Classifying **Elephant** and **Mice** Flows in High-Speed Networks

Mariam Kiran
Anshuman Chabbra (NSIT)
Anirban Mandal (Renci)

Presented at INDIS 2017

ESnet, LBNL

Funded under DE-SC0012636

Talk Agenda

- Current challenges in Elephant and Mice flows: Why bother?
- Unsupervised machine learning techniques: Why?
- Solution: Development of a learning classifier system using GMM
- Current state – lessons learned and exploitation of classification results
- Evaluation and Future work

Myth not in Networks!

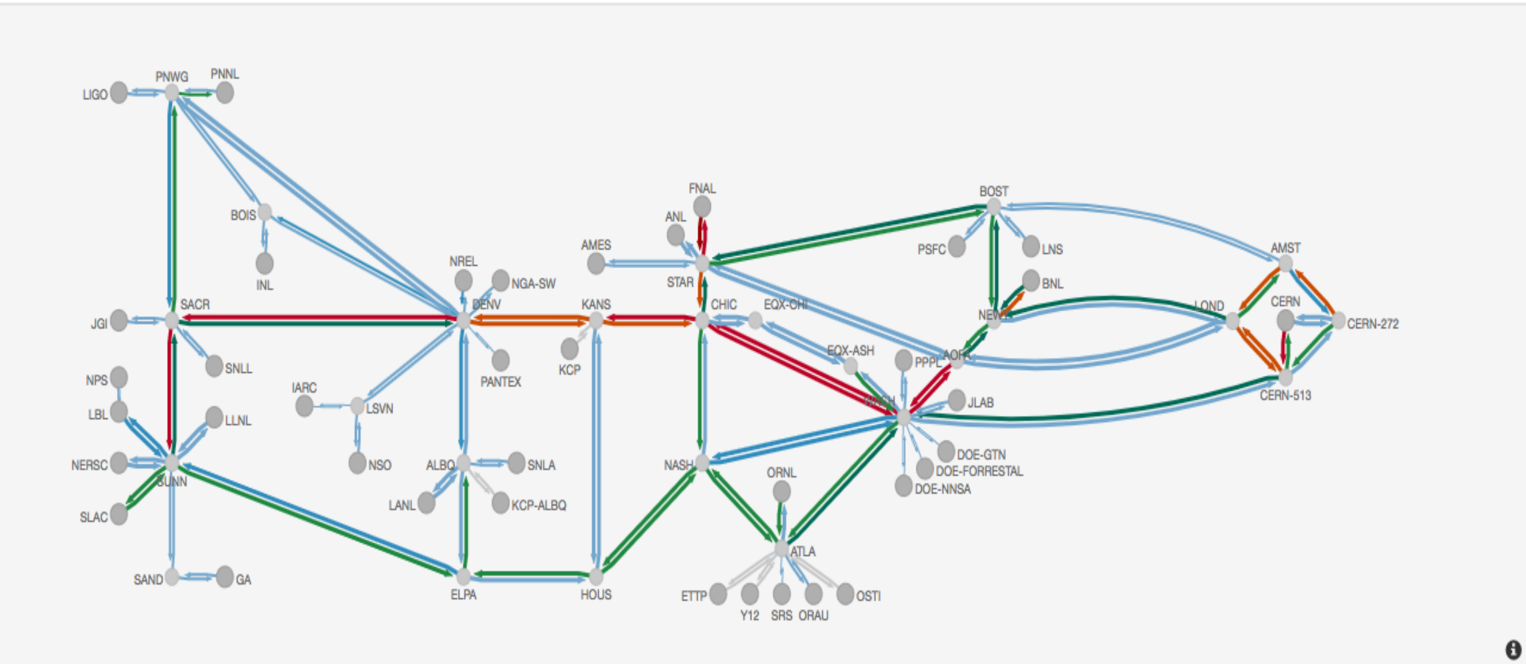
"Elephants scared of Mice"

- Data centers and networks get a mixture of flows
  - <u>Elephant flows:</u>
    - Large size
    - Long-lived
    - Large data transfers
    - Throughput-sensitive
  - <u>Mice Flows:</u>
    - Smaller bursty traffic
    - Short-lived
    - Latency-sensitive

- Scientific networks versus data center traffic
  - **Majority flows:** Elephant flows (Big data files)
    - Gobbles up network buffers causing queuing delay to mice flows
    - Challenges of adaptive routing: Changing paths on-the-go
    - Links also have to be optimized: multi-objective problem

ESnet

Why should we understand flows?



