



Data Stream Mining

Philip S. Yu (psyu@cs.uic.edu)

Professor & Wexler Chair

University of Illinois at Chicago



Why Data Stream

- With advances in technology, various data sources (e.g. sensors, GPS units) can continue to generate data at high rate
- Simple everyday activities/transactions such as using the credit card or the phone are recorded in an automated way
- Either by machines, human beings or model simulations, various data streams are generated at high rate



Real-time Stream Applications

- Smart grid monitoring and management
- Climate change monitoring and weather prediction
- Vehicle trajectory monitoring
- Air or water monitoring
- Trade surveillance for security fraud and money laundering
- Sensor networks for monitoring intelligent oil wells, manufacturing plants, RFID products, etc
- Network monitoring for intrusion detection



Challenges

- Data continue to arrive at high rate potentially with evolving characteristics
- Need to support continual real-time mining and monitoring for immediate actions

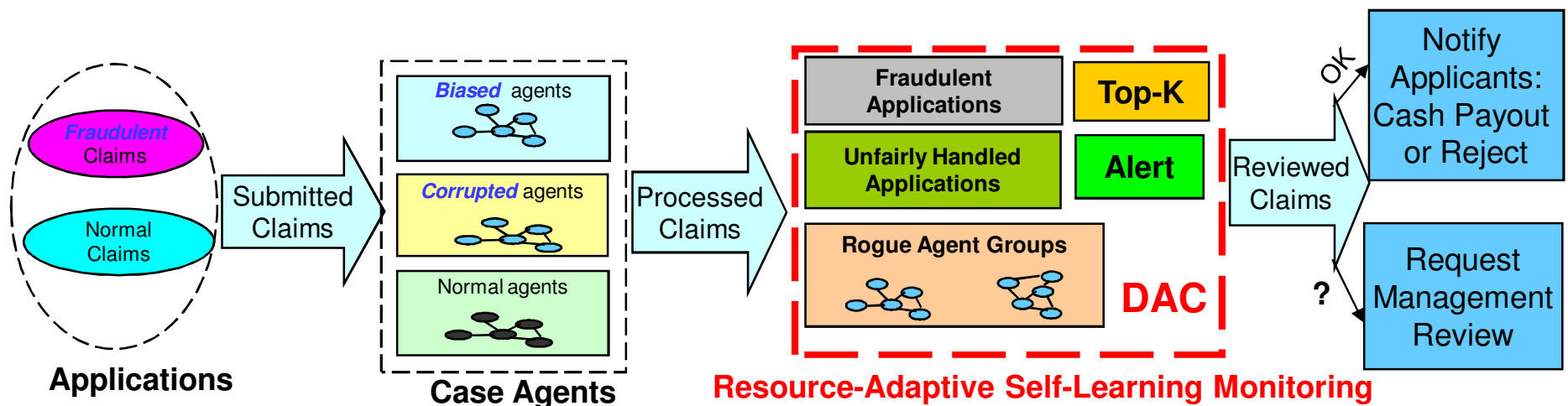
 Require a new model, ***data stream model***



Issues

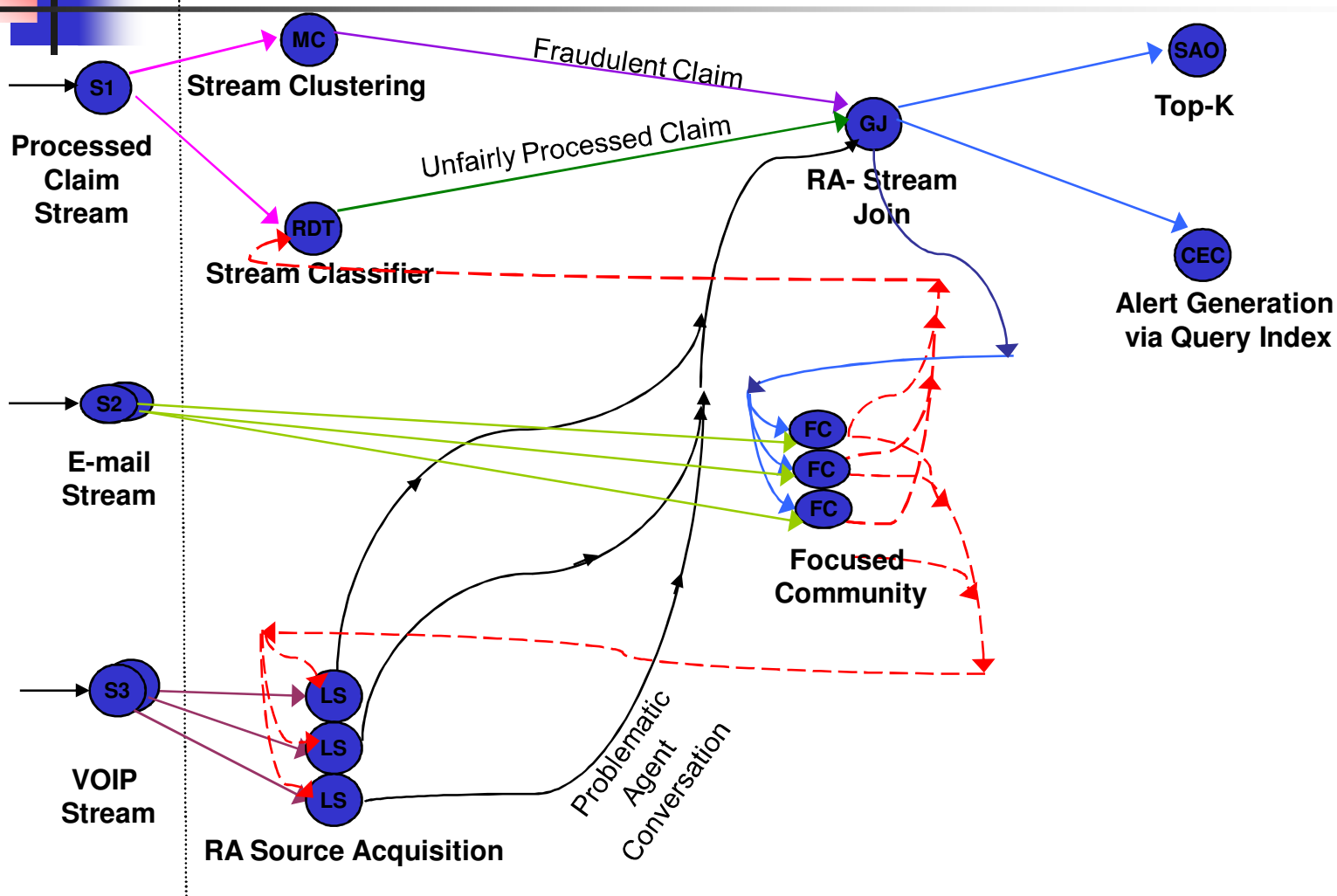
- Real-time: One pass
- Resource constraints
 - Limited memory and processing power
- Evolving stream characteristics
 - Temporal locality
 - New patterns vs outliers/anomalies
- Noisy data

Disaster Assistance Claim Monitoring System



K.L.Wu, P.S.Yu, et. al.: *Challenges and Experience in Prototyping a Multi-Modal Stream Analytic and Monitoring Application on System S*, VLDB07.

