

Sketch-based Change Detection

Balachander Krishnamurthy (AT&T)

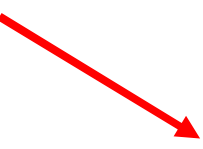
Subhabrata Sen (AT&T)

Yin Zhang (AT&T)

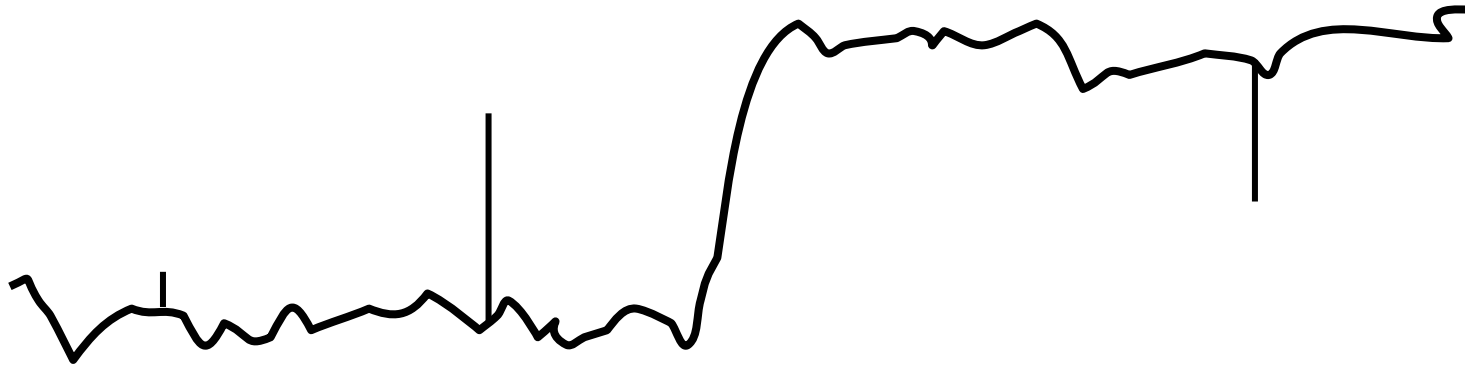
Yan Chen (UCB/AT&T)

ACM Internet Measurement Conference 2003

Network Anomaly Detection

- Network anomalies are common
 - Flash crowds, failures, DoS, worms, ...
 - Want to catch them quickly and accurately
 - Two basic approaches
 - Signature-based: looking for known patterns
 - E.g. backscatter [Moore et al.] uses address uniformity
 - Easy to evade (e.g., mutating worms)
 - Statistics-based: looking for abnormal behavior
 - E.g., heavy hitters, big changes
 - Prior knowledge not required
-  This talk

Change Detection



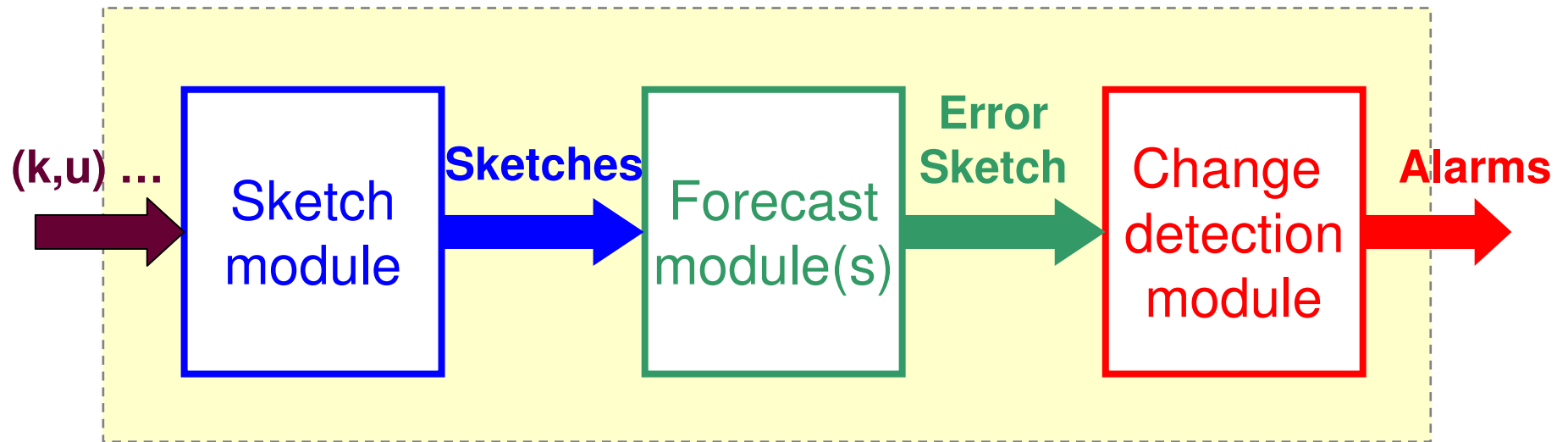
- Lots of prior work
 - Simple smoothing & forecasting
 - Exponentially weighted moving average (EWMA)
 - Box-Jenkins (ARIMA) modeling
 - Tsay, Chen/Liu (in statistics and economics)
 - Wavelet-based approach
 - Barford et al. [IMW01, IMW02]