Network Overview

This map highlights the sites ESnet serves, the s
network and the current traffic load. Clicking on
info button will show details. This map makes so
simplifications, click on info button for more det

LEGEND

ESnet Site
Hub

100G
10G
1G
< 1G

>=50 Gbps
20 - 50
10 - 20
5 - 10
2 - 5
1 - 2

Our networks is very dynamic.

Losing data or jeopardizing applications prevents us to achieving our mission!

Goal is to <u>detect</u> and then <u>manage</u>

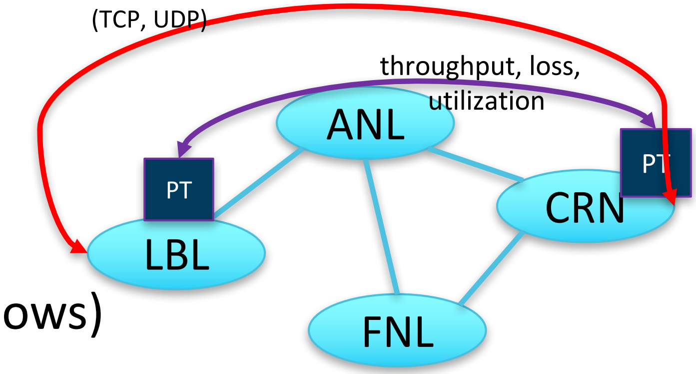
ESnet

Previous work

- Classify traffic for intrusion detection and traffic profiling
  - Number of packets transferred, flow duration, file size
  - Papers link tools to perform dynamic traffic steering
    - Isolating traffic streams
    - Based on size, rate, duration, burstiness, or combination
- However real-time detection is a challenge!
  - Online (as flow arrives) versus offline analysis (periodic)

- S. Shirali-Shahreza et al. Traffic statistics collection with Flexam, in: Proceedings of 2014 ACM SIGCOMM.
- T. Zizhong Cao et al. Traffic steering in software defined networks: planning and online routing, SIGCOMM workshop on Distributed cloud computing.
- Z. Yan et al. A network management system for handling scientific data flows, Journal of Network and Systems Management 24 (2016) 1–33.

ESnet



Lets use Netflow Records

- Netflow: Collected every 5 minutes (aggregated flows)
  - Perfsonar: active testing for health