Disaster Assistance Claim Monitoring System





Stream Clustering

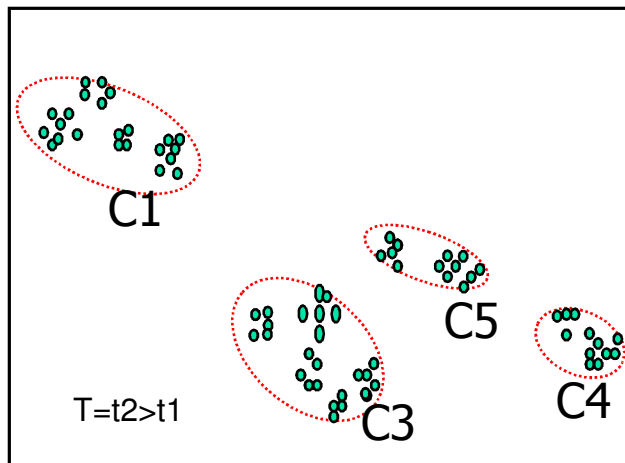
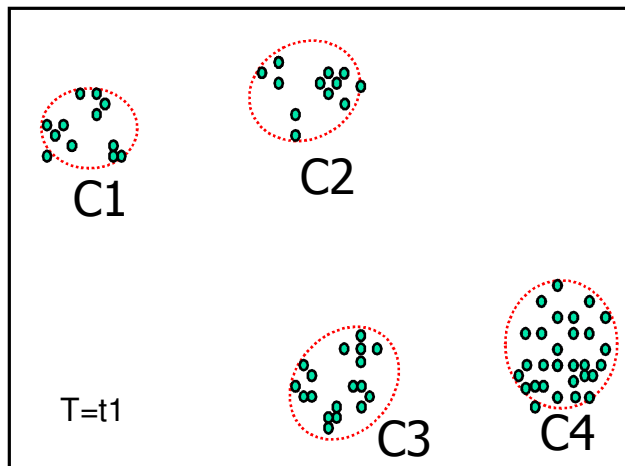
- Stream clustering not only requires as in previous approaches on clustering stream data
 - Achieving one-pass clustering over the data stream
 - Satisfying space and time constraints
- But also imposes challenges on
 - Capturing *temporal locality* of evolving data streams
 - Providing more flexible *exploratory capability* on clustering
 - Capability on analyzing cluster changes over time



CluStream Approach

- CluStream: a framework for clustering evolving data streams
 - OLAP like philosophy: process once & query many times
 - Divide the clustering process into online and offline components
 - Online component: summarizes statistics about the stream data
 - Offline component: answers various user queries based on the stored summary statistics

Stream Clustering





Micro-clustering

- Each point contains d dimensions with a time stamp
- A micro-cluster for n points is defined as a $(2d + 3)$ tuple

$$\left(\overline{CF2^x}, \overline{CF1^x}, CF2^t, CF1^t, n\right)$$

- $\overline{CF2^x}$ and $\overline{CF1^x}$ each corresponds to a vector of d entries
- For each dimension, the sum of the squares of the data values are maintained in $\overline{CF2^x}$
- For each dimension, the sum of the data values are maintained in $\overline{CF1^x}$
- The sum of the squares of the time stamps T_{i_1}, \dots, T_{i_n} are maintained in $CF2^t$
- The sum of the time stamps T_{i_1}, \dots, T_{i_n} are maintained in $CF1^t$

Empirical Results

Network intrusion data set

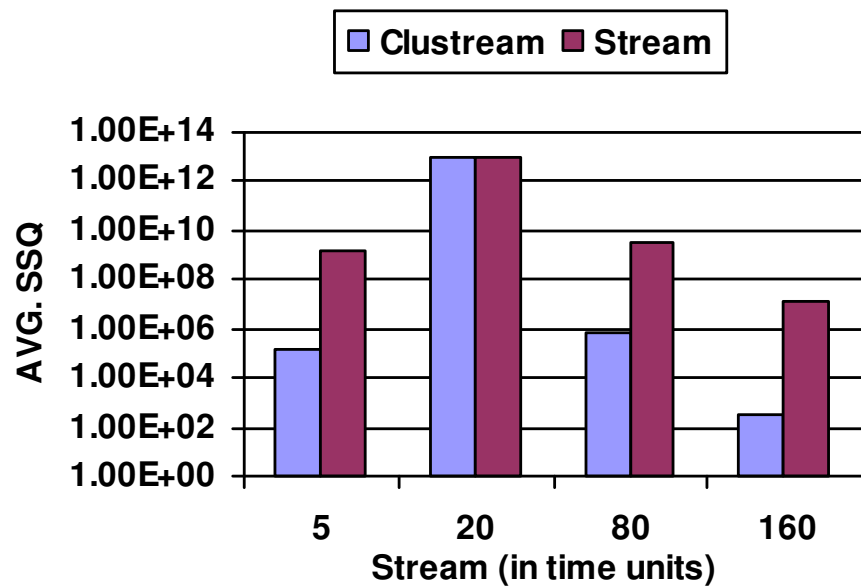


Figure 1. Quality test with horizon=1, stream_speed=2000, Net. Intrusion dataset

Charity donation dataset

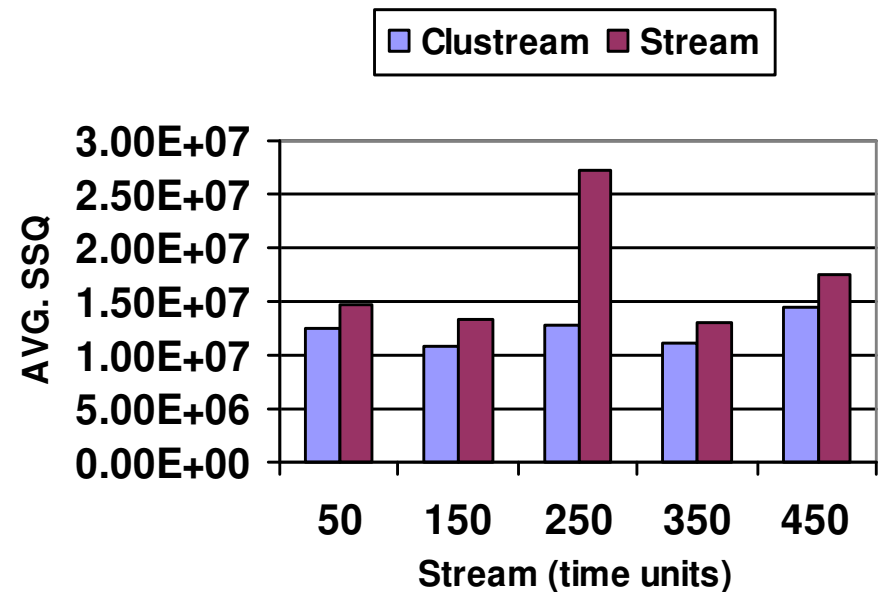


Figure 3. Quality test (stream_speed=200, horizon=4, Charitable donation dataset)

