Mining Data Streams

PHILIP S. YU (PSYU@CS.UIC.EDU)

WEXLER CHAIR IN INFORMATION TECHNOLOGY

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 or human beings, various data streams are generated at high rate

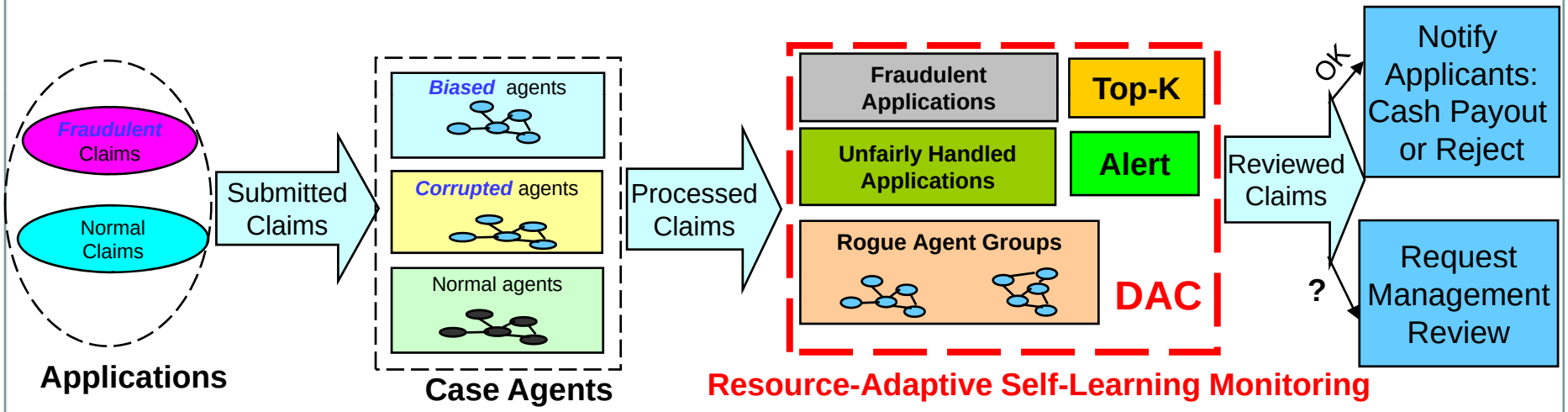
Real-time Stream Applications

- Trade surveillance for security fraud and money laundering
- Bio-surveillance for terrorist attacks
- Sensor network for monitoring intelligent oil wells, manufacturing plants, RFID products, etc
- Network monitoring for intrusion detection
- Emergency room patient monitoring
- Web related applications
  - Click stream mining for real-time personalized recommendations
  - Text stream mining for topic detection

