Self-Regulating Stream Processing: Limitations and Opportunities

Avrilia Floratou
Ashvin Agrawal

Microsoft CISL
Streaming Systems Landscape

Spark Streaming

STORM

Heron

Samza

DataTorrent

Confluent
Self-Regulating Streaming Systems

Manual, time-consuming and error-prone task of tuning various system knobs to achieve SLOs

Maintenance of SLOs in the face of unpredictable load variation and hardware or software performance degradation

Self-Regulating Streaming Systems

- Self-tuning
- Self-stabilizing
- Self-healing
Self-Regulating Streaming Systems

Self-tuning
- Various tuning knobs
- Time consuming tuning phase
- The system should take as input an SLO and automatically configure the knobs.

Self-stabilizing
- Streaming applications are long-running
- Load variations are observed
- The system should react to external shocks and automatically reconfigure itself.

Self-healing
- System performance can be affected by hardware or software delivering degraded quality of service
- The system should identify internal faults and attempt to recover from them.
The First Step: Dhalion

Symptom Detector 1
Symptom Detector 2
Symptom Detector 3
Symptom Detector N

Symptom Detection

Diagnoser 1
Diagnoser 2
Diagnoser M

Diagnosis Generation

Diagnosis 1
Diagnosis 2
Diagnosis M

Resolver 1
Resolver 2
Resolver M

Resolver Selection

Resolver Invocation

Metrics

Tomorrow at session Stream Processing 1
Dhalion Challenges

Dhalion assumes a user that can deeply understand and accurately model the system behavior.

- Diagnosing the problem
- Selecting an action
- Evaluating an action
- Determining similar state-action pairs
Example: Diagnosing the Problem

Given a set of symptoms how can we determine the cause of the symptom?

Existence of Backpressure

The user must be aware of all the possible causes of a symptom in order to perform a successful diagnosis.

Possible causes: Data skew, slow hosts, underprovisioning, network partitioning etc.
When an action is selected, Dhalion checks whether the action has been blacklisted for a similar state of the application. In this case, the action is not executed.

\[ [5, \text{ScaleDown}(4)] \neq [6, \text{ScaleDown}(4)] \]
Example: Evaluating an Action

Scale up resources in order to meet a performance requirement (e.g., latency SLO)

Evaluation Criterion

Does the new latency meet the performance SLO?

Scale Up

Stage 1

Stage 2

The action was correct but did not result in a performance improvement
The Role of Machine Learning

Machine learning has been used in various domains to automate several tasks by limiting user intervention and even make better decisions than human users.

Can we use machine learning in the context of self-regulating steaming systems?

- Diagnosing the problem
- Selecting an action
- Determining similar state-action pairs
Supervised Learning

Diagnosing the problem

Selecting an action

• Collect/Generate training data
• Determine the feature set
• Incorporate supervised learning in Dhalion
Supervised Learning

How to incorporate supervised learning in Dhalion?
• Exploration vs exploitation cost
• Determining the rewards function is challenging
Conclusions

Both reinforcement and supervised learning have their pros and cons.

Machine learning introduces a new set of challenges.

Although machine learning can potentially address some of the major problems that users face, some challenges such as evaluating the actions taken, cannot directly be addressed by machine learning techniques.