*Stream [O'Callaghan et al]

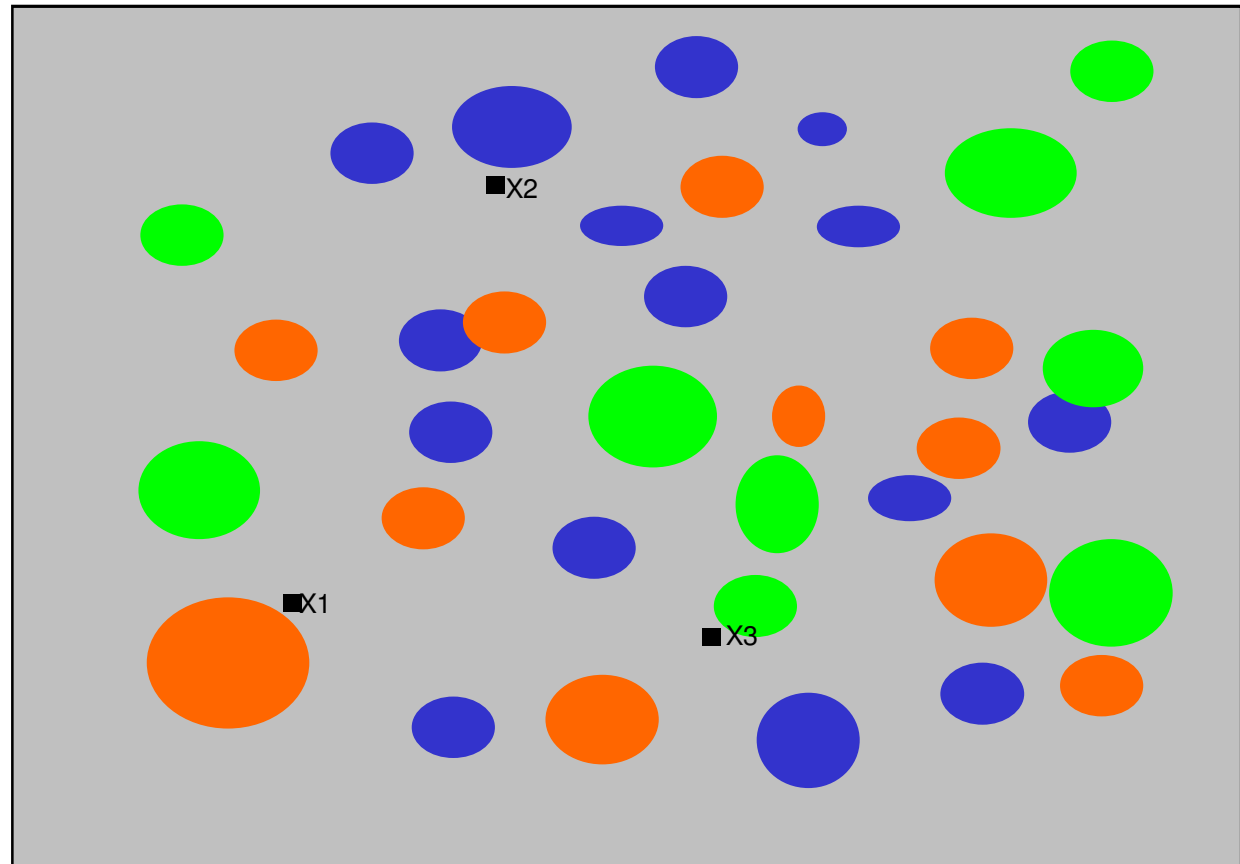
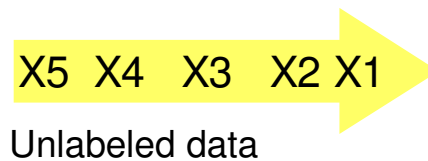
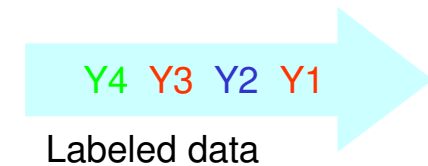


Challenges

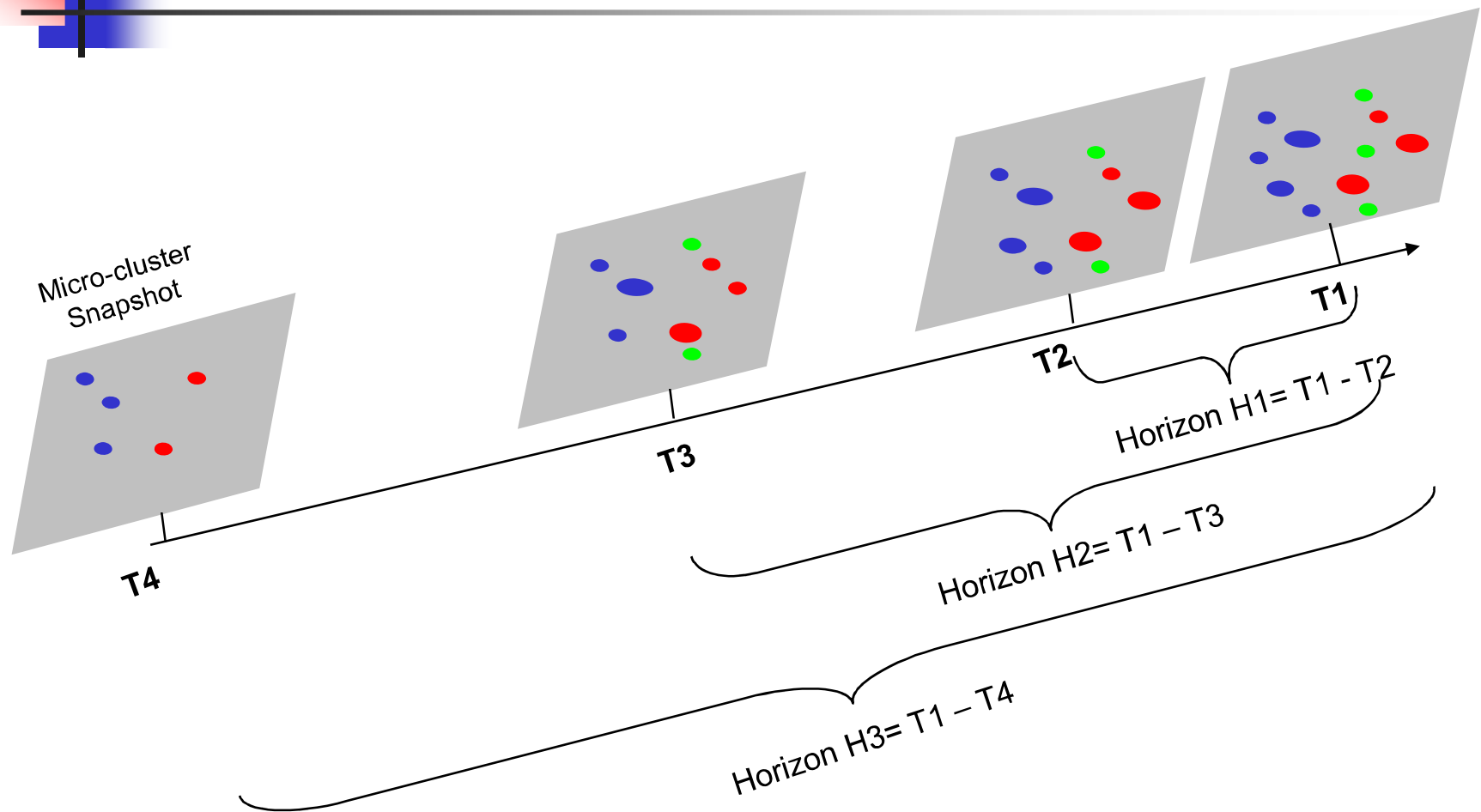
- With data continues to stream in, need to incrementally and rapidly update the model
 - Capture the time-evolving trends and patterns
 - Make time-critical predictions

