Machine Learning for Research Networks

Anna Giannakou

Lawrence Berkeley National Lab

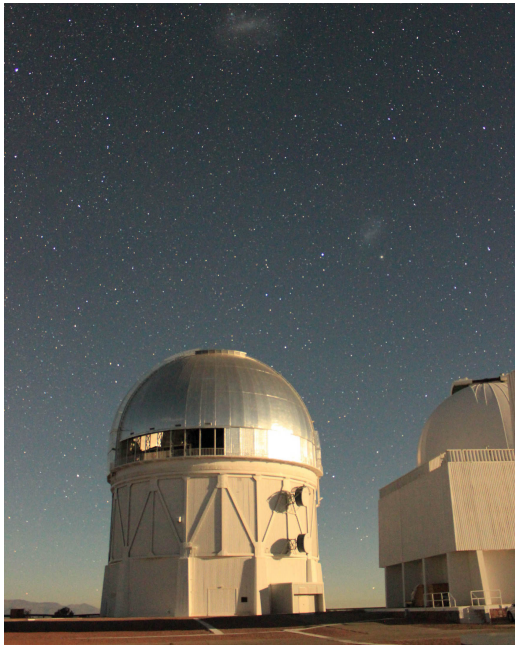
agiannakou@lbl.gov

BERKELEY LAB

Network Performance Impacts Scientific Discovery

Scientific discovery depends on large WAN transfers

Example: Telescope data must be transferred and analyzed in under 24h to avoid backlogs

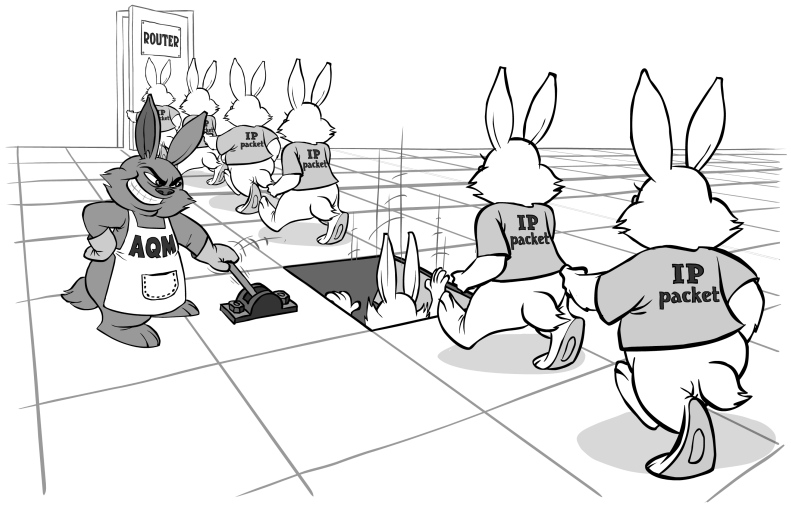


Transfers experience performance degradation events

BERKELEY LAB

Types of Performance Degradation Events

- Packet loss:
  - Path properties
  - End host configuration
- Abnormal transfers:
  - Data exfiltration attempts
  - User errors