The Challenge

- Potentially tens of millions of time series !
 - Need to work at very low aggregation level (e.g., IP level)
 - Changes may be buried inside aggregated traffic
 - The Moore's Law on traffic growth ... ☹️
- Per-flow analysis is too slow / expensive
 - Want to work in near real time
- Existing approaches not directly applicable
 - Estan & Varghese focus on heavy-hitters

Need scalable change detection

Sketch-based Change Detection



- Input stream: (key, update)
- Summarize input stream using sketches
- Build forecast models on top of sketches
- Report flows with large forecast errors

Outline

- Sketch-based change detection
 - Sketch module
 - Forecast module
 - Change detection module
- Evaluation
- Conclusion & future work

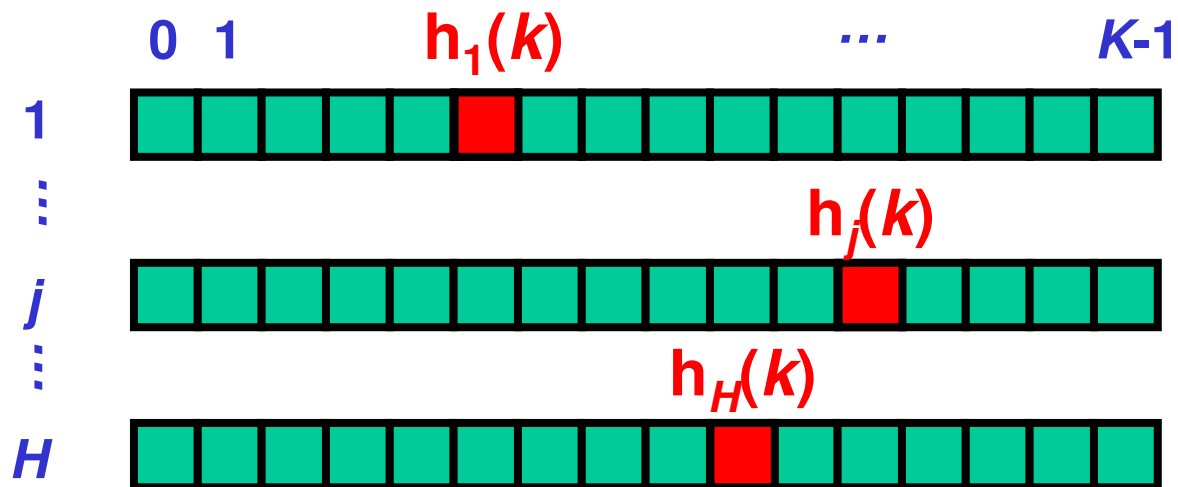
Sketch

- Probabilistic summary of data streams
 - Originated in STOC 1996 [AMS96]
 - Widely used in database research to handle massive data streams

	Space	Accuracy
Hash table	Per-key state	100%
Sketch	Compact	With probabilistic guarantees (better for larger values)

