Comparative Causal Analysis of Network Log Data in Two Large ISPs

Satoru Kobayashi, Keiichi Shima, Kenjiro Cho,
Osamu Akashi, Kensuke Fukuda

AnNet 2022

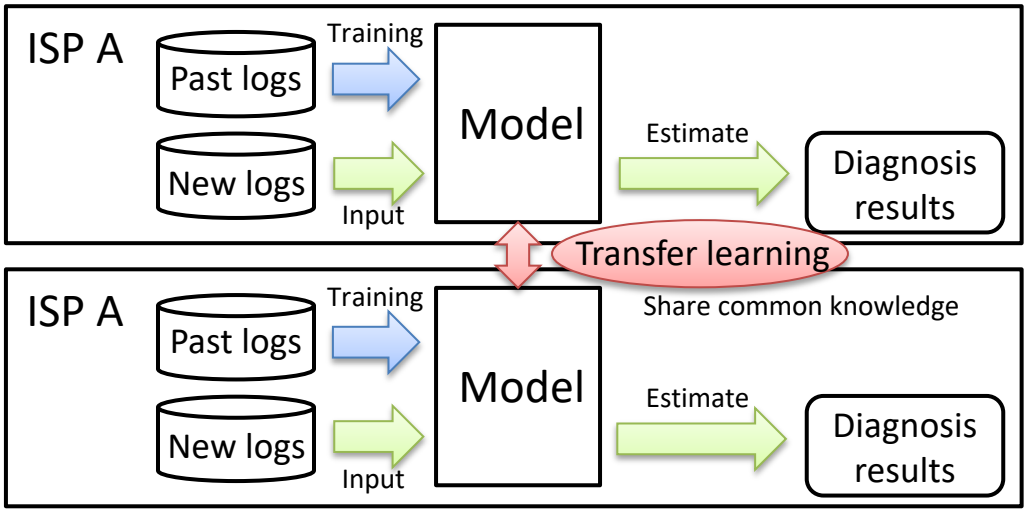
Toward automated network log analysis

- Automated log analysis
  - Necessary for recent large-scale networks and their logs
  - Especially important for root cause analysis of network troubles
- Machine learning approach for network root cause analysis
  - Lack of diversity in training data
  - Weak for unknown trouble cases

➢ Collaborative (inter-ISP) log analysis

Future collaborative log analysis

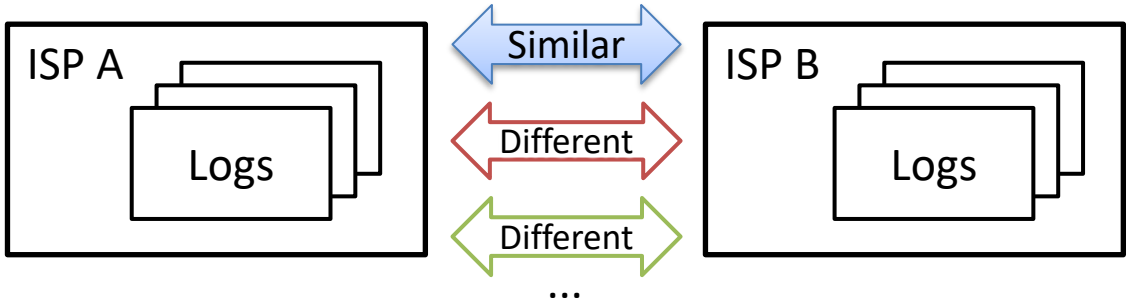
- Collaborative log analysis of multiple ISPs
  - Learn more (and diverse) troubles
  - Can be effective for unknown troubles (If appeared in other ISPs)



Difficulty in collaborative log analysis

- Does there really exist transferable knowledge?
  - If not, transfer learning loses accuracy and reliability
- We need to preliminarily compare multiple log datasets
  - To examine the transfer learning is effective or not in advance

Research goal: Propose a comparative log analysis technique of ISPs



Challenges for comparative log analysis

- Difference of environment (vendors and network topology)
  - Different system behavior
  - Different log formats and variables
  - Different logging behavior (when to log)
  - ➢ Difficult to compare directly
- Data publication policy of ISPs
  - Network logs include sensitive information
  - ➢ Need anonymization