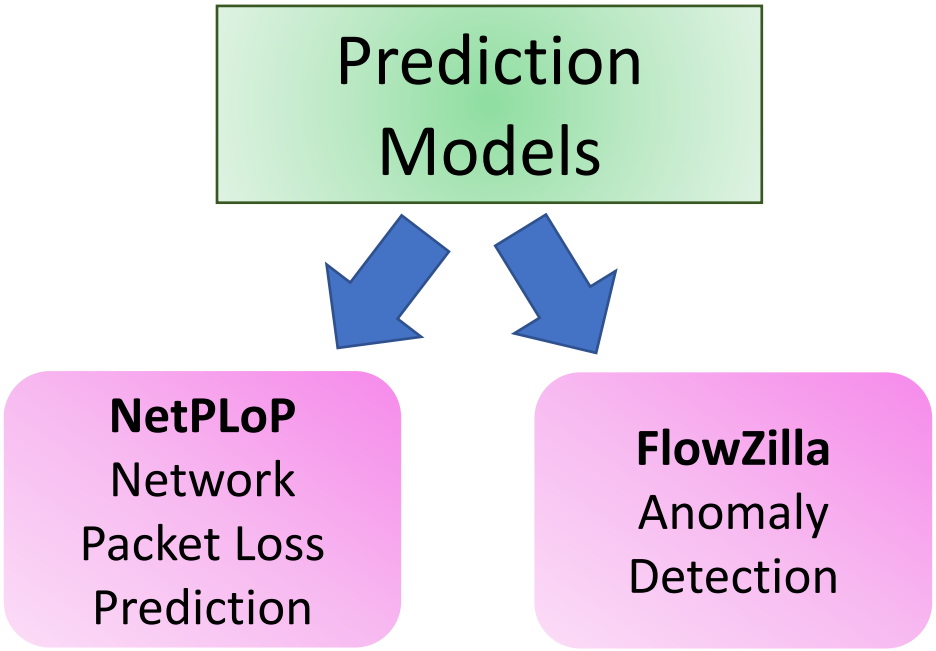


BERKELEY LAB

How do we mitigate the effects of performance degradation events?

- Detect them!
- Develop an accurate lightweight framework for performance degradation detection
- Integrate framework in real network environments
- Ultimately reconfigure the network → optimize data transfers

A lightweight framework for performance degradation detection



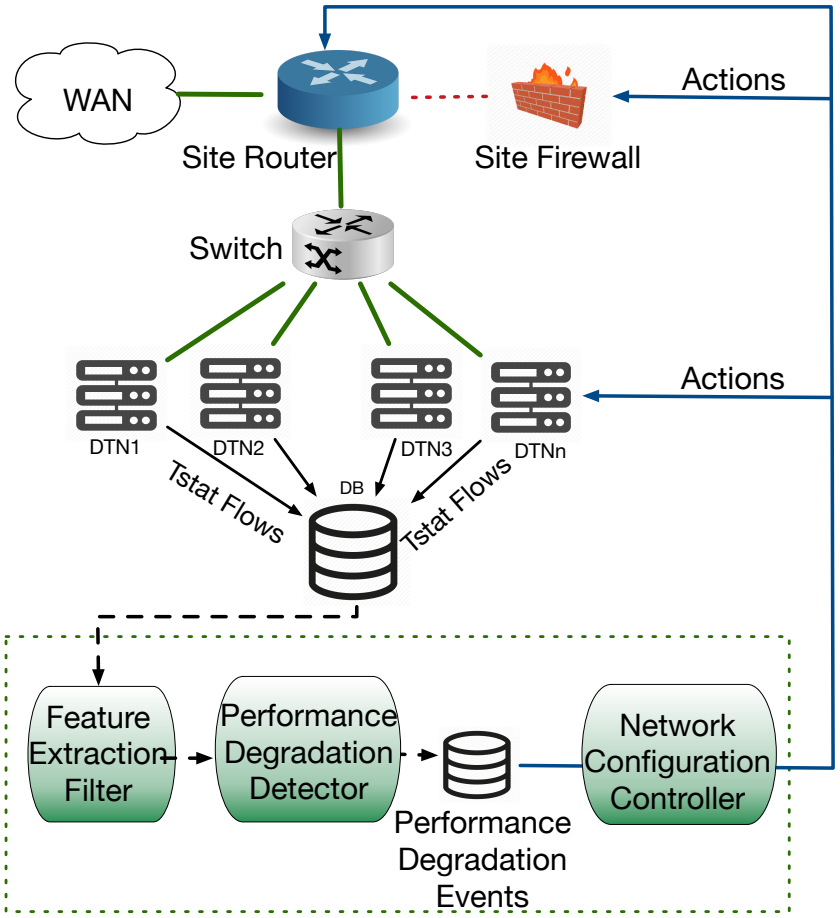
How do we predict packet loss?

NetPLoP

How do we detect abnormal transfers?

FlowZilla

Proposed Architecture



BERKELEY LAB

Issues with Packet Loss Prediction

Generally inaccurate predictions that don't apply to transfers of arbitrary size

How does NetPloP deal with this?