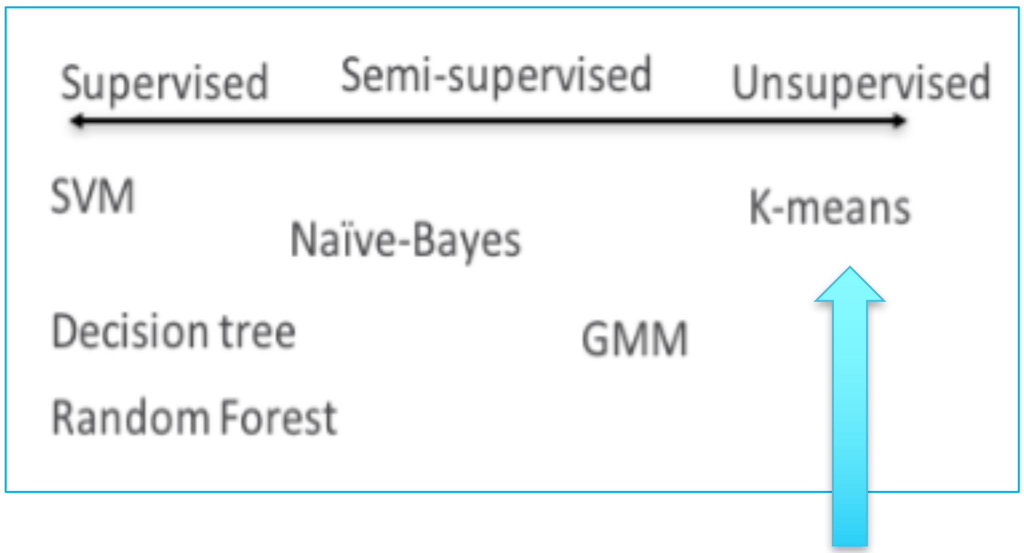
Flow first seen	Duration	Protocol	Source IP:Port	Destination IP:Port	Packets	Bytes	Flows
2017-04-15	00:00:23.040	TCP	50.127.55.32:3455 ->	137.243.29.226:23	0	40	1
2017-04-15	00:00:23.040	UDP	120.129.253.114:9788 ->	121.127.238.102	0	42	1
2017-04-15	00:00:23.850	UDP	120.129.253.114:9433 ->	121.127.151.25	0	42	1

  - Every site is unique: traffic received

Site (1 month)	Mean (size)	Max (size)	Mean (duration)
ROne	0.15	25.6	23.19
RTwo	0.03	36.4	4.14
RThree	0.02	72.5	6.63

Finding elephants and mice in flows

- Exploring Netflow data
- Cluster traffic into TWO groups with **<u>NO</u>** prior knowledge

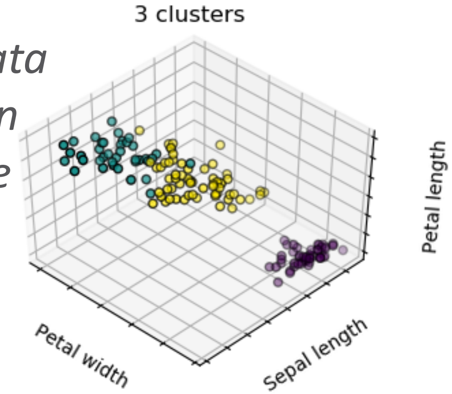


- Unsupervised learning: Organize data into clusters based on attribute values:
  - Find patterns, relationships, similarity across data