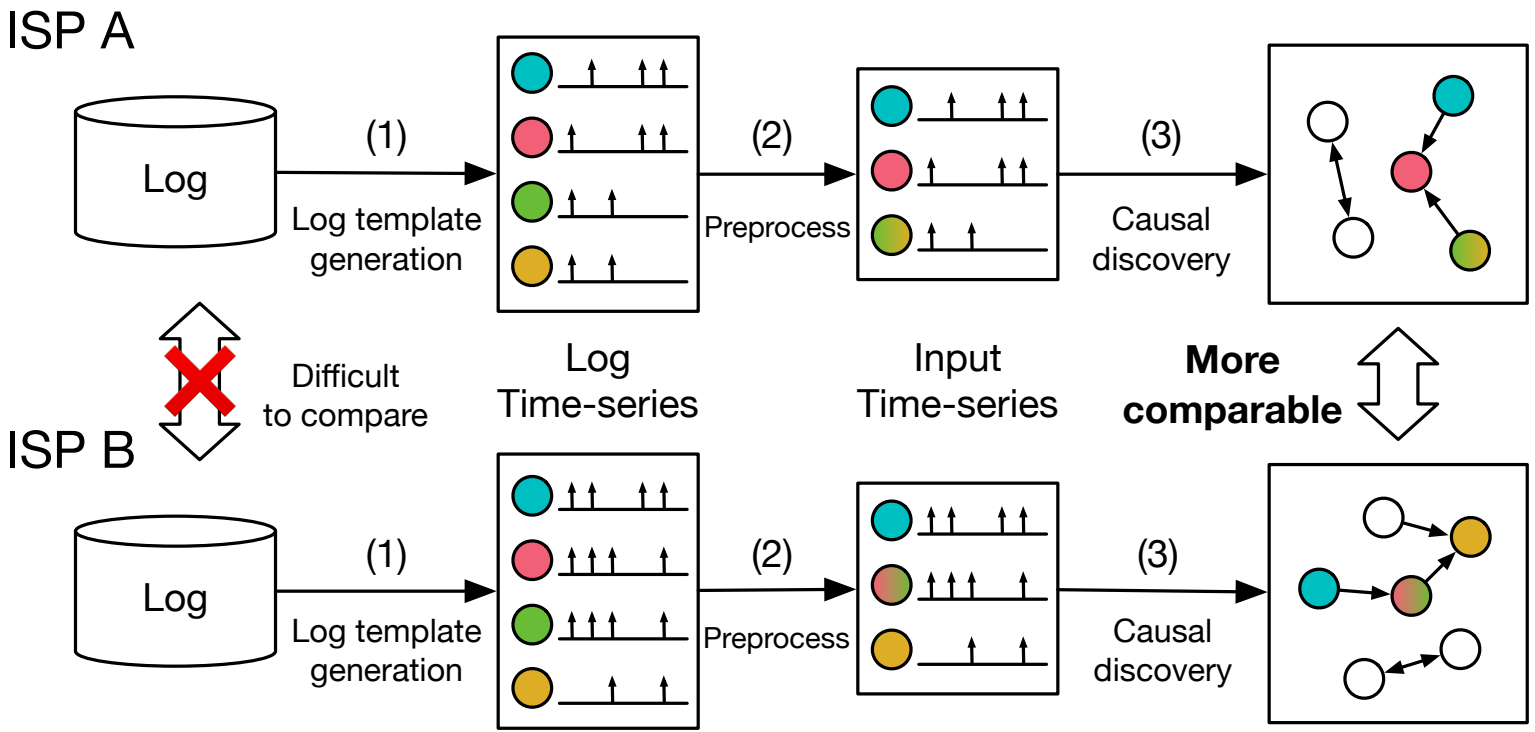
Key idea

- Log messages -> Time-series event with log templates
  - Time-series event: same log template, same host device
  - Helpful for anonymization
- Focus on event causality
  - If there is a same network behavior, there can also be similar causal relations of log events [1]
  - Clear and direct relations without spurious correlation

[1] S. Kobayashi, et al. “Mining Causality of Network Events in Log Data”, IEEE Transactions on Network and Service Management, pp.53-67, vol.15, no.1, March, 2018