Issues

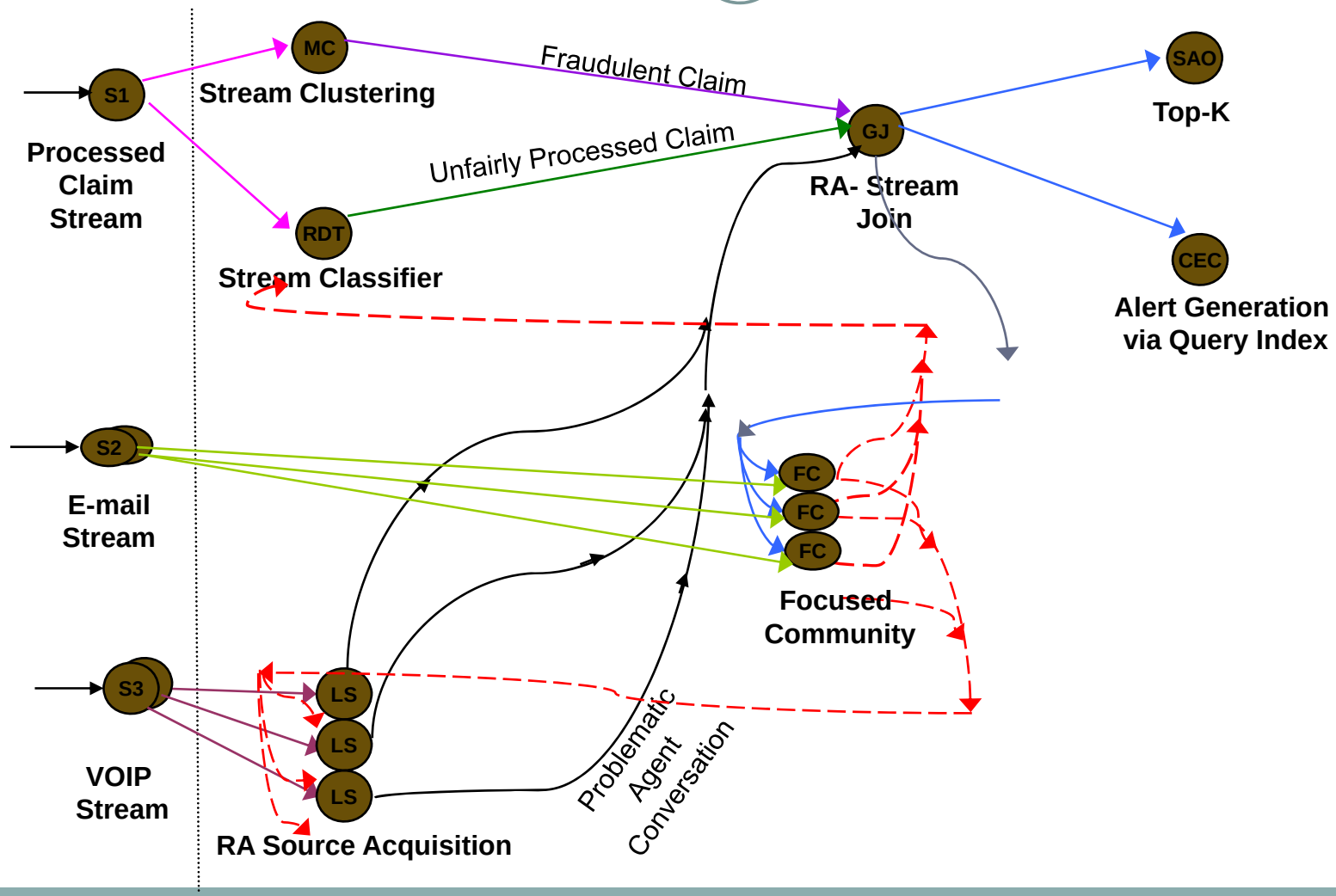
- Real-time: One pass
- Resource constraints
  - Limited memory and processing power
- Evolving stream characteristics
  - Temporal locality
  - New patterns vs outliers/anomalies
- Noisy data

Disaster Assistance Claim Monitoring System



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

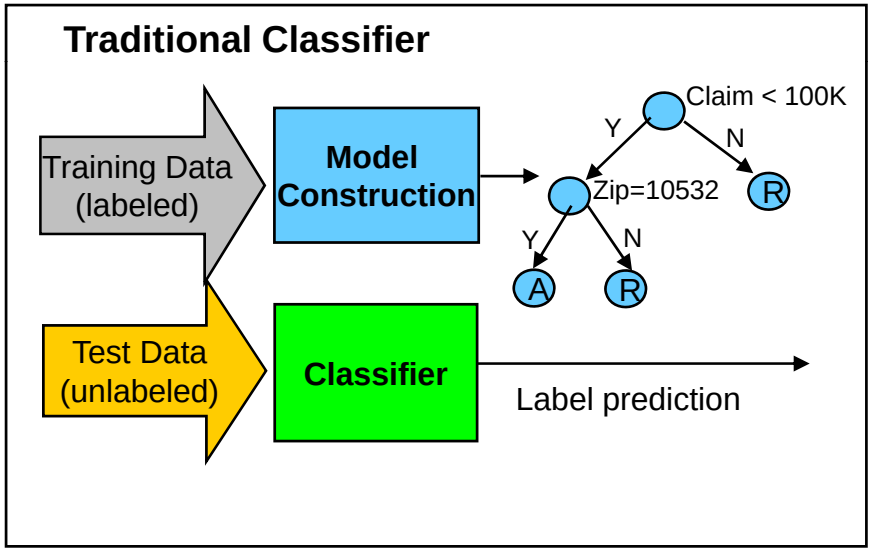
Disaster Assistance Claim Monitoring System