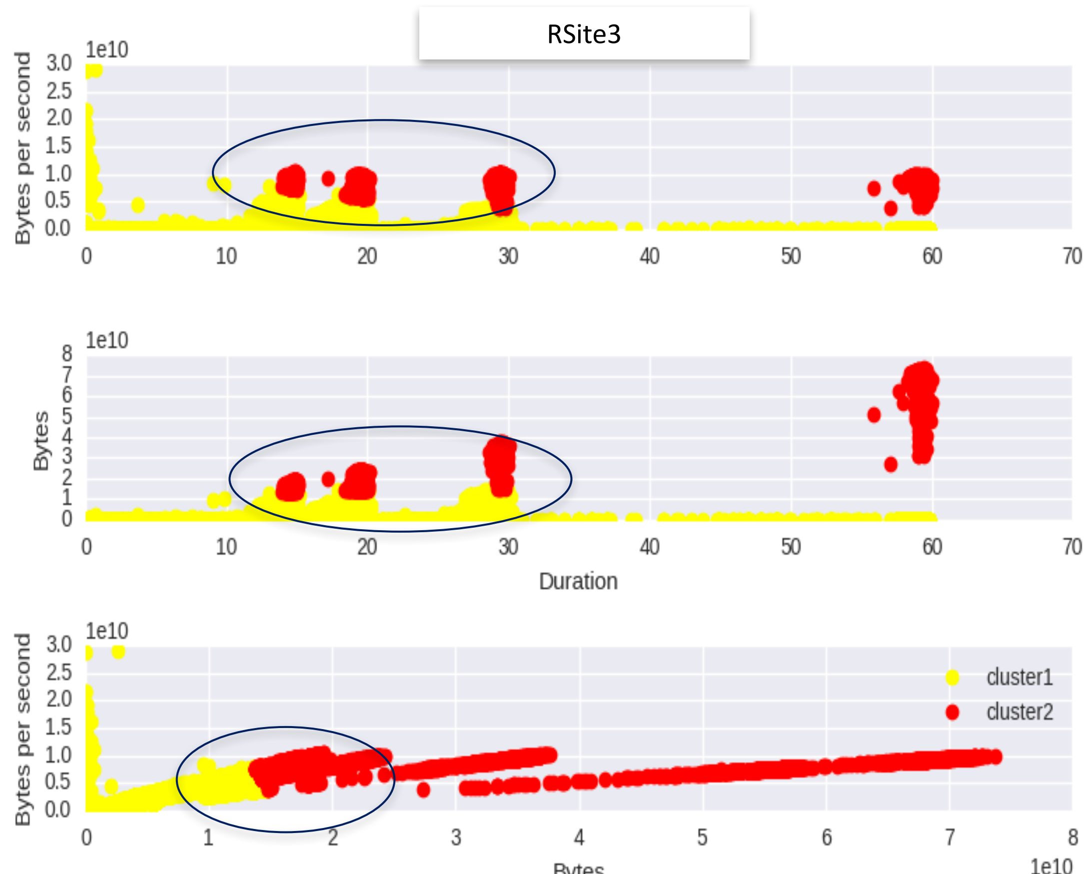
ESnet

K-means results

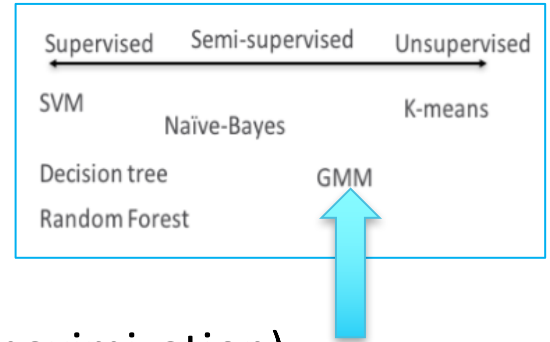
Cluster data based on distance



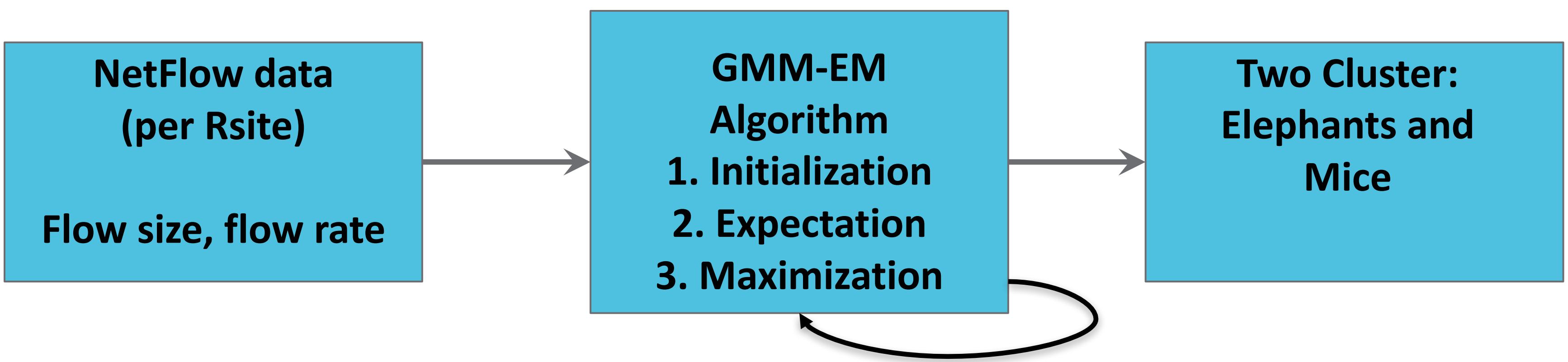
- Start with no knowledge and find centroids with closest data points
- Target: Form 2 clusters based on size and bytes/s
- Results:
  - Overlapping data points in clusters
  - Algorithm fails due to different density and data size in flows
- We need some knowledge in the algorithm