1. Select an appropriate set of flow properties
    1. Average RTT
    2. Throughput
    3. TCP Congestion Window
    4. Size and Duration
2. Train model using linear regression on ”faulty” transfers

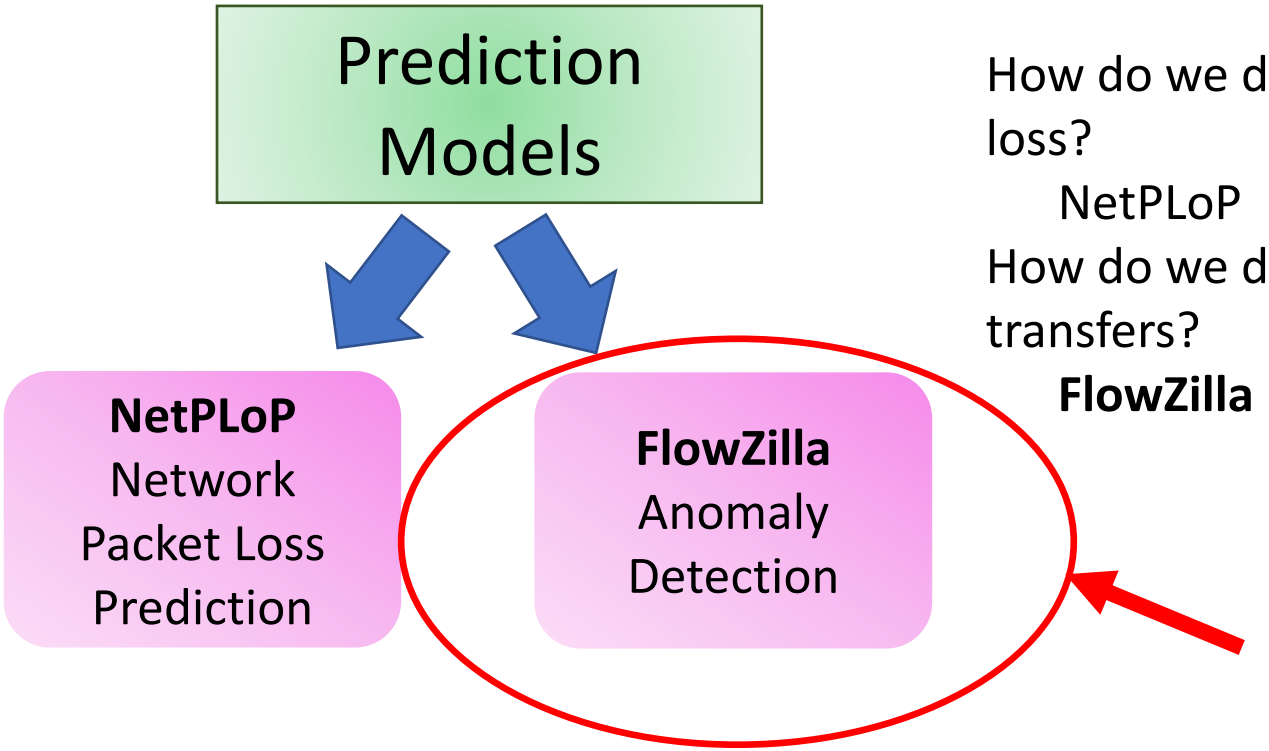
How accurate is NetPLop?

	Accuracy without smoothing	Accuracy with smoothing
Flows between 01/01/2017 – 30/07/2017	60%	97%
Flows between 01/01/2018 – 28/02/2018	<40%	66%