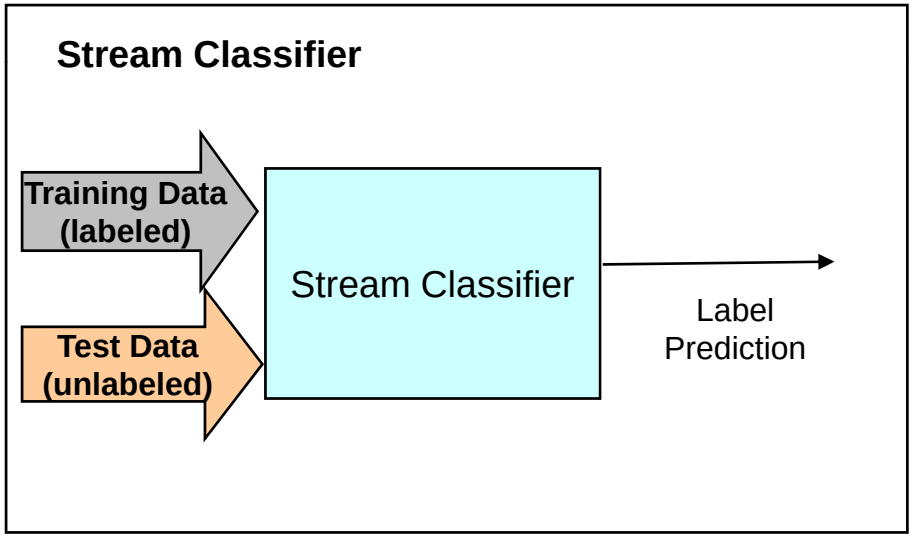


Stream Classification

Resource Adaptive Computation

Stream Classification



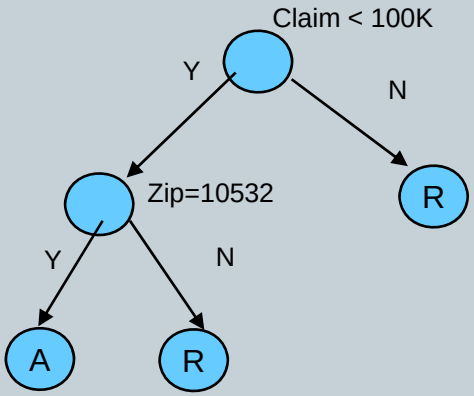


High Dimensional Classification

- The classification problem faces a number of unique problems in the high dimensional case
  - This is because of the exponential number of attribute combinations which can be related to the class variable

Locality based Classifiers

- Classifiers that use locality in one or more subspaces in order to define the classification behavior
  - Nearest Neighbor Classifier: Locality in full dimensional space
  - Decision Tree Classifier: The decision path along a tree typically represents a subspace locality for classifying the test instance



  - Rule based Classifier: the antecedent of a rule represents a subspace locality

Observations

- Subspace Classifiers build the entire classifier a-priori
  - Leads to inflexibility with different kinds of test instances
- Decision Tree:
  - A split at a higher level of the tree may work well with some test instance, but it may not work well with another test instance
- Rule based classifier:
  - The small number of rules pre-selected (from the exponential number of possibilities) may not work well for a particular test instance

Key Solutions

- Choose a subspace dynamically at the time of classification

- Key challenges
  - How to choose from the large number of possibilities efficiently?
  - How to handle the varying rate of incoming test instances?
    - Need quick adjustments