Micro-cluster based On-demand Stream Classifications

Micro-Cluster snapshot between $(T_c - h, T_c)$ on labeled data



On-Demand Stream Classification





Horizon Selection

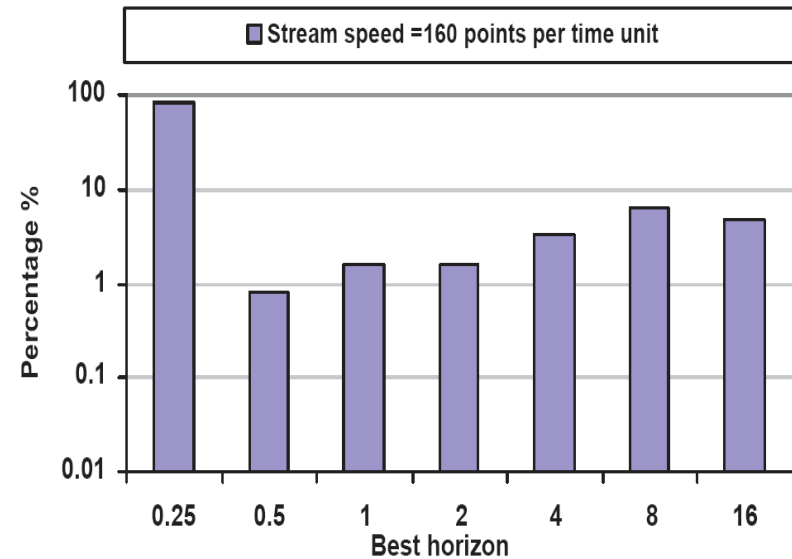
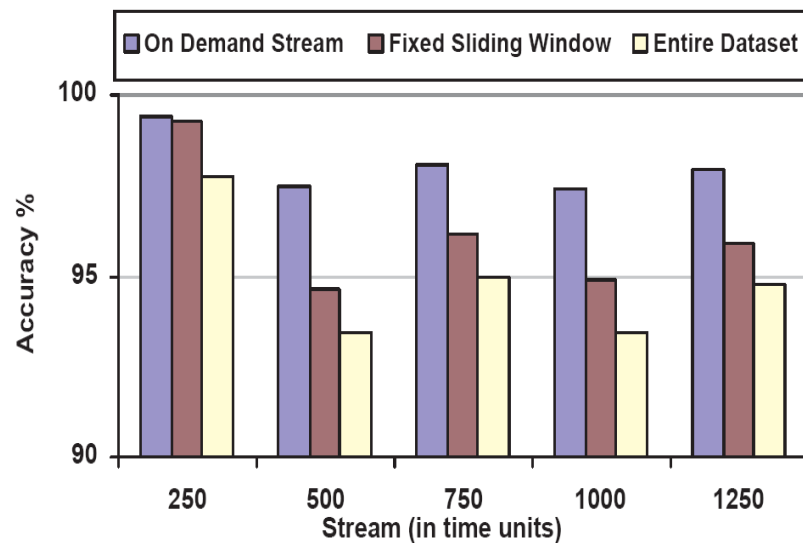
- Tradeoff
 - Too large: conflicting concepts
 - Too small: overfitting
- Optimal horizon is hard to determine as it varies over time
 - Slowly evolving period: large horizon
 - Fast evolving period: short horizon



Dynamic Horizon Selection

- In order to achieve the goal of optimal horizon choice, the incoming training data stream is divided into two parts
 - Select horizon
 - A small portion of the stream is used for the process of *horizon fitting*.
 - Use it to select the horizon that gives the best classification accuracy
 - Update micro-cluster information
 - The remaining majority of the training stream is used for accumulation of the pertinent statistics corresponding to the micro-clusters and class information.

Network Intrusion Dataset



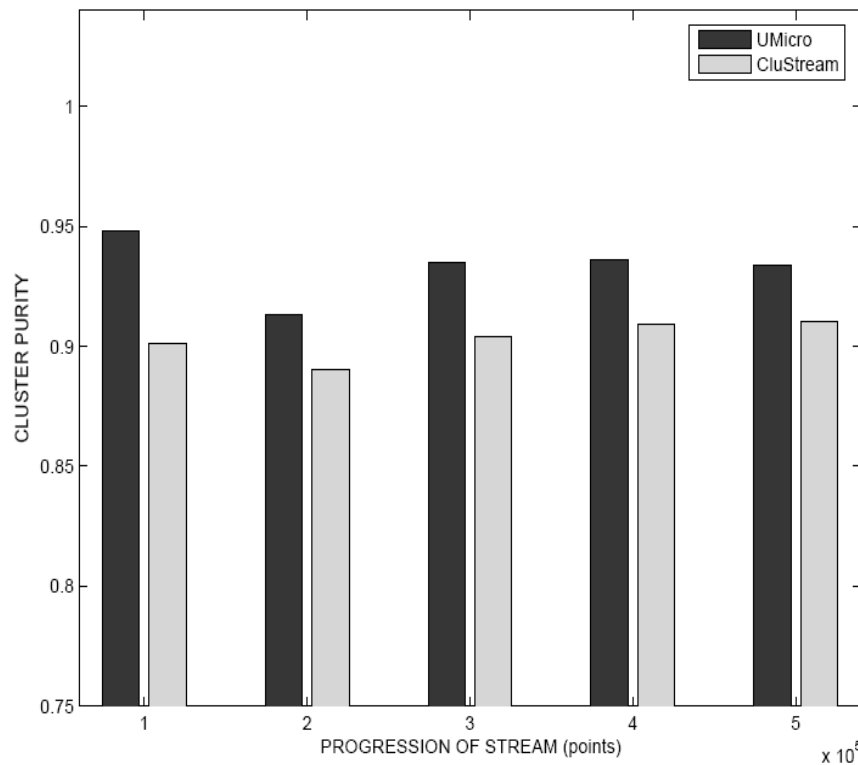
- Accuracy comparison (stream_speed=160)
- Distribution of the (smallest) best horizon