Noise removal significantly improves prediction accuracy

High variability in input variable distribution affects accuracy

Machine Learning for Performance Degradation Detection



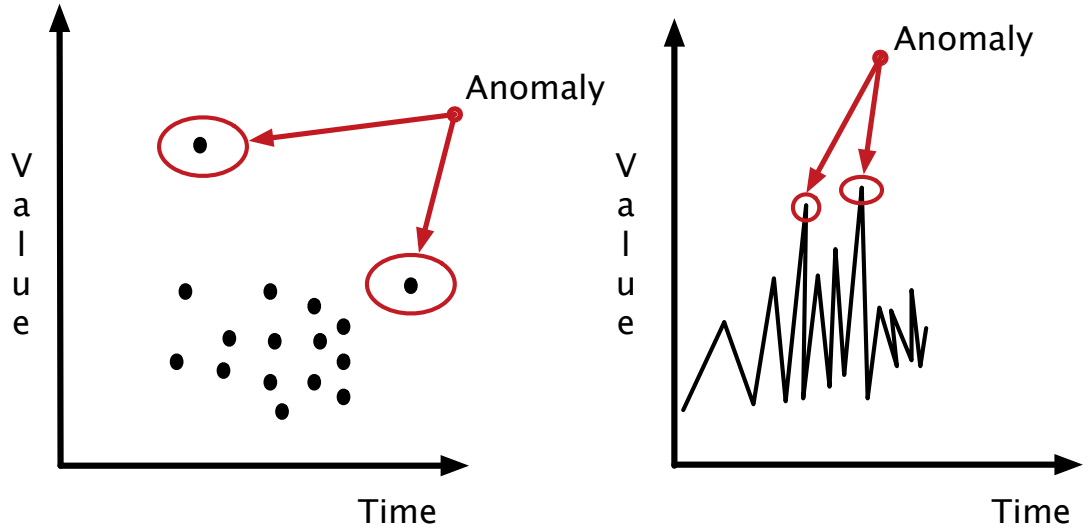
How do we detect packet loss?

NetPLoP

How do we detect abnormal transfers?

FlowZilla

Anomaly Detection - Limitations

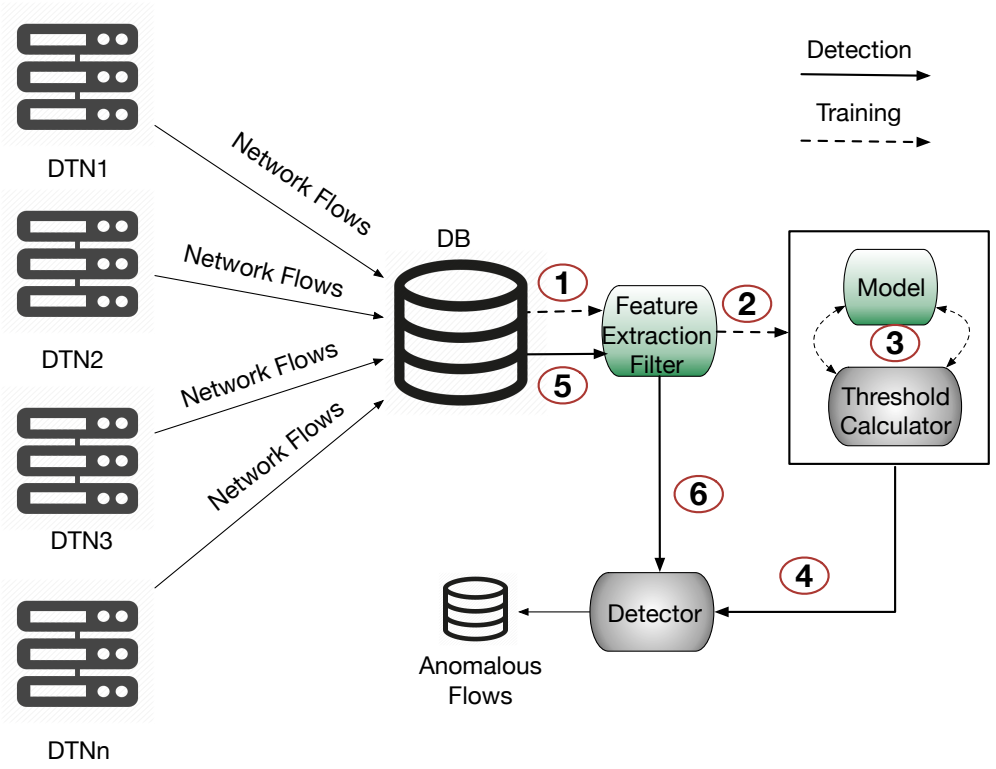


Defining a normal profile is hard due to dynamic feature distribution

Too many false positives

BERKELEY LAB

FlowZilla: A technique for Detecting Anomalies in Traffic volumes



Model input features:

- Network Throughput
- Flow Duration
- Source/Destination IP
- Linear regression

Threshold Calculator:

