	754
HS-Tree	0.709 ± 0.063	306
iForestASD	0.534 ± 0.000	19876
RS-Hash	0.500 ± 0.140	892
RCF	0.664 ± 0.006	665
LODA	0.734 ± 0.067	2617
Kitsune	0.673 ± 0.000	821
DILOF	0.822 ± 0.000	260
xSTREAM	0.541 ± 0.070	34
MSTREAM	0.510 ± 0.000	0.08
Ex. IF	0.659 ± 0.014	889
MEMSTREAM	**0.959** ± 0.002	55

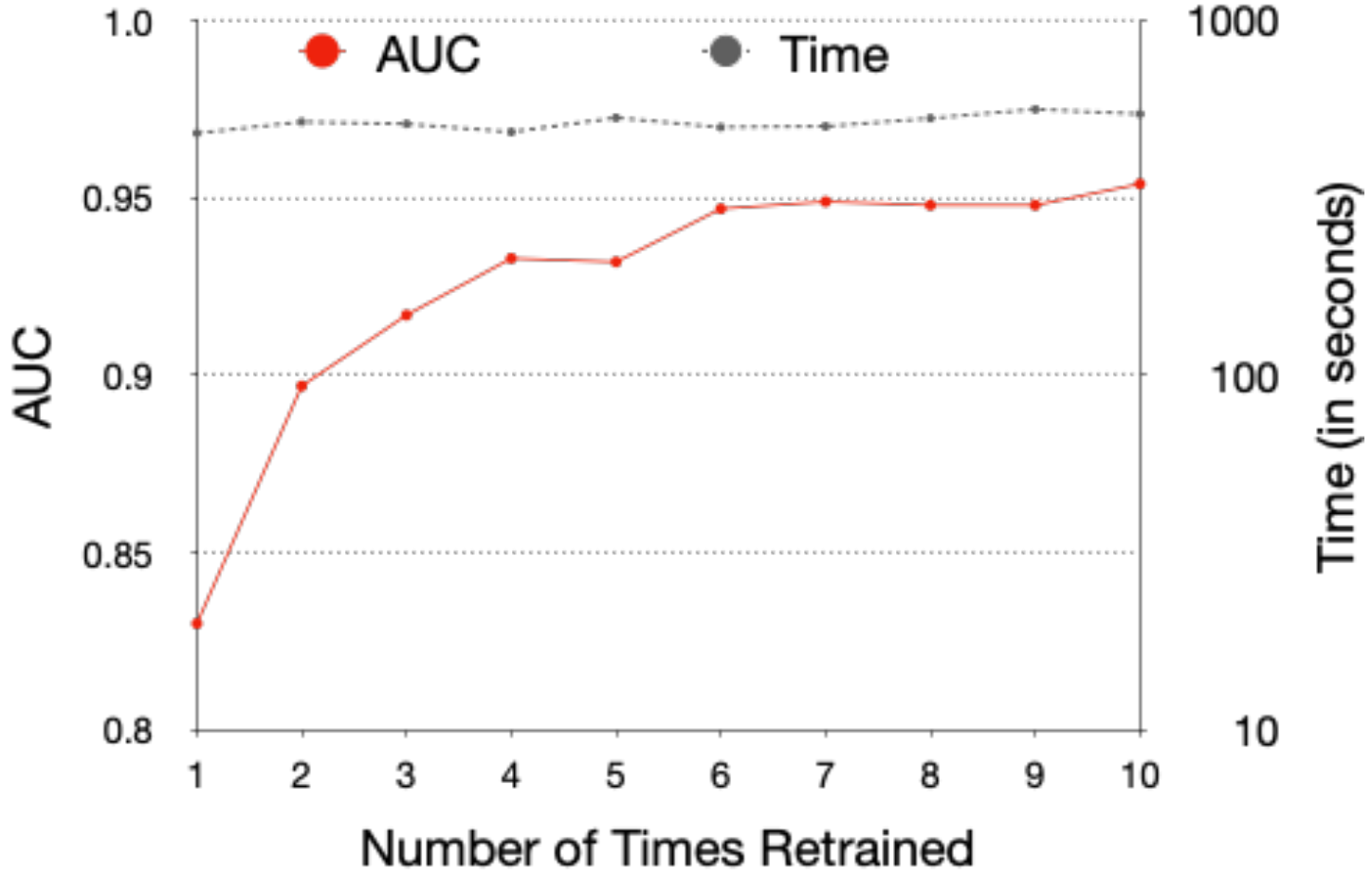
Concept Drift



Concept Drift



Retraining


AUC
Time
1.0
0.95
0.9
0.85
0.8
AUC
1000
100
10
Time (in seconds)
1
2
3
4
5
6
7
8
9
10
Number of Times Retrained

Self Correction

γ	High $\beta(=1)$	Appropriate $\beta(=0.001)$
0	0.771	0.933
0.25	0.828	0.966
0.5	0.848	0.967
1	0.888	0.965

Ablations

	Component	Ablations			
(a)	Memory Update	None 0.938	LRU 0.946	RR 0.946	FIFO 0.980
(b)	Feature Extraction	Identity 0.822	PCA 0.863	IB 0.959	AE 0.980
(c)	Memory Length (N)	128 0.950	256 0.980	512 0.946	1024 0.811
(d)	Output Dimension (D)	$d/2$ 0.951	d 0.928	$2d$ 0.980	$5d$ 0.983
(e)	Update Threshold (β)	1 0.980	0.1 0.938	0.01 0.938	0.001 0.938
(f)	KNN coefficient (γ)	0 0.980	0.25 0.939	0.5 0.937	1 0.936