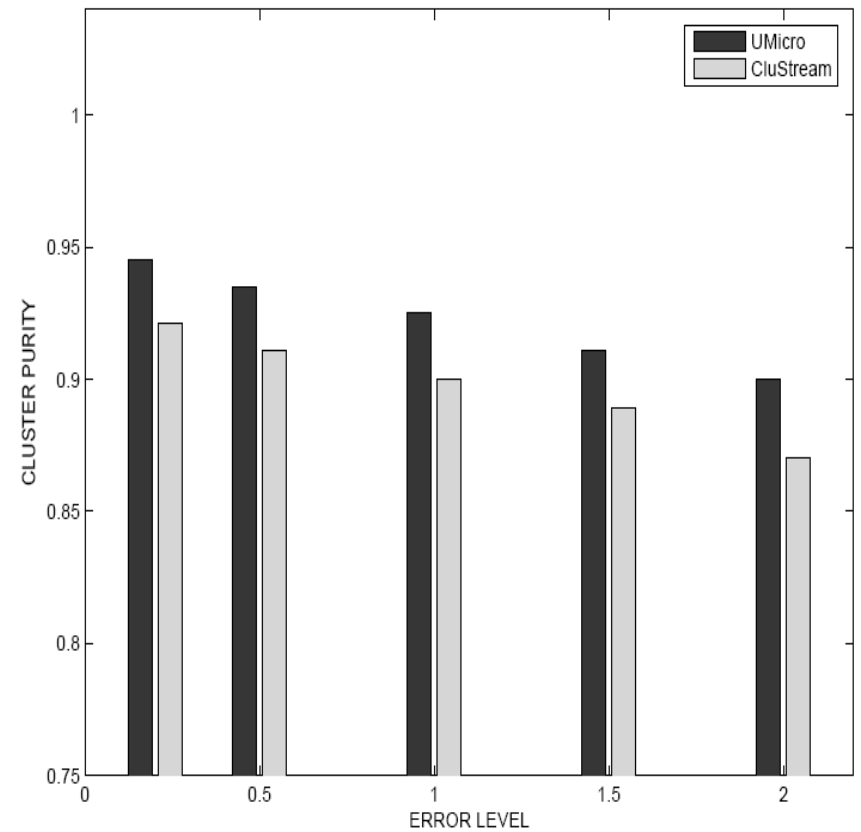
Uncertain Data

- Uncertain data has become ubiquitous because of new ways of collecting data.
- A large number of applications have recently been designed in the context of uncertain data.
- The use of uncertainty in the distance or similarity computations often can improve the quality of the results.

Network Intrusion Data Set

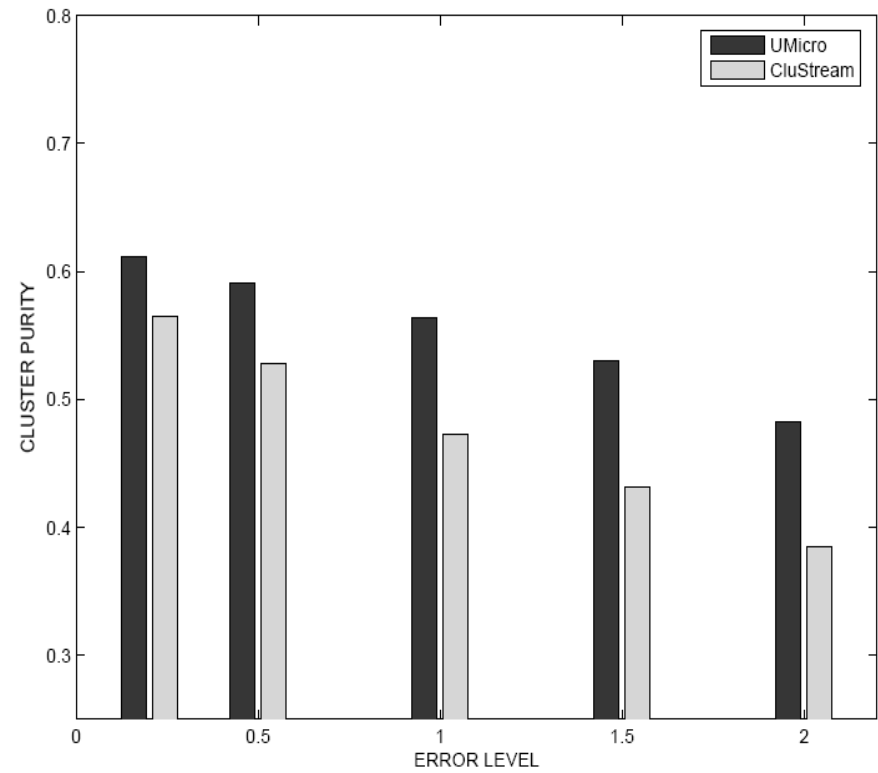
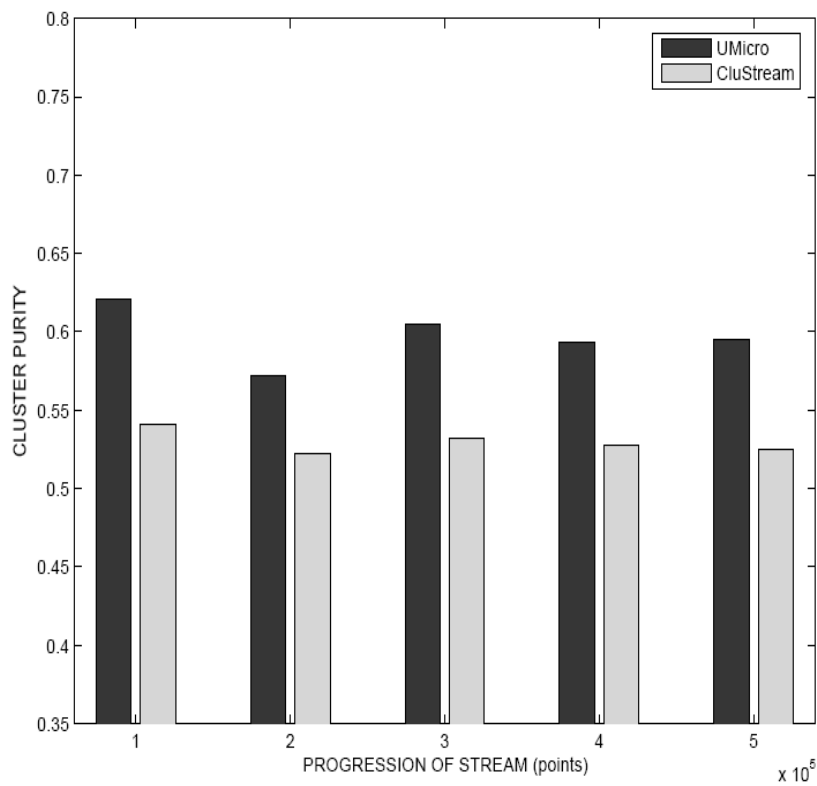