Gaussian Mixture Model (Semi-supervised)



- Scikit-learn python library for GMM-EM (Expectation maximization)
  - Only 30 lines of code
  - Semi-supervised: Initialize with some knowledge
    - Assume 10% elephant and 90% mice and then refine μ_e=0.1, μ_m=0.9
    - Compute probability of flow belonging to cluster and update μ_e, μ_m
    - Compute mixture coefficients per site
    - Repeat process until converge to a local optimum.



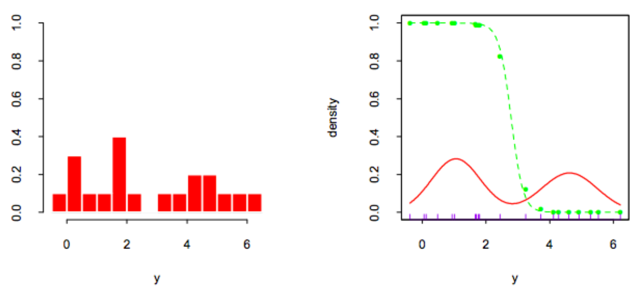
ESnet

Working of GMM-EM algorithm

- Flow characteristics are dependent:
  - Per site
  - Per time of the day

$$p(X) = \pi_e \mathcal{N}(X|\mu_e, \Sigma) + \pi_m \mathcal{N}(X|\mu_m, \Sigma)$$

- GMM assumes there is a Gaussian distribution of mixture of classes
  - Data set is a mixture of elephant and mice flows



- Maximum likelihood fit to Gaussian density (red)
- Observation data set (green) also called responsibility

$$\mu_e = \mu + \pi_e(max(X))$$

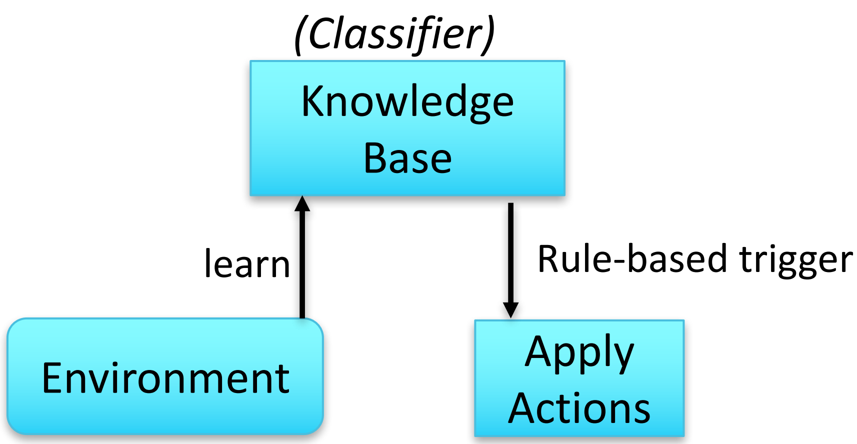
$$\mu_m = \mu - \pi_m(min(X))$$

- Initialization Step: 10% flows are elephant in my traffic (0.1,0.9)
- Expectation Step: Compute belonging to a cluster based on Gaussian equations
- Maximization Step: Keep re-iterating till converge