CONCLUSION

Conclusion

Setting	Anomaly Type	Data Structure	Method
Graph	Edges	Count-Min Sketch	MIDAS [AAAI20 & TKDD22]
Graph	Edges + Subgraphs	Higher-Order Sketch	ANOEDGE/ANOGRAPH [Under Submission]
Multi-Aspect Data	Records	Count-Min Sketch	MSTREAM [WWW21]
Multi-Aspect Data	Records	Autoencoder + Memory	MEMSTREAM [WWW22]

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

FUTURE WORK

- AnoEdge/AnoGraph: Symmetrical → Rectangular H-CMS
- MemStream: Memory Replacement Policies
- Embeddings, Heterogeneous graphs
- Wide range of data stream rates: Parallel Computing
- Exploring new applications: Predictive maintenance, environmental monitoring, social media data streams, medical data
- Graphs → Multi-Aspect Data → Complex Data
- Multi-Modal approaches
- Incorporating semi-supervision/human-assisted feedback
- Hybrid of deep learning models and streaming data structures

Impact

Open Source Traction
MIDAS was implemented in C++, Python, Golang, Ruby, Rust, R, Java, and Julia
Our projects received 900+ stars on GitHub

Awards
MStream was the WWW'21 Best Paper Finalist
MIDAS won the popular choice award at Microsoft Azure Hackathon'20

Invited Talks:
Invited by the MIT's Data Systems Group, Oxford's Alan Turing Institute, NYU's Center for Data Science etc.

Press Coverage
Our research was covered by ACM TechNews, Bloomberg, AIhub, Hacker News, KDnuggets, and others.

Community Service
Co-organised the ODD workshop at KDD'21 with collaborators from CMU, Facebook and Google

Other Publications

1. **Siddharth Bhatia∗**, Arjit Jain∗, and Bryan Hooi. “ExGAN: Adversarial Generation of Extreme Samples”. *AAAI Conference on Artificial Intelligence (AAAI)* 2021 [* equal contribution].
2. **Siddharth Bhatia**, Yiwei Wang, Bryan Hooi, and Tanmoy Chakraborty. “GraphAnoGAN: Detecting Anomalous Snapshots from Attributed Graphs”. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)* 2021.
3. Koki Kawabata∗, **Siddharth Bhatia**∗, Rui Liu, Mohit Wadhwa, and Bryan Hooi. “SSMF: Shifting Seasonal Matrix Factorization”. *Conference on Neural Information Processing Systems (NeurIPS)* 2021. [* equal contribution].
4. Yiwei Wang, Yujun Cai, Yuxuan Liang, Henghui Ding, Changhu Wang, **Siddharth Bhatia**, and Bryan Hooi. “Adaptive Data Augmentation on Temporal Graphs”. *Conference on Neural Information Processing Systems (NeurIPS)* 2021.
5. Jiabao Zhang, Shenghua Liu, Wenting Hou, **Siddharth Bhatia**, Huawei Shen, Wenjian Yu, and Xueqi Cheng. “AugSplicing: Synchronized Behavior Detection in Streaming Tensors”. *AAAI Conference on Artificial Intelligence (AAAI)* 2021.
6. Xiaobing Sun, Wenjie Feng, Shenghua Liu, Yuyang Xie, **Siddharth Bhatia**, Bryan Hooi, Wenhan Wang, and Xueqi Cheng. “MonLAD: Money Laundering Agents Detection in Transaction Streams”. *ACM International Conference on Web Search and Data Mining (WSDM)* 2022.
7. Ying Sun, Wenjun Wang, Nannan Wu, ChaoChao Liu, **Siddharth Bhatia**, Yang Yu, and Wei Yu. “AAAN: Anomaly Alignment in Attributed Networks”. *Knowledge Based Systems* 2022.
8. **Siddharth Bhatia** and Sudipto Guha. “Semi-Supervised Anomaly Detection via Sketches”. *(Under Submission).*
9. Shivin Srivastava, **Siddharth Bhatia**, Lingxiao Huang, Jun Heng Lim, Kenji Kawaguchi, Vaibhav Rajan. “Don’t Just Divide, Polarize and Conquer!”. (Under Submission).
10. Rui Liu, **Siddharth Bhatia**, Bryan Hooi. “Isconna: Streaming Anomaly Detection with Frequency and Patterns”. *(Under Submission).*

Thank you Admin!

Wei Ngan Chin
Li-Shiuan Peh
Beng Chin Ooi
Xiaokui Xiao
Line Fong
Agnes Ang
Aminah Ayu
Thiba Ahwahday
Irene Chuan
Catharine Tan
Sarada A
Aerin Oon
Goh Lee Kheng

Thank you Collaborators!

Christos Faloutsos, CMU
Philip S. Yu, UIC
Pan Li, Purdue University
Arjit Jain, Google
Mohit Wadhwa, Google
Neil Shah, Snap
Ritesh Kumar, Tower Research
Shenghua Liu, Chinese Academy of Sciences
Nannan Wu, Tianjin University
Ying Sun, Tianjin University
Koki Kawabata, SANKEN
Kijung Shin, KAIST
Tanmoy Chakraborty, IIT
Minji Yoon, CMU
Rui Liu, NUS
Kenji Kawaguchi, NUS
Yiwei Wang, NUS
Vaibhav Rajan, NUS
Shivin Srivastava, NUS

Thank you!

Advisor: Bryan Hooi