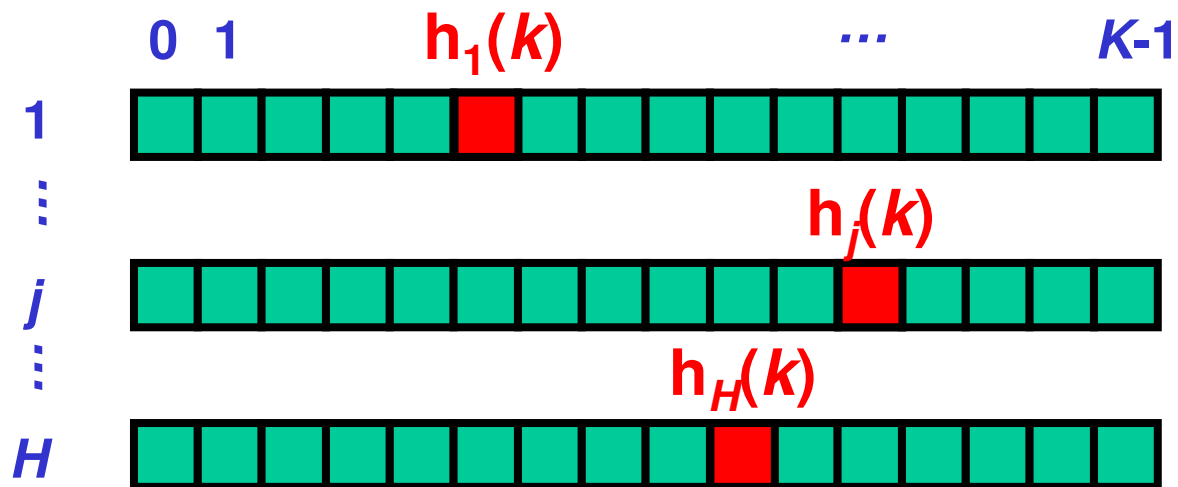
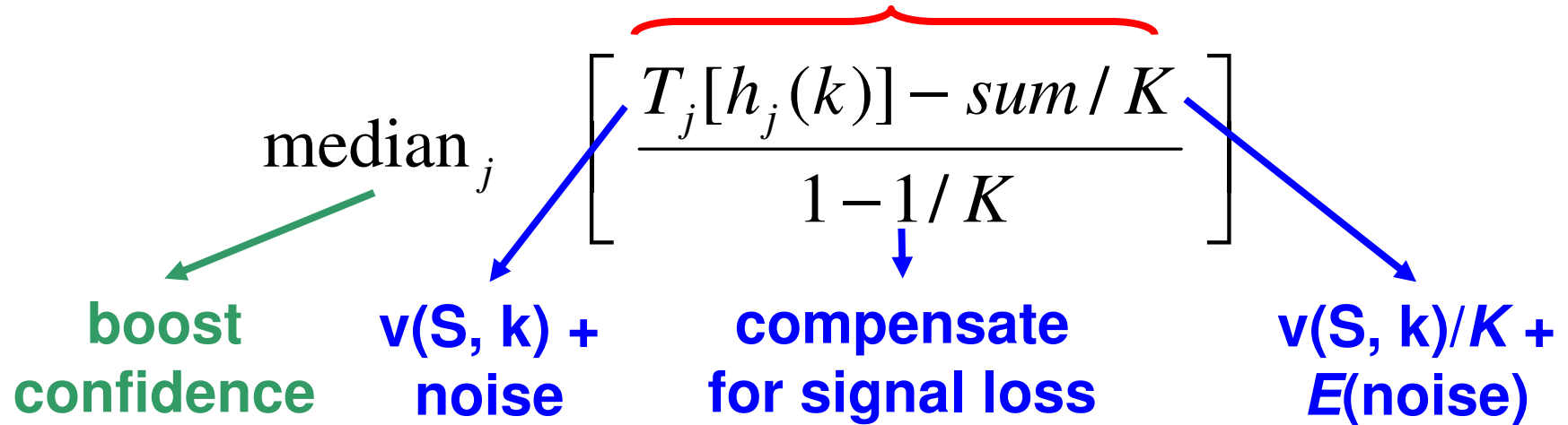
K-ary Sketch

- Array of hash tables: $T_j[K]$ ($j = 1, \dots, H$)
 - Similar to count sketch, counting bloom filter, multi-stage filter, ...
- Update (k, u) : $T_j[h_j(k)] += u$ (for all j)



K-ary Sketch (cont'd)

- Estimate $v(S, k)$: sum of updates for key k
unbiased estimator of $v(S, k)$ with low variance

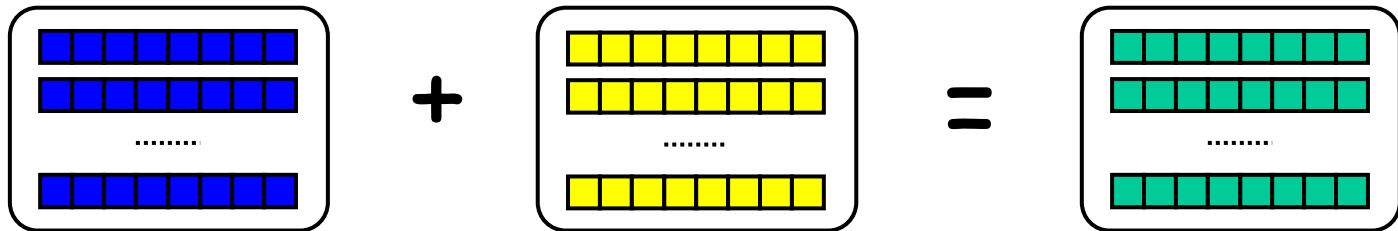


K-ary Sketch (cont'd)