Network Intrusion Data Set



Network Intrusion Data Set

Forest Cover Data Set



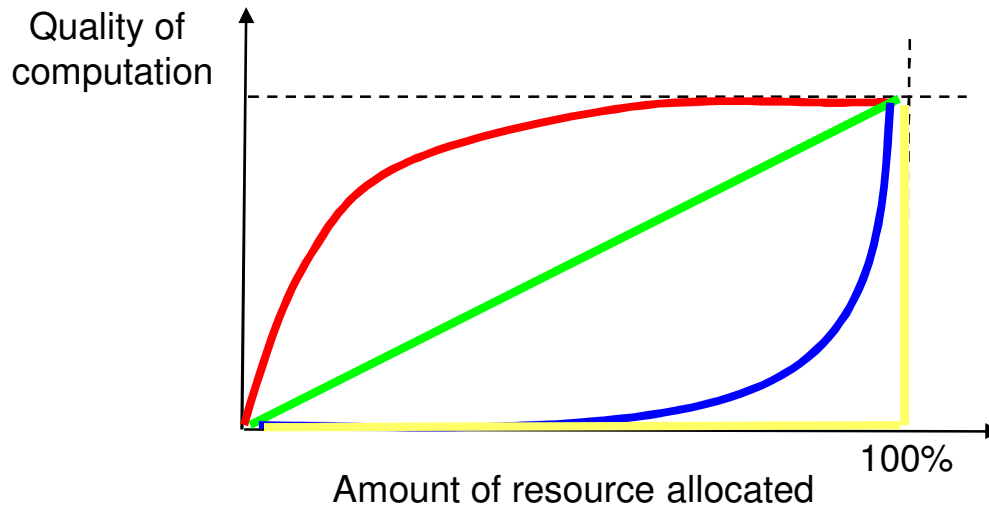


Resource-adaptive computation

- In a data stream environment,
 - Input stream rate is determined by external sources
 - Resource allocation depends upon other competing tasks
- A given computation task needs to
 - Be aware of (or monitor) the resource allocated
 - Make a best effort computation to perform the task under the resource and time constraints
- Trade-off between
 - Quality of computation
 - Resource consumption

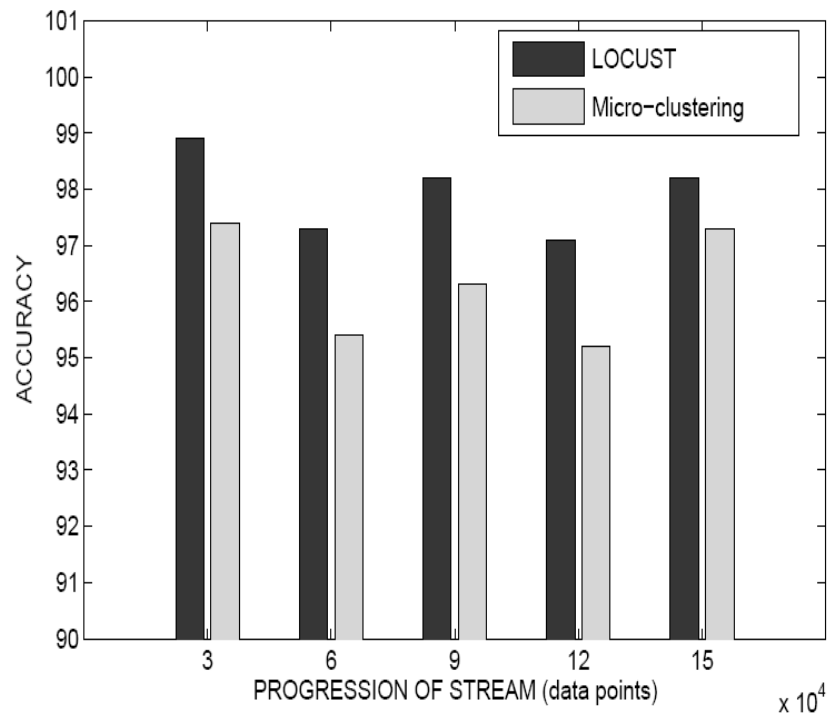
Challenge on Resource-Adaptive Computation

- Behavior of computation algorithms

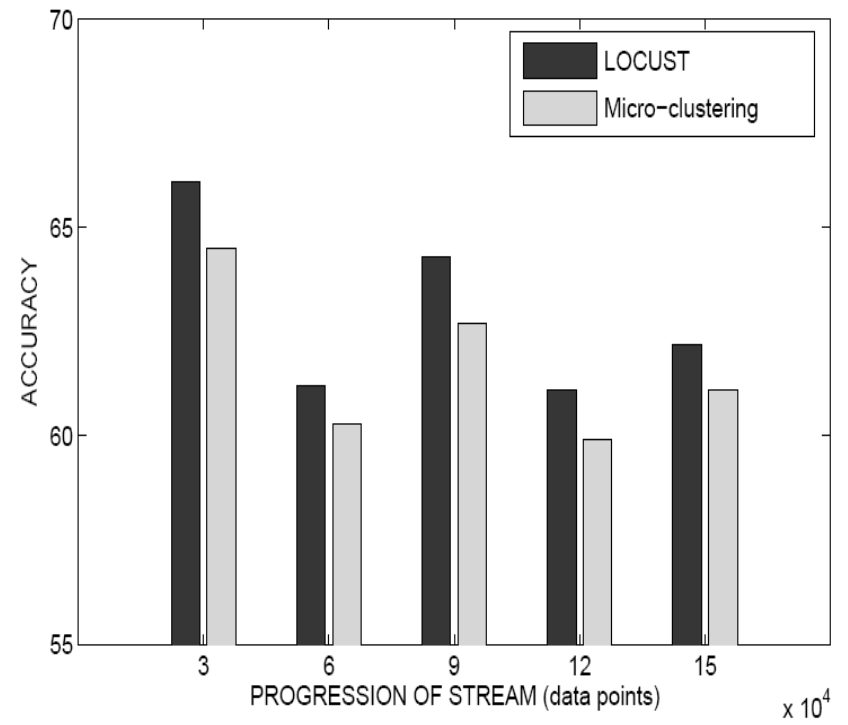


- Goal is to design super-linear algorithm that can generate 80% of results with 20% of the resource