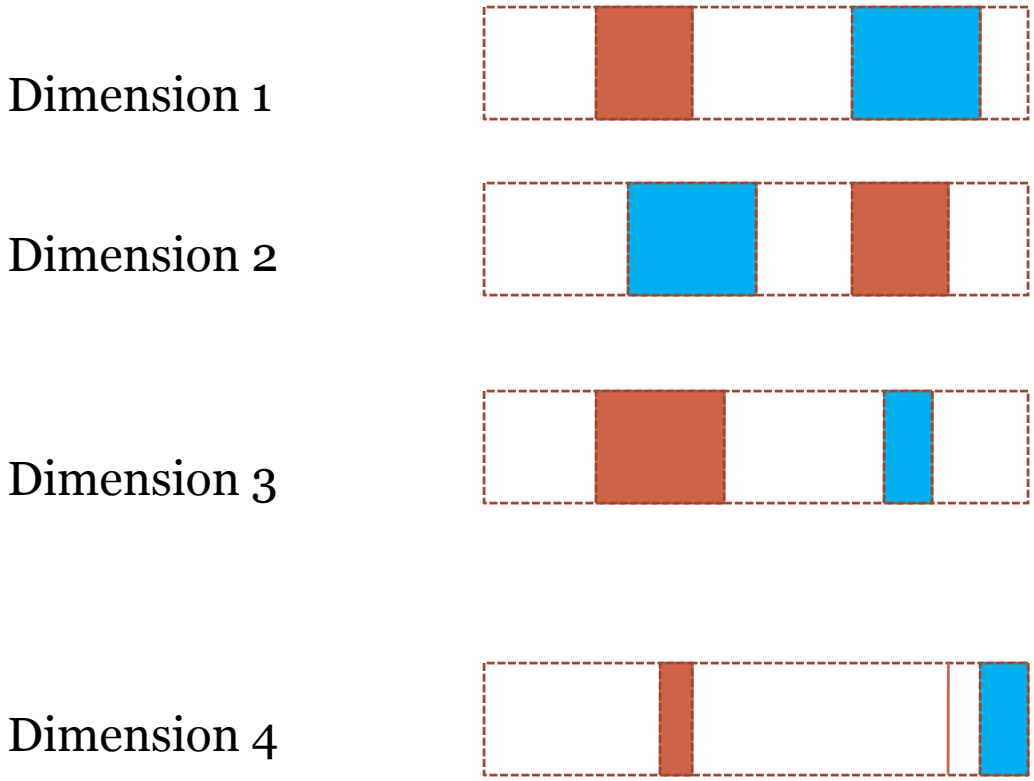
Broad Approach

- Construct an intermediate data structure on the training data
  - OLAP type approach
- Data structure should be flexible enough to dynamically construct local subspaces for each test instance efficiently
- Use multiple local subspace samples in order to perform the classification
  - Robustness of classification depends on number of samples (adjustable)

Intermediate Data Structure

- The intermediate data structure used is an inverted histogram representation
- Construct a histogram along each dimension using Φ- equi-depth ranges
  - Creates Φ inverted list along each dimension

Subspace Selection



S1={2, 3}

S2={1, 2, 4}

S3={3,4}

S1={1,4}

S2={2,4}

S3={1,2,3}

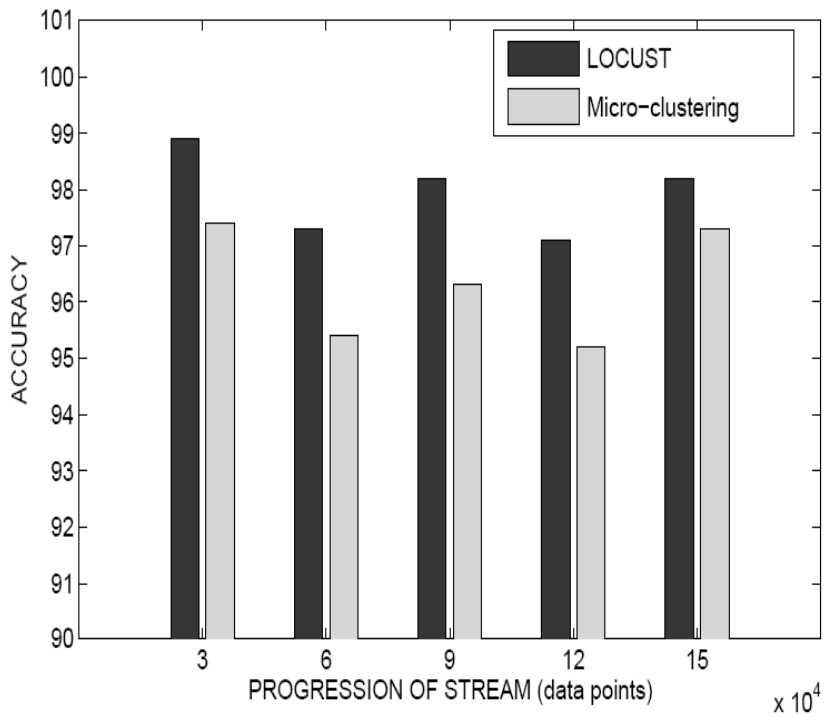
Dynamic Subspace Construction

- A sample of γ-inverted lists along the different dimensions creates a subspace of dimensionality of γ
- Majority of subspace samples provides classification
- Use class variable in the sampling process
  - Maintain the class entropy of each inverted list and sample as a function of the class entropy
  - Greater bias – faster convergence, but local optima