- Estimate the second moment (F_2)

$$F_2(S) = \sum_k [v(S, k)]^2$$

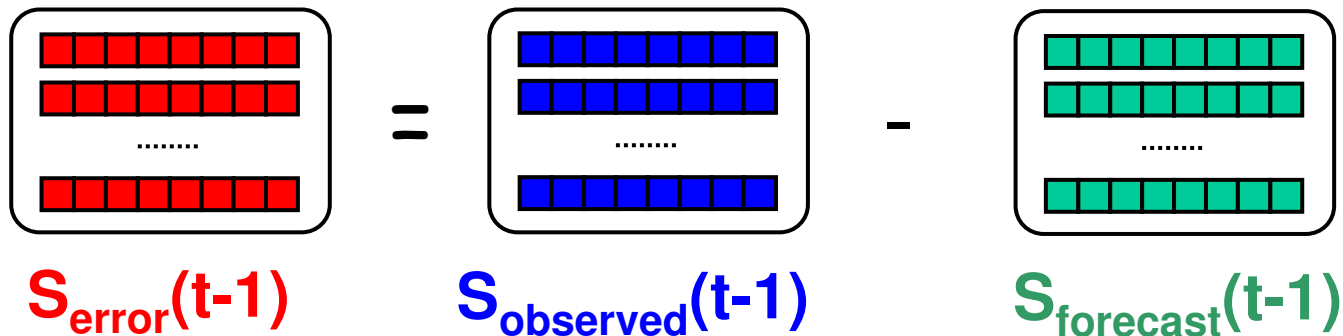
- Sketches are linear
 - Can combine sketches



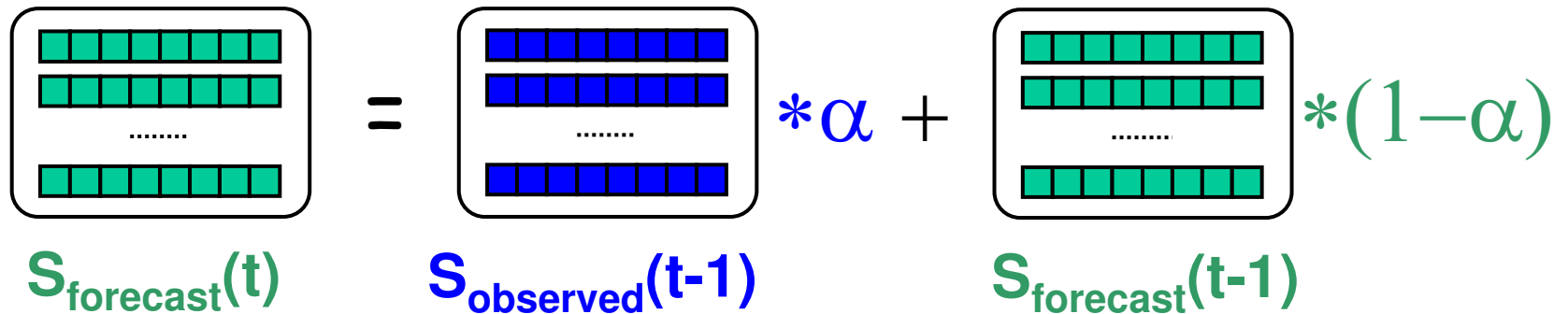
- Can aggregate data from different times, locations, and sources

Forecast Model: EWMA

- Compute forecast error sketch: S_{error}



- Update forecast sketch: S_{forecast}



Other Forecast Models

- Simple smoothing methods
 - Moving Average (MA)
 - S-shaped Moving Average (SMA)
 - Non-Seasonal Holt-Winters (NSHW)
- ARIMA models (p,d,q)
 - ARIMA 0 ($p \leq 2, d=0, q \leq 2$)
 - ARIMA 1 ($p \leq 2, d=1, q \leq 2$)

Find Big Changes

- Top N
 - Find N biggest forecast errors
 - Need to maintain a heap
- Thresholding
 - Find forecast errors above a threshold

$$v(S_{\text{error}}, k) \geq \text{Thresh} \times \sqrt{F_2(S_{\text{error}})}$$