Accuracy Results

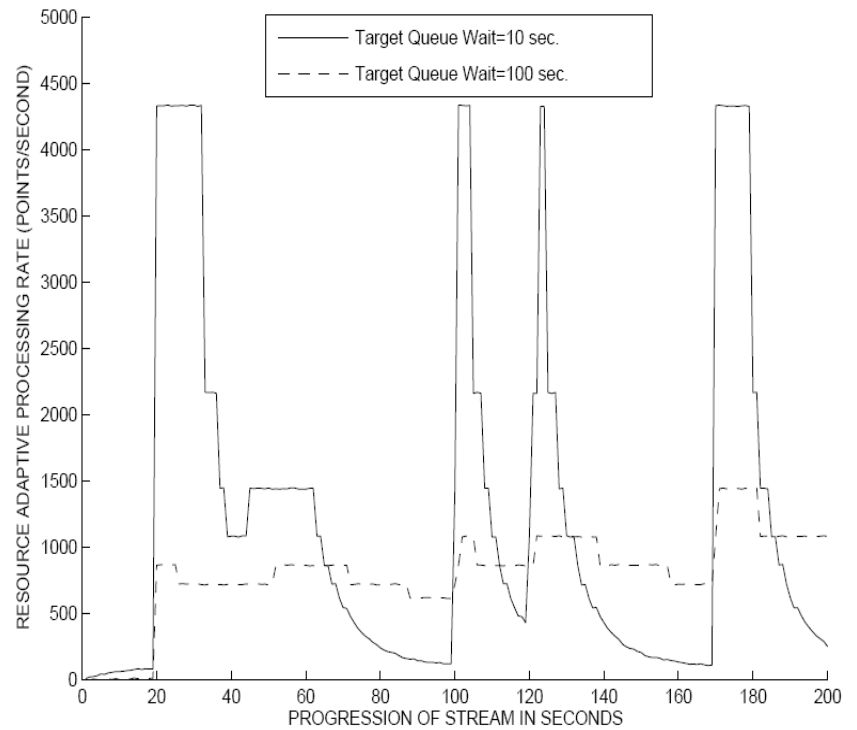
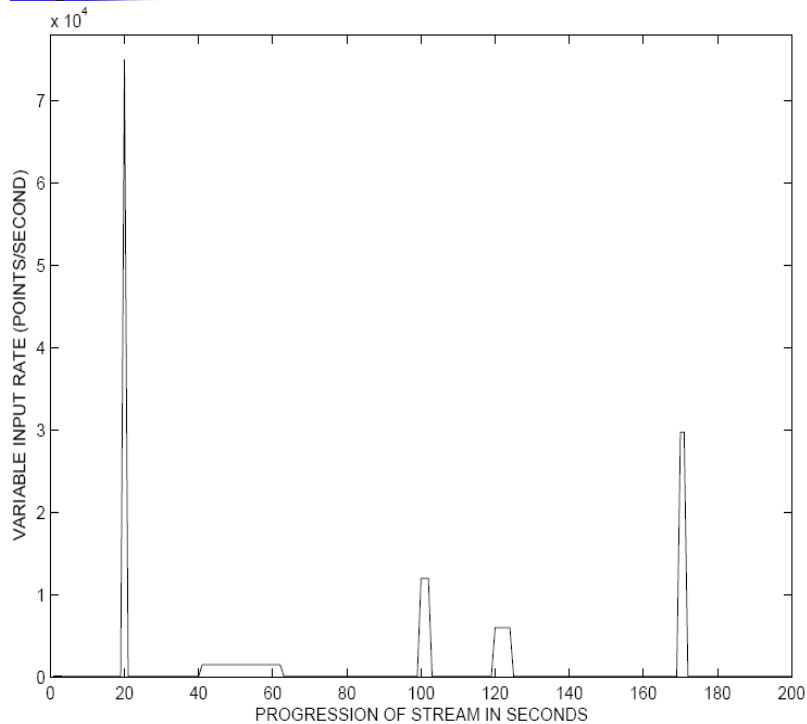


Network Intrusion Data



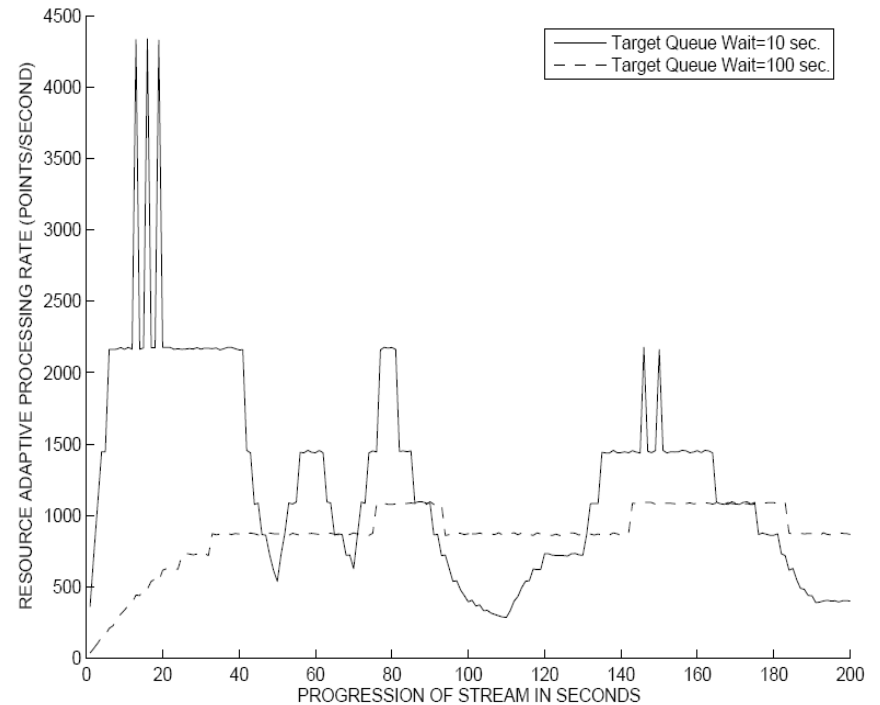
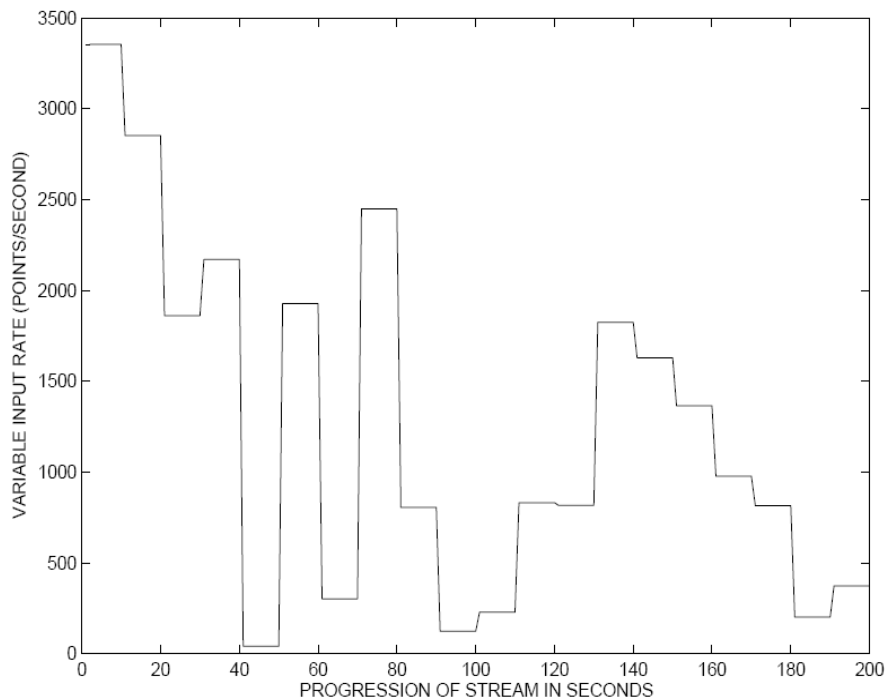
Forest Cover Data