Analysis flow overview

1 DAG for
1-day log data

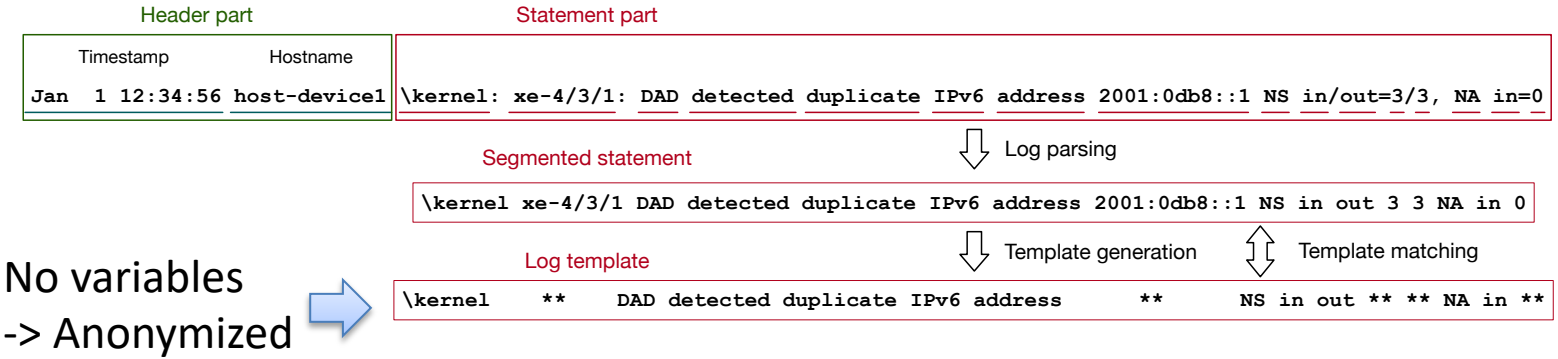


1 time-series node: events with 1 log template from 1 host device

(1) Log template generation

1. Parse log messages into header information and statements
2. Generate log templates from log statements
3. Classify log messages with the templates

amulog [2]

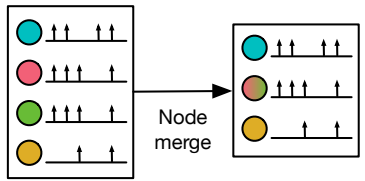


No variables
-> Anonymized

[2] S. Kobayashi, et al. “amulog: A General Log Analysis Framework for Comparison and Combination of Diverse Template Generation Methods”, International Journal of Network Management, Wiley, 2021

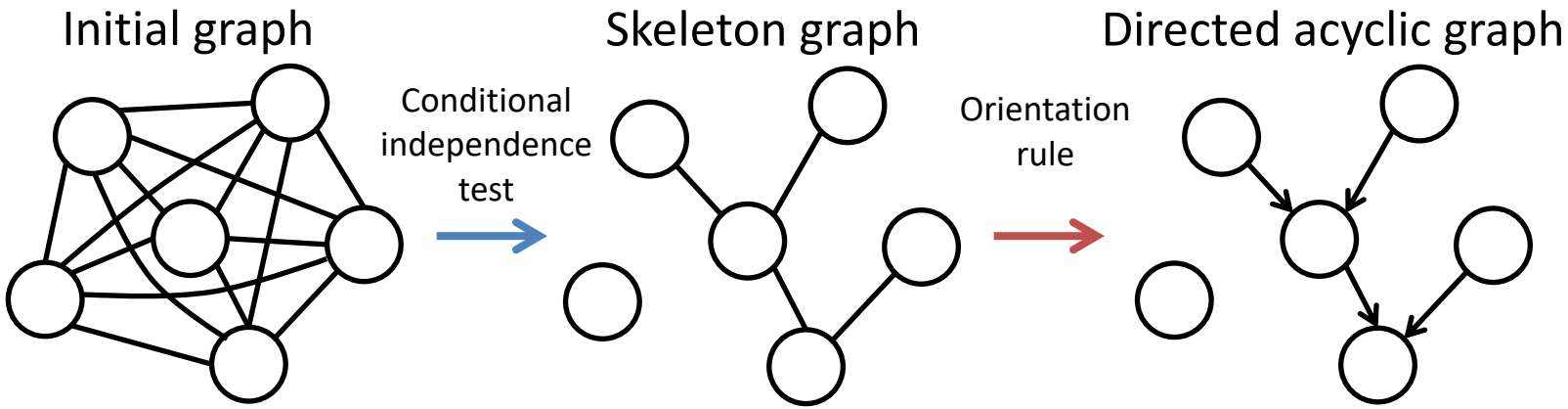
(2) Preprocessing of input time-series nodes

- Decrease processing time of causal discovery
  - ➢Remove periodic component of log time-series [1]
    - Ignore daily or weekly (planned) system behavior
  - ➢Prune causal edge candidates with prior knowledge [3]
    - Considering network topology and protocol layers of events
  - ➢Merge completely synchronizing time-series nodes (new)
    - Decrease the number of input nodes
  - – These three methods can be used together



[1] S. Kobayashi, et al. “Mining Causality of Network Events in Log Data”, IEEE Transactions on Network and Service Management, pp.53-67, vol.15, no.1, March, 2018
[3] S. Kobayashi, et al. “Causal analysis of network logs with layered protocols and topology knowledge”, CNSM’19, pp.1-9, 2019

(3) Causal discovery



- PC algorithm [4]
  - Relatively fast causal discovery method (available for large dataset)
  - With G square test (for binary time-series)

[4] P. Spirtes, et al. “Causation, Prediction and Search.” The MIT Press, 1996.