Evaluation Methodology

- Accuracy
 - Metric: similarity to per-flow analysis results
 - This talk focuses on
 - TopN (Thresholding is very similar)
 - Accuracy on real traces (Also has data-independent probabilistic accuracy guarantees)
- Efficiency
 - Metric: time per operation
- Dataset description

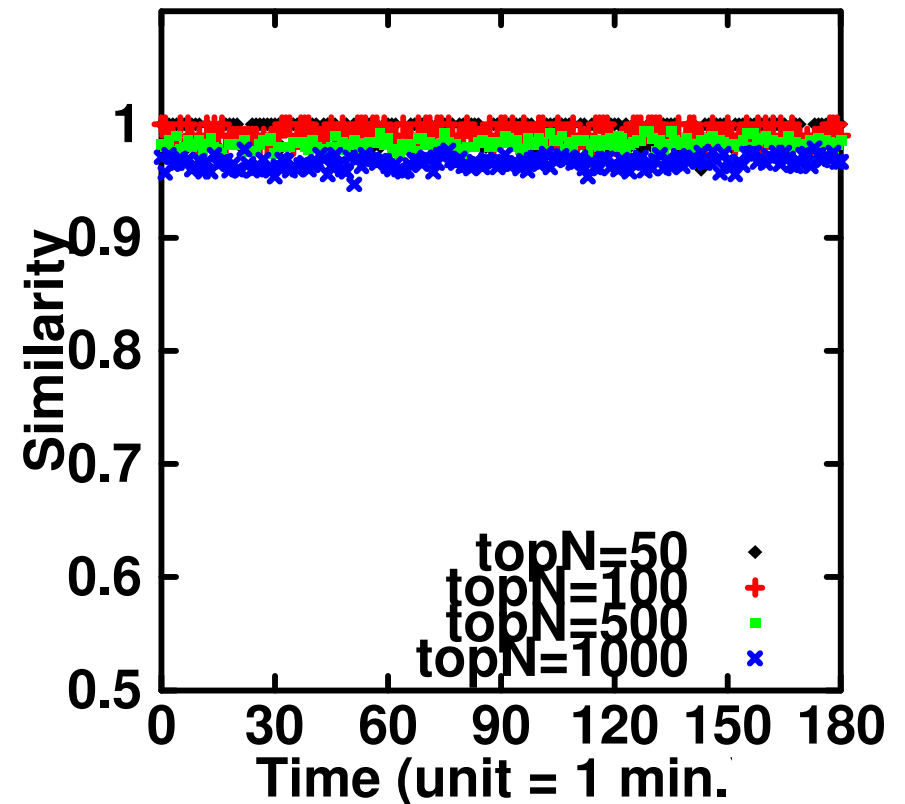
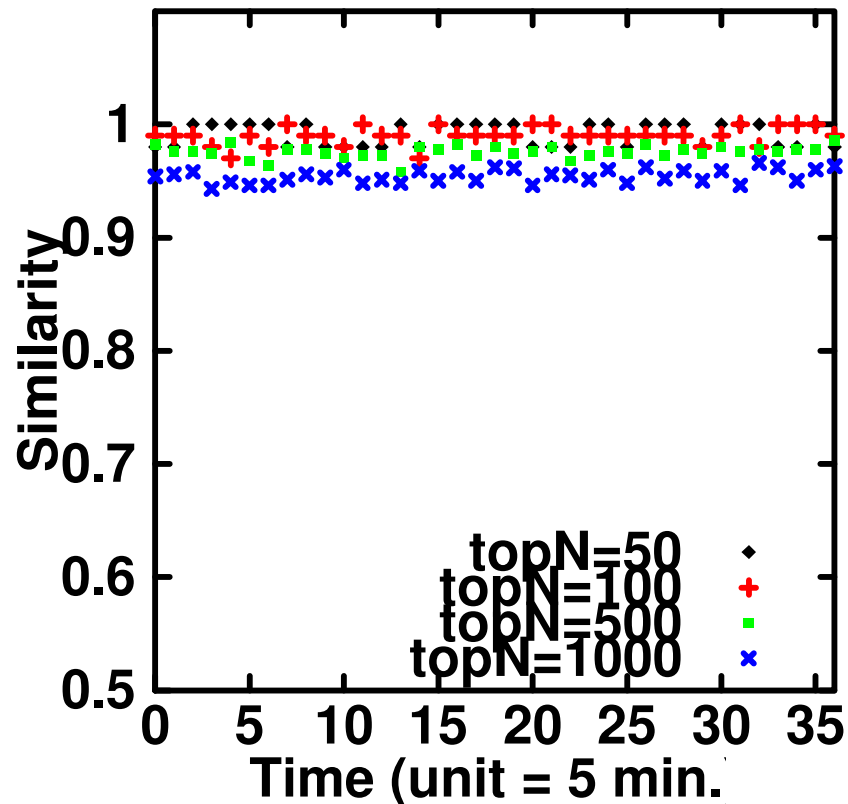
Data	Duration	#routers	#records	
			Total	Range
Netflow	4 hours	10	190M	861K - 60M