ESnet

Use Classification to build a LCS

- LCS = Learning Classifier System

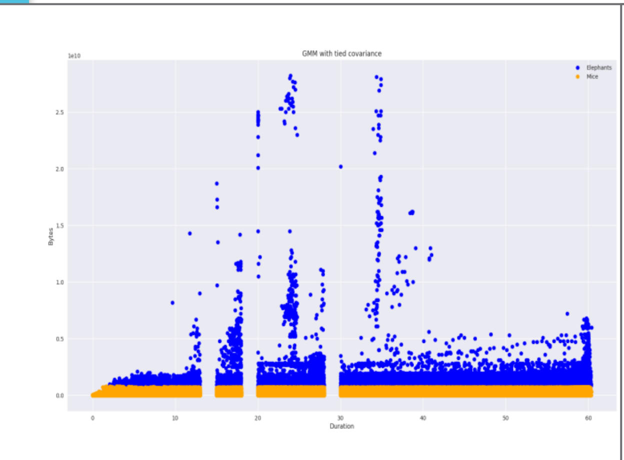


- Each site is different, and flow characteristics change over time
- Classifier will find different characteristics of elephants and mice:
  - Not have a predefined definition e.g. thresholds

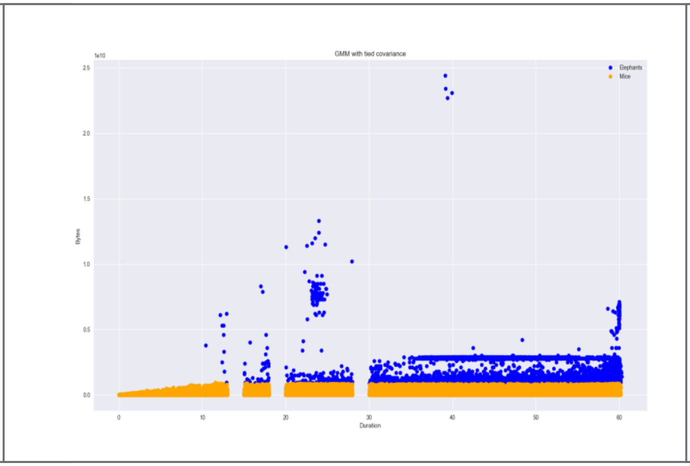
Results

Semi supervised gives better results

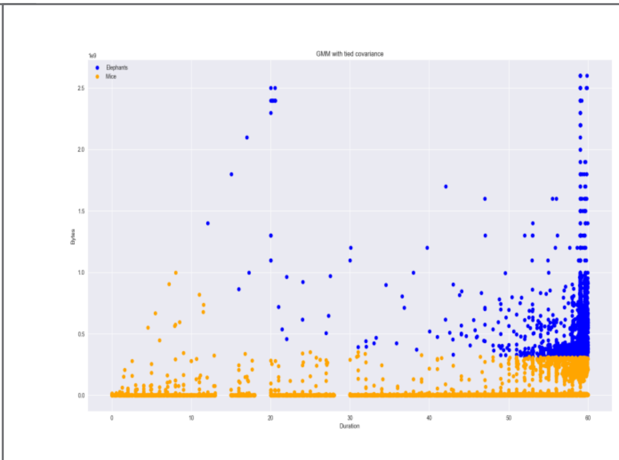
- Clear clusters found!
- Each site cluster has different characteristics



Rsite1



Rsite2



Rsite3

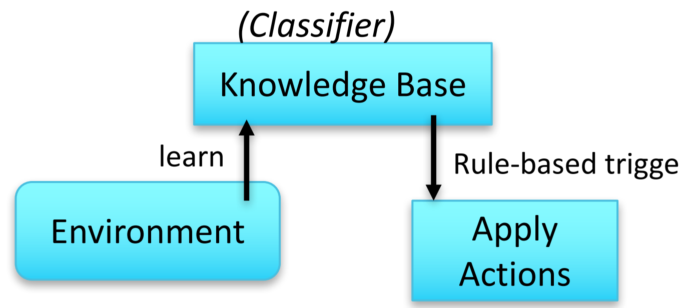
- Blue = Elephant, Orange = Mice
- Rsite1 more Elephants flows compared to Rsite2/Rsite3
- Mice flow ranges are different for Rsite3

What lessons did we learn?