Adjustments with Input Streams



Network Intrusion Data

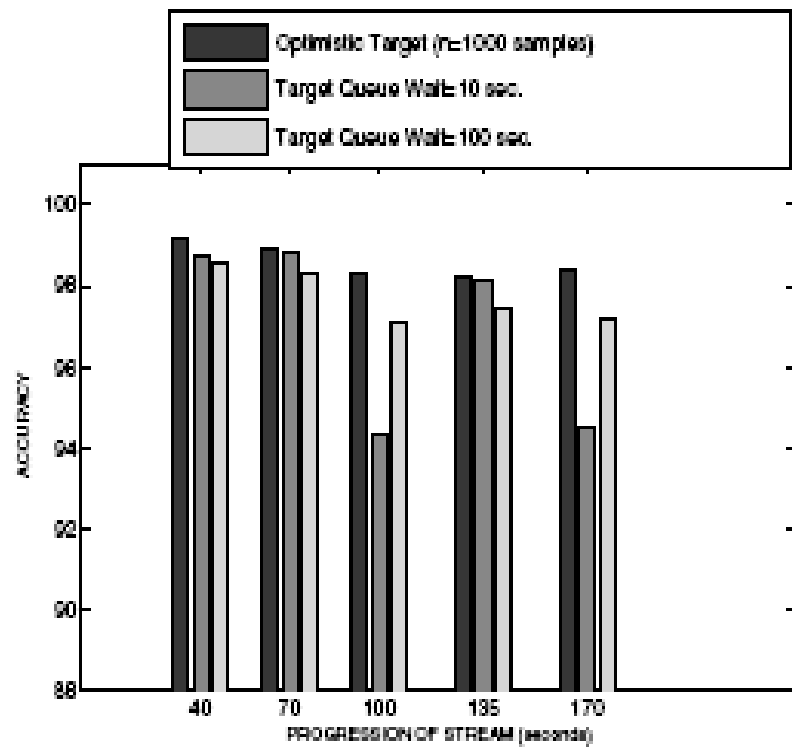
Adjustments with Input Streams



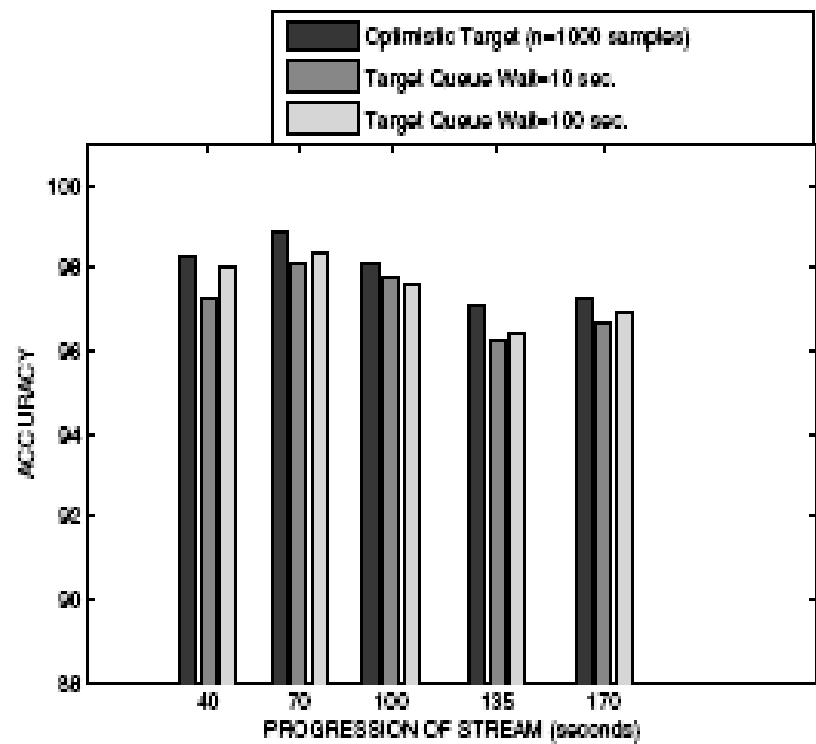
Network Intrusion Data



Accuracy



(a) Effectiveness for Network Intrusion (A)



(b) Effectiveness for Network Intrusion (B)



Other Topics of Interests

- Graph mining
 - Feature compositions
- Graph OLAP
- Graph indexing
- Trajectory indexing



Summary

- Stream Clustering
- Stream Classification
- Uncertain data
- Resource Adaptive computation



References on Stream Mining

- C. Aggarwal, P.S. Yu, " A Framework for Clustering Uncertain Data Streams ", ICDE08.
- C. Aggarwal, P.S. Yu, "LOCUST: An Online Analytical Processing Framework for High Dimensional Classification of Data Streams", ICDE08.
- C. Aggarwal, J. Han, J. Wang, P.S. Yu, "A Framework for Clustering Evolving Data Streams", VLDB03.
- K.L.Wu, P.S.Yu, et. al.: *Challenges and Experience in Prototyping a Multi-Modal Stream Analytic and Monitoring Application on System S*, VLDB07.



References on Graphs

- X. Yan, H. Cheng, J. Han, P.S. Yu, "Mining Significant Graph Patterns by Scalable Leap Search", ACM SIGMOD, June 2008.
- C. Chen, X. Yan, F. Zhu, J. Han, P.S. Yu, "Towards Online Analytical Processing on Graphs", IEEE Intl. Conf. on Data Mining, Dec. 2008.
- X. Yan, J. Han, P.S. Yu, "Graph Indexing based on Discriminative Frequent Structure Analysis", ACM Trans. on Database Systems, Vol. 30, No. 4, Dec. 2005.
- M. Vlachos, A. Anagnostopoulos, M. Hadjieleftheriou, E. Keogh, P.S. Yu, "Global Distance-based Segmentation of Trajectories", ACM SIGKDD, Aug. 2006.