Experimental parameters

Parameter	Values
H	1, 5, 9, 25
K	8K, 32K, 64K
N	50, 100, 500, 1000
Interval	1 min., 5 min.
Model	6 forecast models
Router	10 (this talk: Large, Medium)

Accuracy

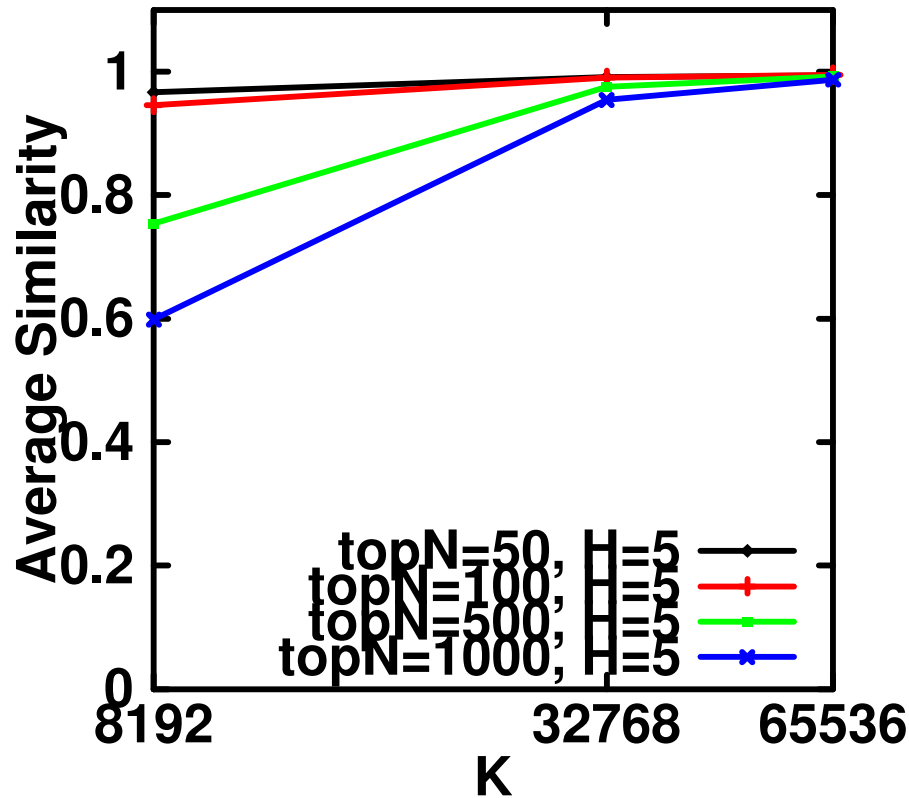
H = 5, K = 32768, Router = Large



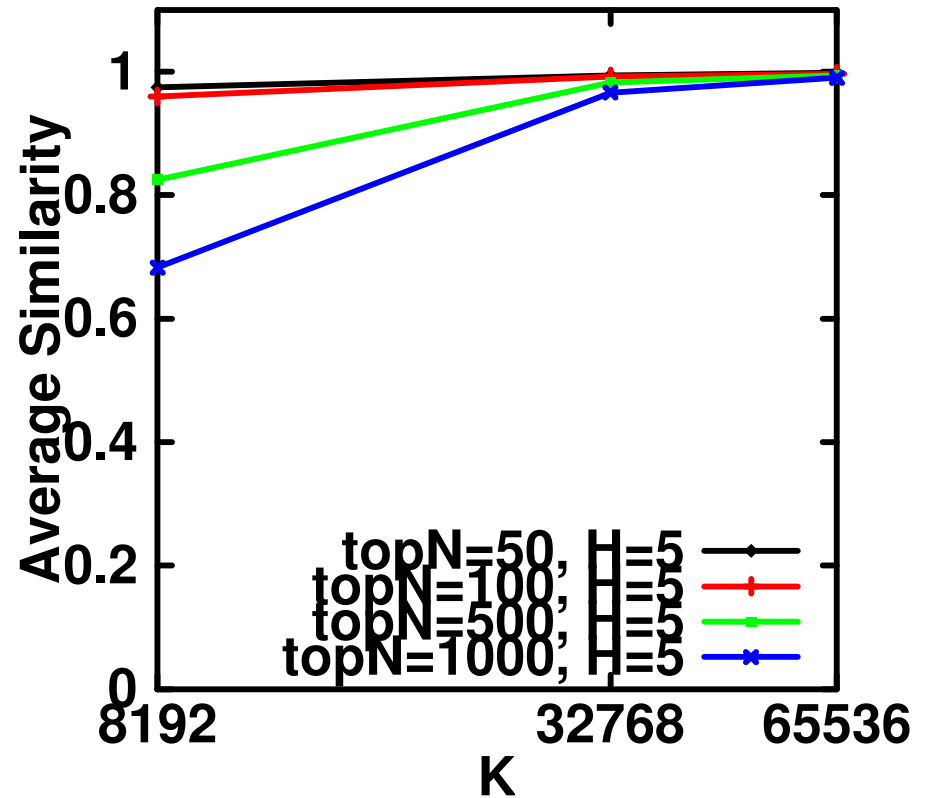
$$\text{Similarity} = | \text{TopN_sketch} \cap \text{TopN_perflow} | / N$$