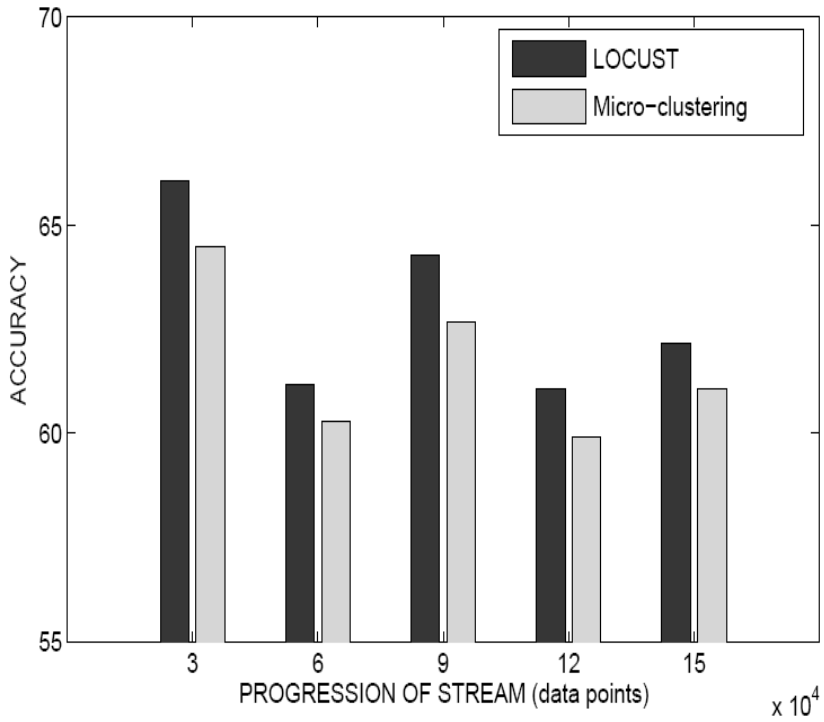
Regulation of Processing Speed

- The rate of incoming test instances cannot be controlled, therefore
  - Processing needs to be adjusted
- The processing speed is controlled by
  - Number of subspace sampled (n_s) used in a particular iteration
- Solution:
  - Maintain a queue of test instances to be processed, and
  - Aim for a constant queue waiting time

