Metadata format for benchmarking anomaly detection algorithms

Youki Kadobayashi
NICT / NAIST
youki-k <at> is.naist.jp

Anomaly detection algorithms: The problem

- We are still in the dark ages
    - Incompatible datasets
    -
    - Incomparable results
-
- No technical method to accurately communicate the result of anomaly detection, even if we share the common dataset
- Inability to benchmark their performance

Metadata format for anomaly detection algorithms

- Separate file for each algorithm
- XML-based
- header, {record1, record2, …}
-
- Envelope information: rely on datcat tools

Header

- Algorithm name
- Algorithm version
- Algorithm URL
- Parameters given to the algorithm
- Date of analysis
- Analyst name
- Analyst organization
- Target dataset
- DATCAT dataset name

Record

- Each record consists of:
  - src, dst, start_time, end_time, anomaly_type, anomaly_value
-
- Arbitrary number of records
-
- Either src or dst can be wildcard

API

- label_data(int handle, in_addr_t src, in_addr_t dst, time_t start, time_t end, string anomaly_type, float anomaly_value)
- label_data_ex(int handle, in_addr_t[] src, in_addr_t[] dst, time_t start, time_t end, string anomaly_type, float anomaly_value)

Slicing

- Slice anomalous segments of pcap data
  - Based on anomaly_type, anomaly_value
-
- Slice pcap data according to start_time, end_time
-
- Useful for generating synthetic dataset

Merging

- Insert pcap slice B into pcap slice A
  - At particular time offset
-
- Useful for benchmarking anomaly detection algorithms with synthetic dataset

Comparison

- Visualize the spotted anomalies along timeline
-
- Compute coverage and support, generate HTML table

Current status

- Implementation in progress
-
- Your comments are welcomed
-
- youki-k <at> is.naist.jp