Accuracy (cont'd)

Model = EWMA, Router = Large



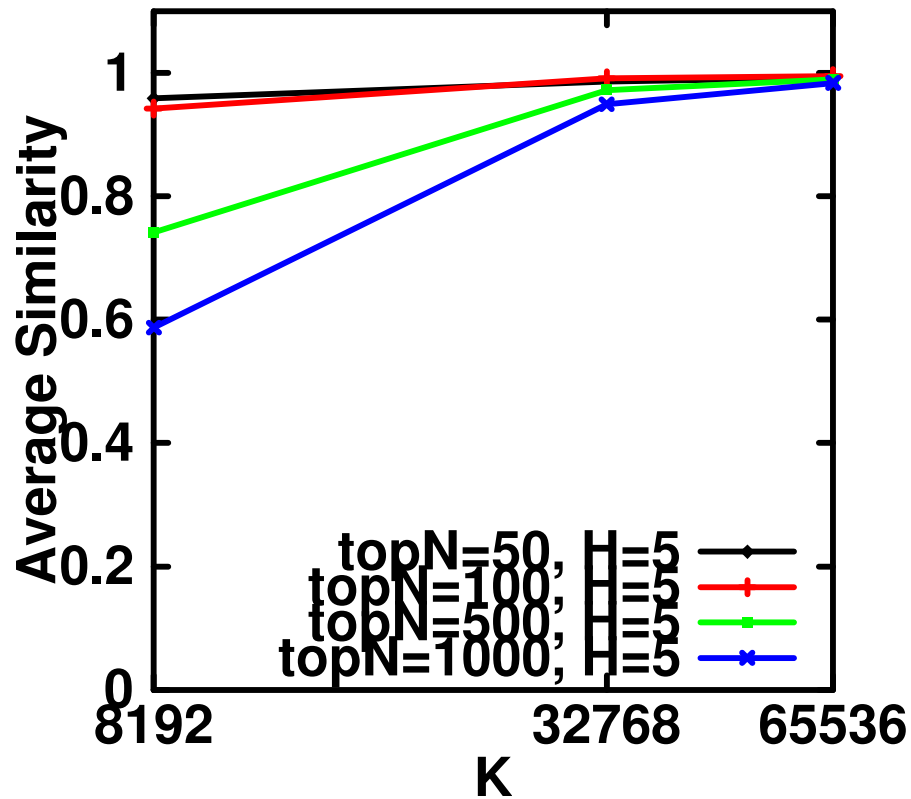
Interval = 5 min.