- $T = \mu + x$ where
  μ = mean value of
  $|V_{real} - V_{pred}|$
  x such that 90% of the flows are legitimate

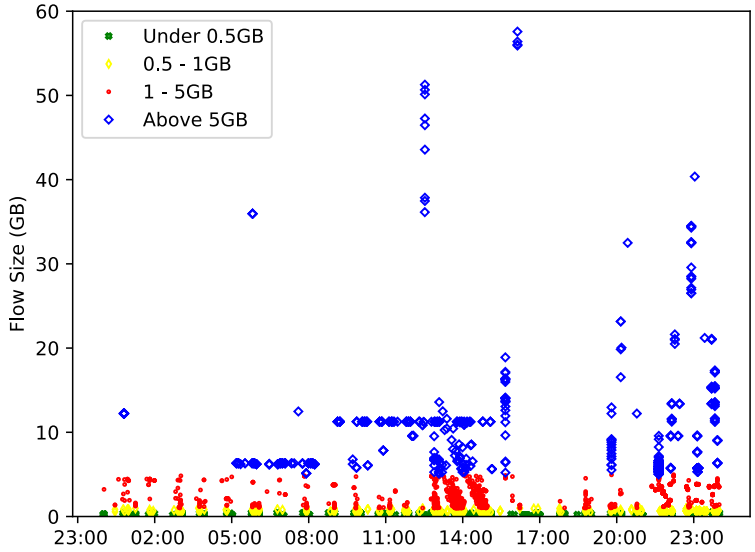
Training Data:

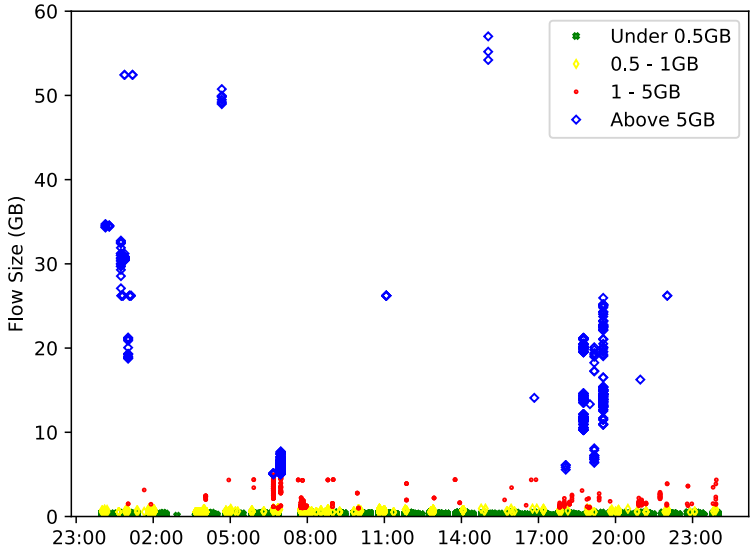
- Flows between 10/01/2017 - 11/30/2017

BERKELEY LAB

FlowZilla: Adaptive Threshold

- Threshold definition can be tricky
  - Too high → False negatives
  - Too low → False positives
  - Constant value does not account for seasonal trends





Data transfers are one week apart

FlowZilla Detection Results

Model trained on data transfers between 01/10/2017 – 30/11/2017

Experiment	Total Anomalies	Anomaly Size	True Positives	False Negatives	Total # of Flows
1	40	1-5 GB	34	6	12810
2	40	10 GB	37	3	30595

Detection rate remains above 80% in both experiments

Summary

AI provides the opportunity to realize smart networks

- Insights for real-time reactivity reduce human factor

Early results show reasonable accuracy and low training cost

Next Steps:

Real time deployment

Expand to additional types of performance degradation events

Acknowledgments

This work was funded by the US Department of Energy under contract no. DE-AC02-05CH11231

- Special thanks to: Sean Peisert, Dan Gunter, Dipankar Dwiwedi, Josh Bovenhof, Ravi Cheema, Jon Dugan, Eric Pouyoul, Mariam Kiran, Brian Tierney, Alberto Gonzales, Jason Leigh, Alan Whinery, Ed Balas, Dan Doyle, CJ Kloote, Jennifer Schopf, Chin P. Guok, Inder Monga.

Questions?