Deep Learning for Anomaly Detection and Cause Identification

Quentin Fournier

quentin.fournier@polymtl.ca

École Polytechnique de Montréal
Laboratoire DORSAL

Automatic Cause Detection of Performance Problems

- Setup:
  - Apache server
  - $\approx$ 50000 requests
  - $1 - 1000$ clients
- Problem: $\approx$ 200 slow requests
- Objective:
  - Automatically detect anomalies
  - Highlight possible causes

Automatic Cause Detection of Performance Problems

Proposed approach

1. Extract features such as `blocked for disk`, `blocked for network`, `blocked for CPU`, etc.

2. Apply an unsupervised method to detect abnormal requests (dbscan)

3. Once abnormal requests are detected, group them by their behaviour (k-means)

4. Do a statistical analysis to find hints of the cause (n-gram)

Automatic Cause Detection of Performance Problems

- Conclusion:
  - PHP OpCache contention
  - Difficulty to initialise sockets
- Paper accepted and presented at the 3rd International Workshop on Software Faults.

You are welcome to ask about my previous works during hackathon!

Current Drawbacks

- Most of the existing works require a domain expert:
  - specify the normal behaviour
  - provide relevant features
  - set a threshold
  - etc.
- Each method is somewhat specific to the problem:
  - type of data
  - type of anomaly
  - environments
  - etc.
- Systems change quickly, methods need to be adapted constantly.

Current Drawbacks

If we spend less time designing methods to detect anomalies... we have more time to solve them!

Properties of the Ideal Anomaly Detection Method

One would like a method:

- Data-agnostic: works on any trace
- Anomaly-agnostic: detects any type of anomaly
- Online: adapts to changes over time
- Interpretable: provides the reason for the anomaly (hint of the cause)

This method is a unicorn...

Research Scope

Let us start by addressing partially:

- Data-agnostic:
  - Work on kernel events
  - Extend later to userspace events
- Anomaly-agnostic:
  - Only performance anomaly
  - No specific design allowed

Research Scope

Let us start by addressing partially:

- Offline:
  - Faster than online
  - Readily adaptable to online
- Interpretability:
  - Unsolved problem...
  - Which part of the trace is most useful to detect the anomaly

Deep Learning, a Potential Solution.

- Neural networks have the potential to solve all of these problems at once!
- Novel techniques are proposed every month
- Most of them have never been applied to trace data and anomaly detection
- In particular the attention mechanism is promising

Attention Mechanism

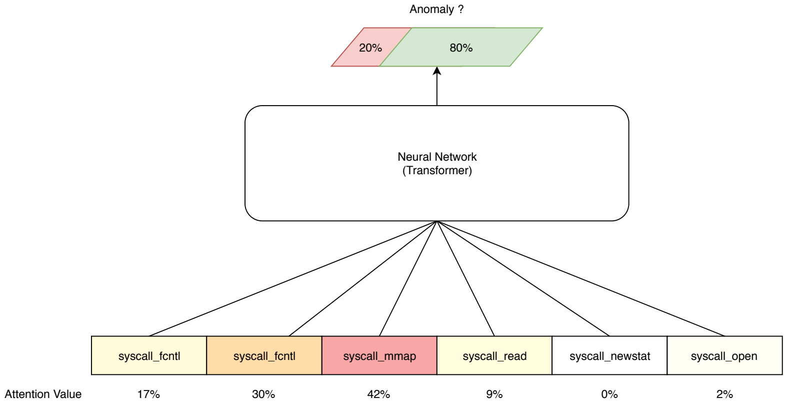
The attention mechanism tells us:

- How important is each input to compute the output
- Which part of the trace helps the most for the anomaly detection

Properties:

- Robust to long sequences
- Fast to train
- In most state-of-the-art models
- Provides insights into the network behaviour

Attention Mechanism



A neural network that detects anomalies of an execution from system calls with attention.

Framework

1. Generate requests with different bottlenecks:
    - CPU
    - Memory
    - Network
    - etc.
2. Train a model:
    - Input: subsequence of kernel events
    - Output:
        - Duration
        - Label (normal/anomaly)
        - Following events

Framework

What is the network utility?

- Classify if a new request is an anomaly
- Predict if a currently running request will become an anomaly
- Highlight possible anomaly cause based on the attention values

Conclusion

- Adapt the latest deep learning contributions → anomaly detection
- Rather unexplored problem
- Start simple, take small steps

Thank You