Accuracy Results

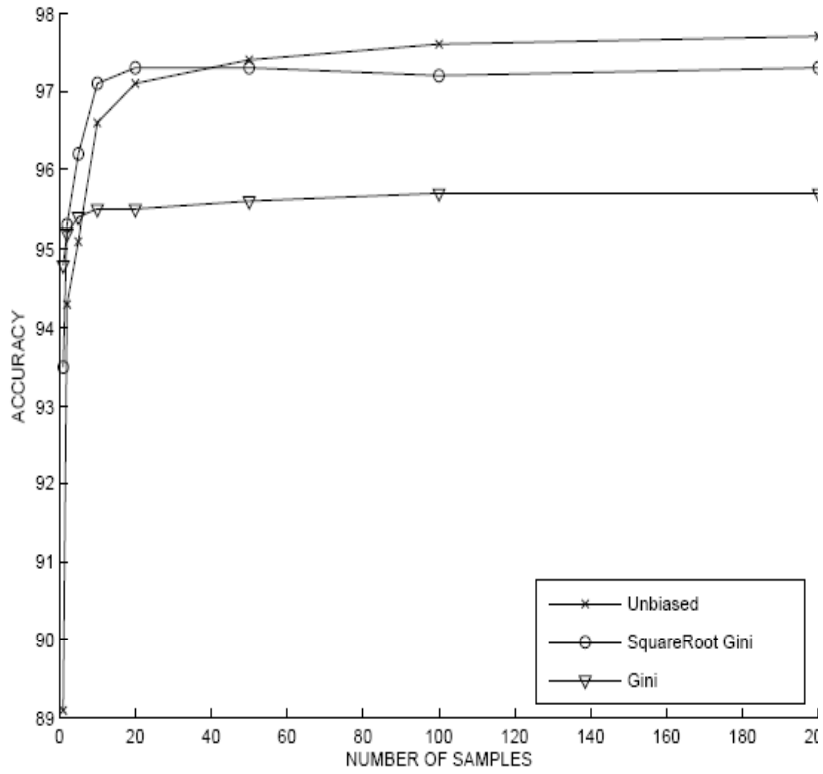


Network Intrusion Data

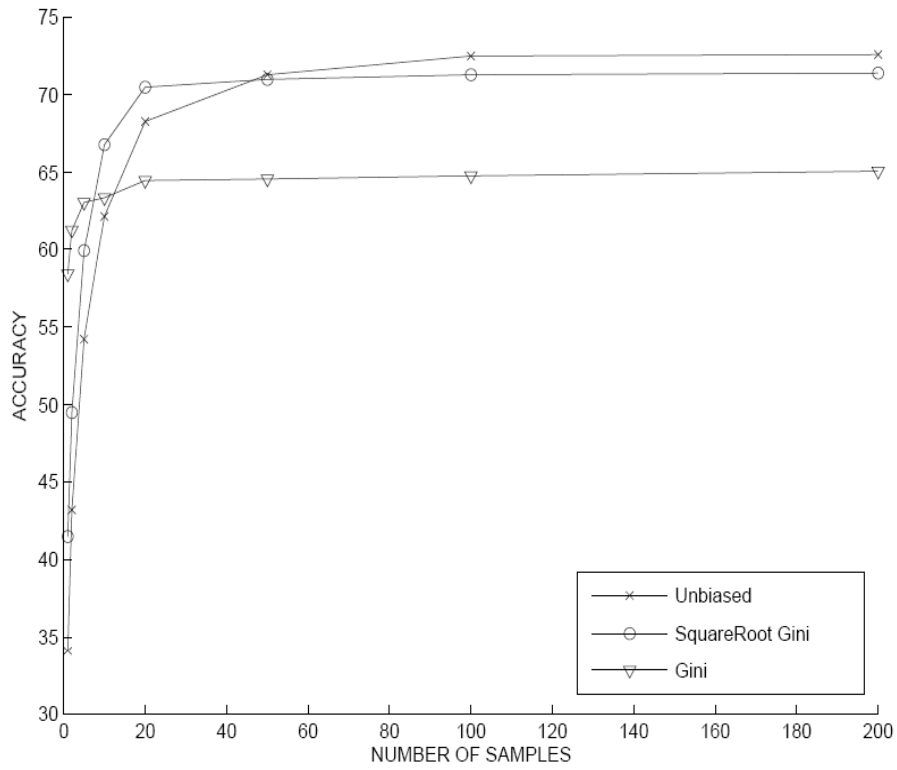


Forest Cover Data

Accuracy with Number of Samples

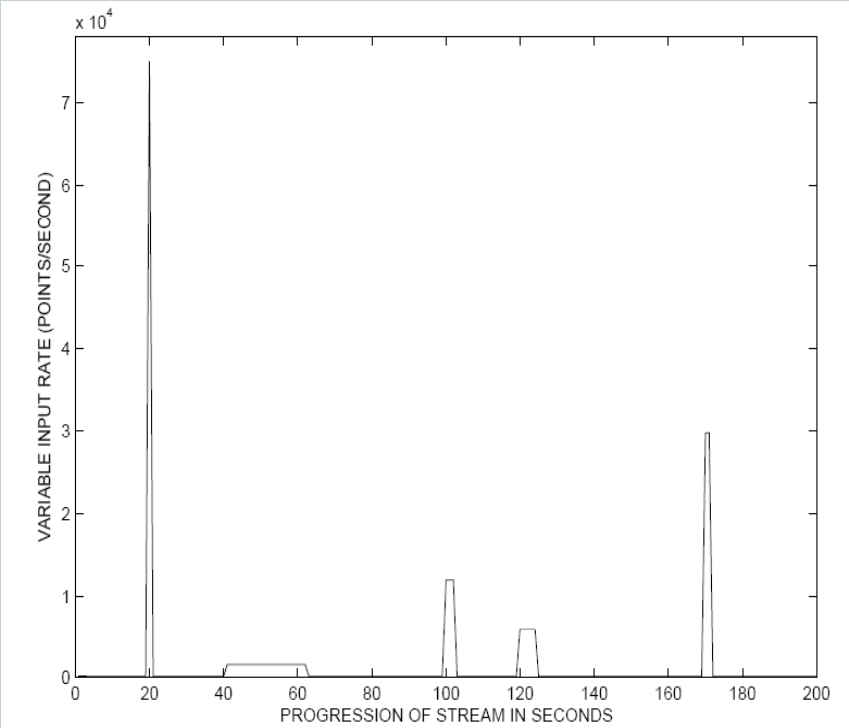


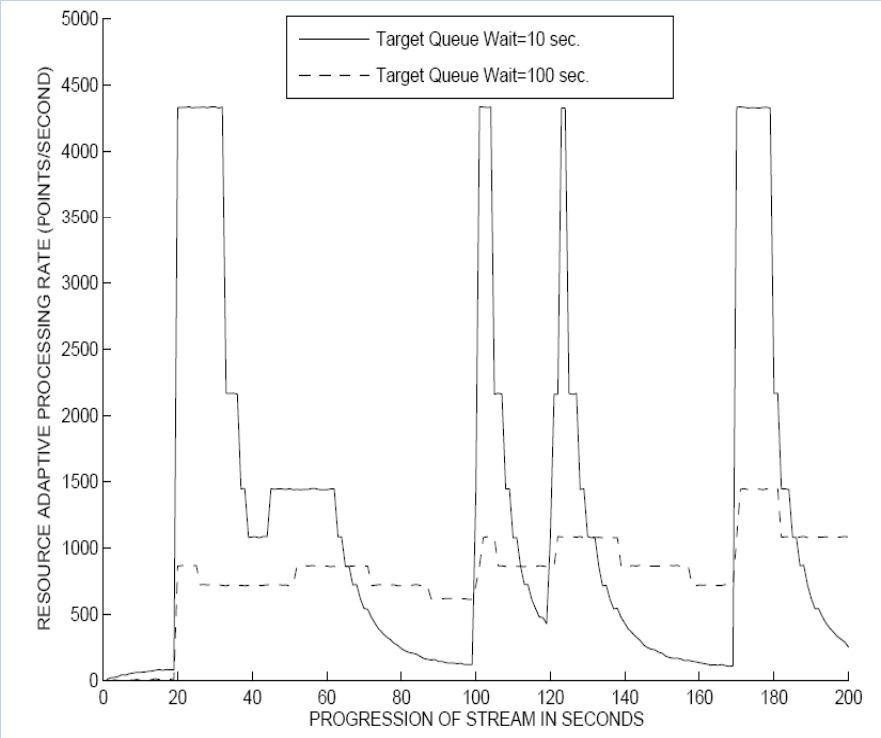
Network Intrusion Data



Forest Cover Data

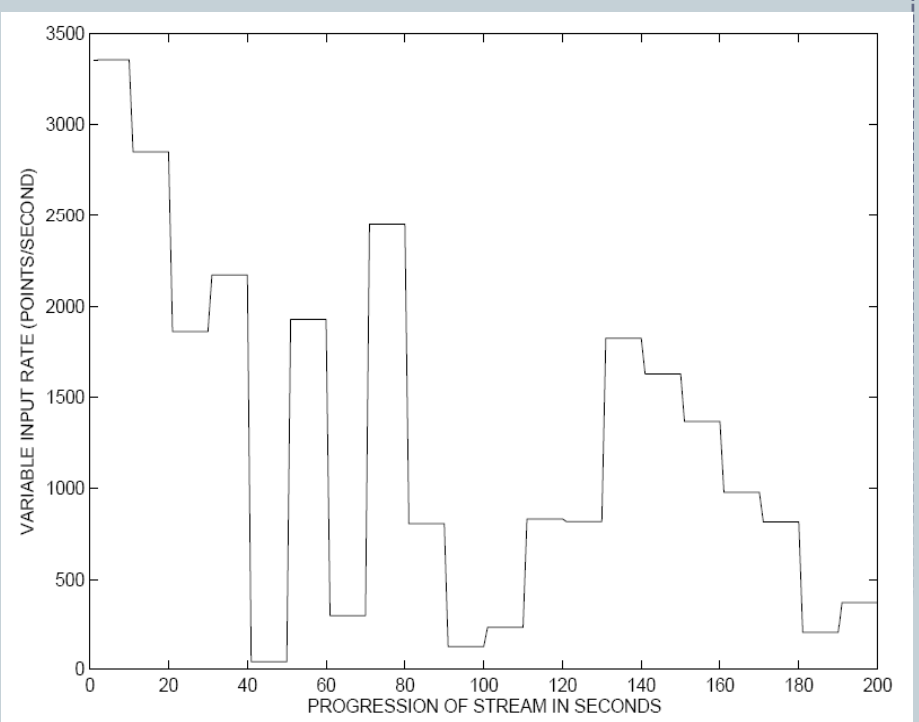
Adjustments with Input Streams

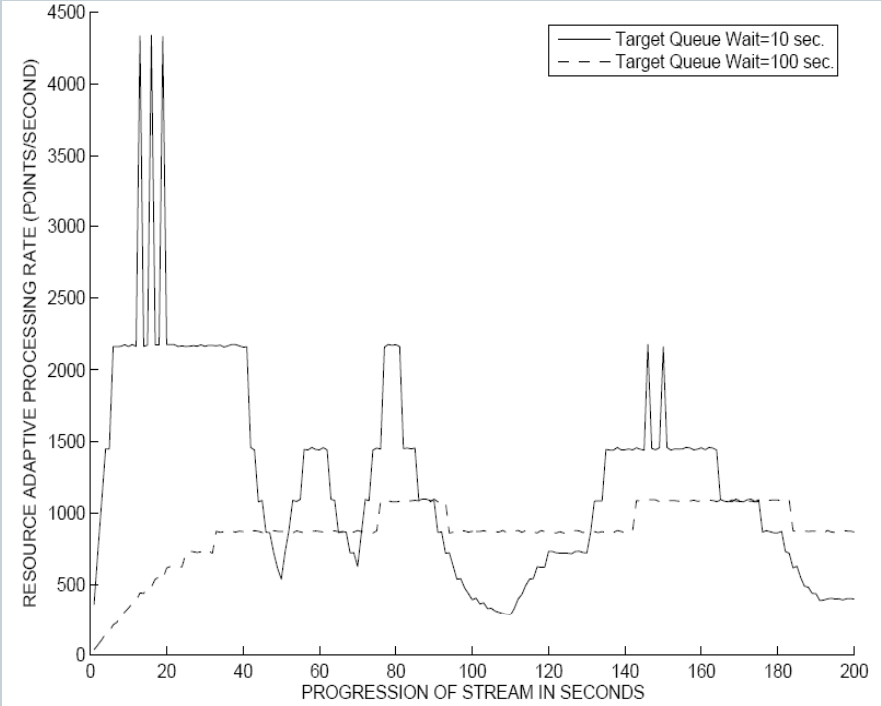