Evaluation outline

- Datasets
- Validation of node-merging preprocessing
- Evaluation (Comparative causal analysis of ISPs)
  - Causal analysis results
  - Details of Circuit-related causal edges
  - Case study

Datasets

- ISP A
  - nation-wide ISP in Japan
  - 56,968,361 log lines
  - 92 days
  - 1,861 hostnames
  - 36 corresponding trouble tickets
  - 5,182 log templates
  - NOT using preprocessing of prior knowledge
- ISP B
  - nation-wide ISP in Japan
  - 34,722,785 log lines
  - 365 days
  - 131 hostnames
  - 88 corresponding trouble tickets
  - 1,789 log templates
  - Using preprocessing of prior knowledge

Validation of node-merging preprocessing

- Causal results of ISP B without/with node merging
  - (each value is the average of every 1-day log data)
  - Processing time: 76.0 sec -> 36.3 sec (52% decreased)
  - Number of nodes: 360.1 -> 279.4
  - Number of edges: 56.8 -> 49.8
  - Corresponding trouble tickets: 70/88 -> 71/88

Equivalent reliability with smaller results

Node merging enables:
- Faster calculation
- More reliable causal results

Evaluation - Comparative causal analysis of two ISPs

Causal analysis results

Network	#Nodes	#Edges	#Tickets
ISP A	2,758.3	349.8	18 (42%)
ISP B	279.4	49.8	71 (81%)

Classification of tickets

Network	Class	#All tickets	#Tickets with edges
ISP A	Circuit	22	15 (68%)
	Connection	7	0 (0%)
	Device	7	3 (43%)
ISP B	Circuit	22	14 (63%)
	Connection	55	50 (91%)
	Device	7	4 (57%)
	Blackout	4	3 (75%)

Similar results in Circuit troubles

Evaluation - Details of Circuit-related causal edges

Aggregated with adjacent nodes of causal edges

Network	Node label	Days w/ logs	Days w/ edges	Days w/ tickets (edges/tickets)
ISP A (92 days)	MPLS	88	69	12 (17%)
	System	92	92	5 (5%)
	Interface	92	92	5 (5%)
	Monitor	90	53	4 (4%)
	OSPF	61	5	1 (20%)
ISP B (365 days)	Monitor	191	60	10 (17%)
	MPLS	39	13	4 (31%)
	BGP	315	291	4 (1%)
	Interface	318	211	3 (1%)
	OSPF	54	1	1(100%)

MPLS, Interface, Monitor:
Found in many days
-> Regular behavior

OSPF:
Logs regularly appear,
but causality is rare
-> Anomalous if OSPF has causality with others

Another aggregation of causal edges related to Circuit tickets

Network	Label 1	Label 2	Same host	Days w/ edges	Days w/ tickets (edges/tickets)
ISP A (92 days)	MPLS	MPLS	✓	11	5 (45%)
	System	System	✓	91	3 (3%)
	MPLS	MPLS		28	5 (18%)
	Monitor	Monitor		22	3 (14%)
	System	System		13	2 (15%)
	Interface	Interface		59	3 (5%)
	Monitor	OSPF		1	1 (100%)
	Interface	OSPF	✓	1	1 (100%)
	Interface	Monitor		1	1 (100%)
	Monitor	OSPF		1	1 (100%)
	Interface	OSPF	✓	1	1 (100%)
	Interface	Monitor		1	1 (100%)
	System	MPLS		2	2 (100%)
	System	Interface		3	1 (33%)
	Interface	Monitor	✓	1	1 (100%)
	Monitor	Monitor	✓	1	1 (100%)
	Monitor	MPLS		1	1 (100%)
ISP B (365 days)	Monitor	Monitor	✓	28	9 (32%)
	BGP	BGP	✓	215	4 (2%)
	Monitor	MPLS	✓	5	4 (80%)
	Interface	Interface	✓	166	3 (2%)
	Monitor	BGP	✓	3	1 (33%)
	Monitor	MPLS		1	1 (100%)
	MPLS	MPLS	✓	1	1 (100%)
	MPLS	MPLS		3	1 (33%)
	OSPF	OSPF	✓	1	1 (100%)