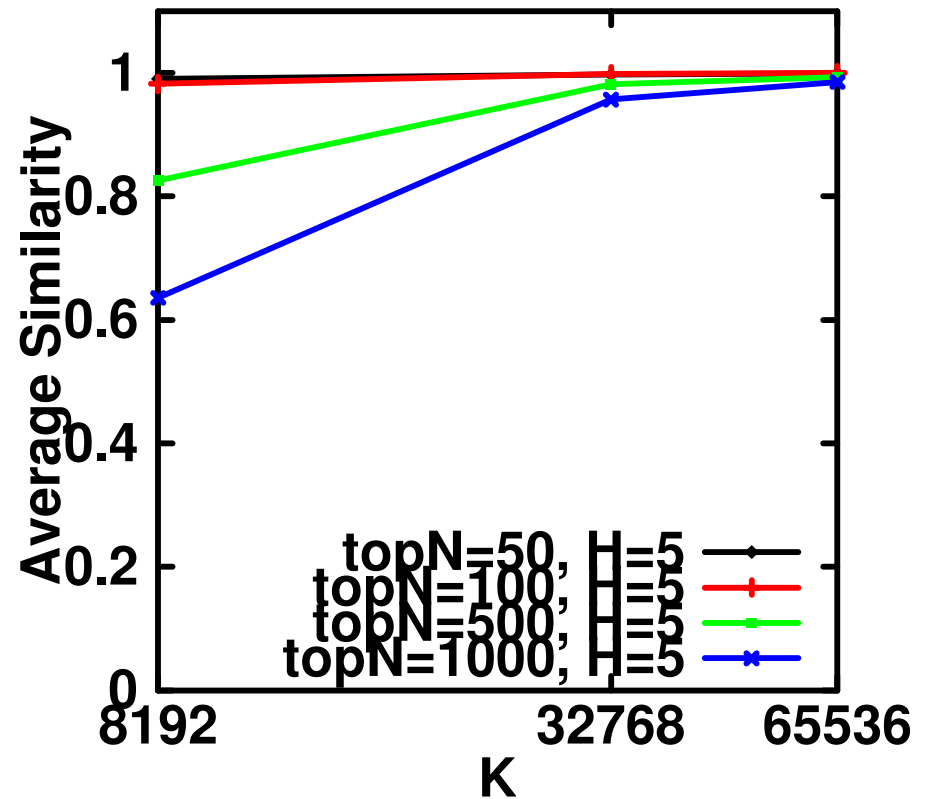
Interval = 1 min.

Accuracy (cont'd)

Model = ARIMA 0, Interval = 5 min.



Router = Large



Router = Medium

Accuracy Summary

- For small N (50, 100), even small K (8K) gives very high accuracy
- For large N (1000), $K = 32K$ gives about 95% accuracy
- Router, interval, and forecast model make little difference
- H generally has little impact

Efficiency

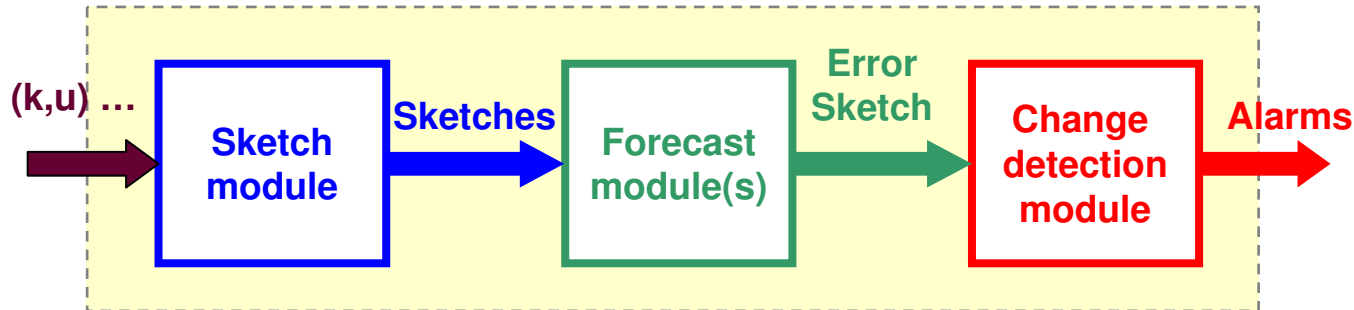
Operation (H=5, K = 64K)	Nanoseconds per operation	
	400MHz SGI R12k	900MHz Ultrasparc-II
Hash computation	34	89
Update cost	81	45
Estimate cost	269	146

1 Gbps = 320 nsec per 40-byte packet

Can potentially work in near real time.

Conclusion

- Sketch-based change detection



- Scalable
 - Can handle tens of millions of time series
- Accurate
 - Provable probabilistic accuracy guarantees
 - Even more accurate on real Internet traces
- Efficient
 - Can potentially work in near real time

Ongoing and Future Work

- Refinements
 - Avoid boundary effects due to fixed interval
 - Automatically reconfigure parameters
 - Combine with sampling
- Extensions
 - Online detection of multi-dimensional hierarchical heavy hitters and changes
- Applications
 - Building block for network anomaly detection

Thank you!

Heavy Hitter vs. Change Detection

- Change detection for heavy hitters
 - Can potentially use different parameters for different flows
 - Heavy hitter \neq big change
 - Need small threshold to avoid missing changes
 - Aggregation is difficult
- Sketch-based change detection
 - All flows share the same model parameters
 - Aggregation is very easy due to linearity