Network Intrusion Data

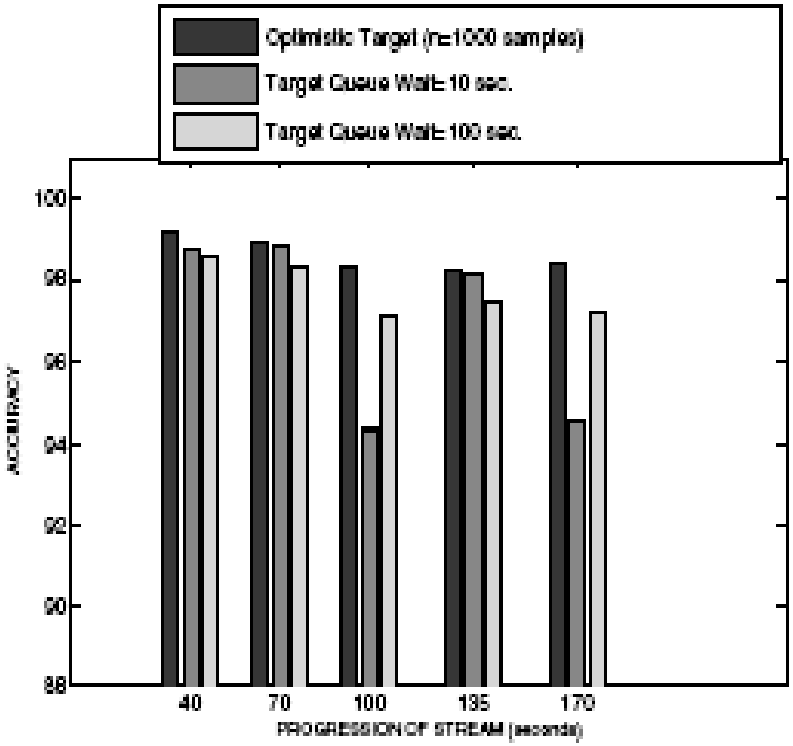
Adjustments with Input Streams



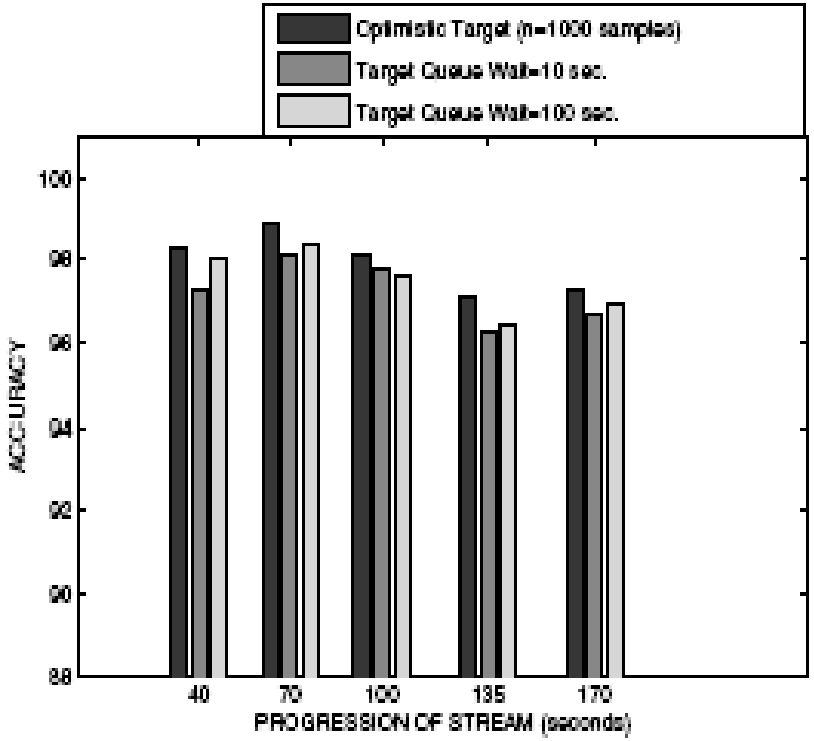


Network Intrusion Data

Accuracy



(a) Effectiveness for Network Intrusion (A)



(b) Effectiveness for Network Intrusion (B)

Conclusions and Summary

- Method for high dimensional classification of streams
- Adaptive method which can adjust with the speed of the stream
- Provides robust classification

References

- C. Aggarwal, P.S. Yu, "LOCUST: An Online Analytical Processing Framework for High Dimensional Classification of Data Streams", ICDE08.
- C. Aggarwal, P.S. Yu, " A Framework for Clustering Uncertain 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.

(http://www.informatik.uni-trier.de/~ley/db/indices/a-tree/y/Yu:Philip_S=.html)