- Clustering leads to more statistical analysis on what elephants/mice are
- Too much Noise in data:
  - First few netflow records contained Perfsonar tests,
    - being classified as elephant flows, had to be cleaned
- Needed some knowledge for semi-supervised:
  - Leads to skewed results of elephants lying in top 10% size and rate
  - Need an independent verification with ground truth data
    - E.g. Simulating GridFTP transfers to see if recognized as elephants
- ML BlackBox problem:
  - Using ML libraries does not expose internal algorithm workings
  - Propose building ‘open’ libraries

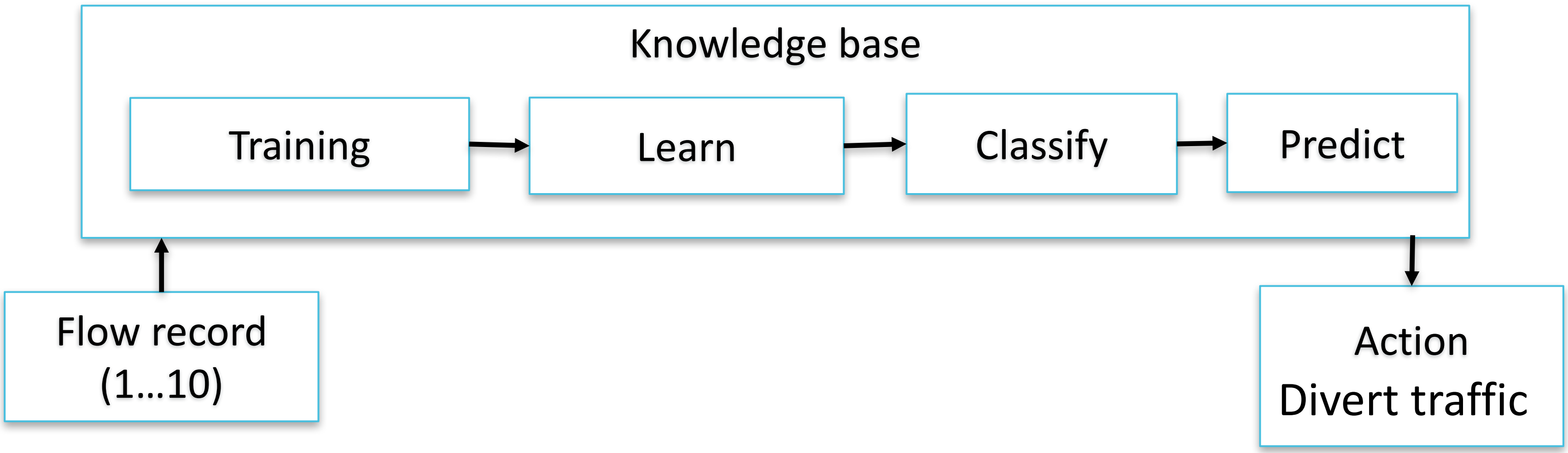
ESnet

Is Netflow enough?



- Initial idea was:
  - Can we to Active Traffic Steering using identified clusters?
- There is Noise: difficult to recognize
  - Link testing data
  - No track of congestion on link
  - Bad configuration
  - Sampling rate can be altered
- Additional infrastructure required
  - Sflow: Expensive but is it worth it?
- More end-to-end data
  - Whether flows captured belong to same stream? Interface/port data
  - I/O data

Building Learning classifier system



- Active steering: Netflow data is past data
  - Thresholding mechanisms are good approaches!
  - Needs more testing for how flows can be isolated
- Not do active steering but learn about sites
  - how heavy traffic is?
  - Add more links, add more infrastructure, fault management

ESnet

Conclusion

- Overall was easy to implement but has its caveats
- Focused on online training and learning per site: Unique compared to existing works in area
- Processing time is fairly fast
- Next steps
  - Working through the GMM algorithm to plot how Gaussian mixture changes
  - Run real-time tests to see if we can isolate traffic streams based on netflow classification
  - Understand flow behavior across sites

Thankyou

- Any Questions?

  - We do have an open PostDoc position (ML in Networks)

  Please reach out

  - <mkiran@es.net>

ESnet