Aggregated with node pairs of causal edges

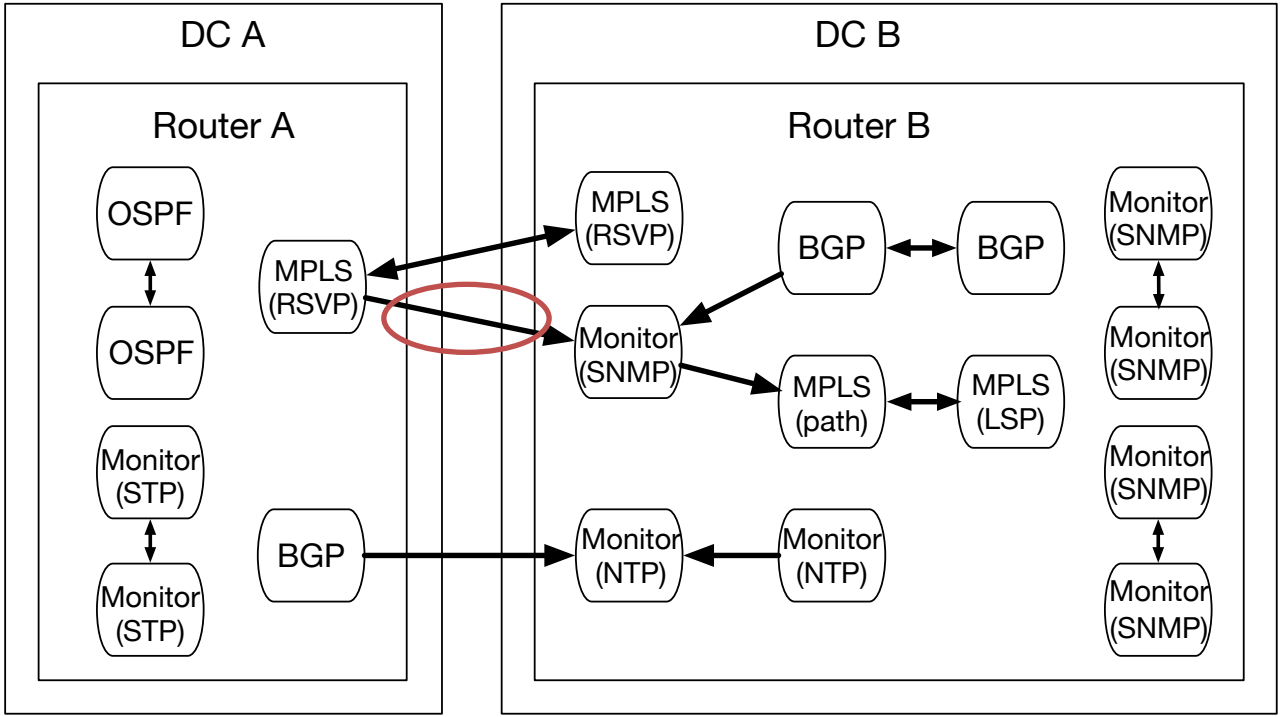
Edges between same labels
(Within a protocol function)
-> Relatively frequent (regular)
but sometimes related to troubles

Edges between different labels:
(Communication between protocols)
-> **Anomalous and related to troubles**

Mainly adjacent to subordicate functions (**Interface** and **Monitor**)

Evaluation - Case study

One of Circuit troubles in ISP B



Edges across devices between different labels (Rare and large behavior)

Found similar edge in ISP A too

-> Similar behavior in different ISPs

Discussion

- Causal approach is effective for dataset comparison
  - Logs appear regularly in any classes -> Which to focus?
  - Log causality can reveal large and relational behaviors
- How about other tickets (Connection and Device)?
  - Difficult to compare at least between these ISPs
  - Connection: Depends largely on used network protocols
  - Device: Depends largely on device vendors and models

Conclusion

- Goal: Comparative log analysis between different ISPs
- Approach: Causal discovery of time-series events classified with log templates and host devices
- Performance: Improved with node-merging by decreasing 52% of the processing time
- Result: Contribute to finding similar behaviors in two ISPs (especially on Circuit-related troubles)
- https://github.com/amulog/logdag

This work is supported by MIC/SCOPE #191603009
and JSPS KAKENHI Grant Number JP22K17886.