Causal Analysis of Network Log Events

Satoru Kobayashi

Project Researcher, NII

Feb 18, 2020

Outline

- Background
- Approach: Causal analysis
- Challenges and Solutions
  - Generating time-series from log data
  - Decreasing false edges
  - Improving processing time
- Evaluation with SINET data
- Conclusion

Difficulty of leveraging system log in network management

- Huge dataset
  - Large scale and complicated systems
  - 150,000 lines / day in SINET 5
  - Automated analysis required
- Difficulty in automated analysis
  - Free-format and sparse data
  - Contextual information required for troubleshooting

Automated analysis of system log

- Usage of log data in existing automated analysis
  1. Anomaly detection
  2. Fault localization
  3. Root cause analysis
- Analyzing log data is more effective for root cause analysis than other data
  - Contextual information in time-series
  - Semantic information in log statements

Root cause analysis with log data

- Traditional correlation-based approaches
  - – Enormous false positives of spurious correlation
- Existing causal approaches [1,2]
  - – Only considering logs close to troubles
- ➢ Graph-based causal analysis approach
  - – Efficient analysis
  - – Exploratory approach for all notable events

[1] Z. Zheng et al. "3-Dimensional root cause diagnosis via co-analysis," in ACM ICAC, 2012, pp. 181.

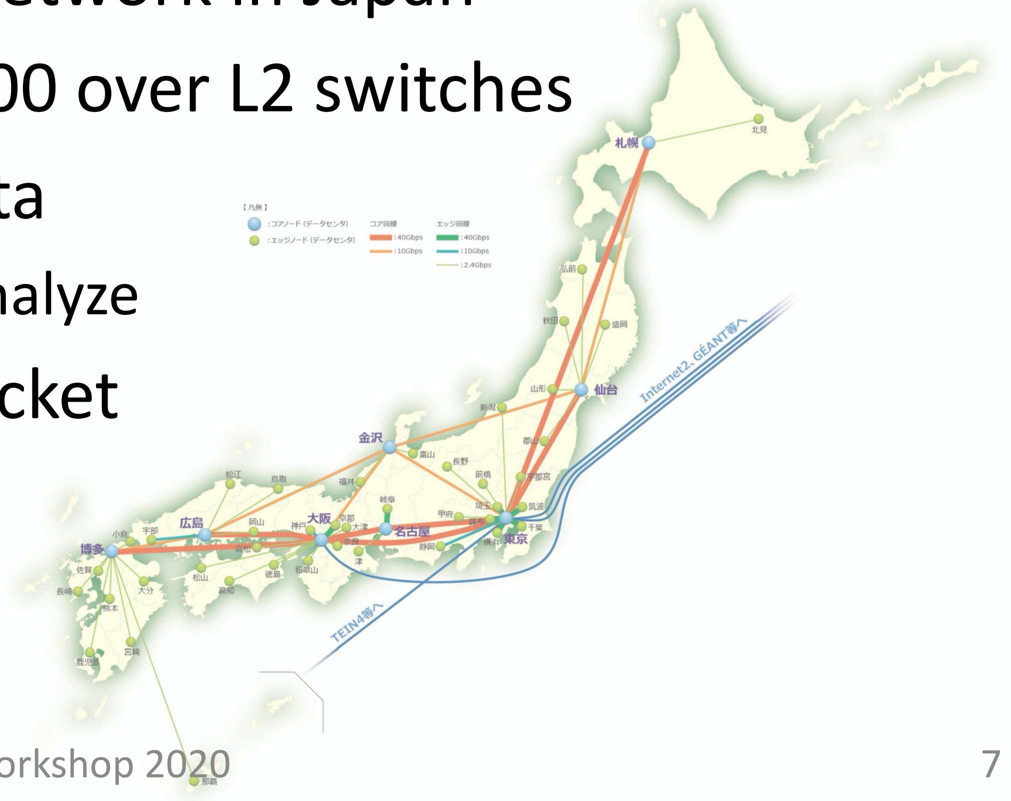
[2] A. Mahimkar et al. "Towards automated performance diagnosis in a large iptv network," in ACM SIGCOMM, 2009, pp. 231–242.

Goal

- Extract contextual information as causal relations among network events in log data
  - Time series analysis + Causal inference
  - Exploratory approach with wide-range data
  - Available in large-scale network
- Support troubleshooting of system failures
  - Operators can understand system behavior easily

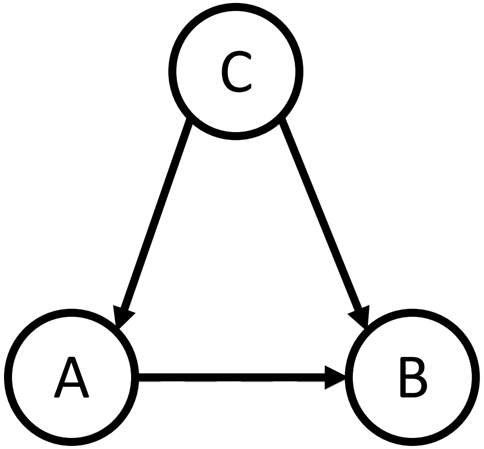
Dataset

- SINET4
  - (https://www.sinet.ad.jp/en/top-en)
  - A nation-wide R&E network in Japan
  - 8 core routers and 100 over L2 switches
  - 15 months syslog data
    - 3.5 million lines to analyze
  - 12 months trouble ticket
    - For evaluations



Causal Inference

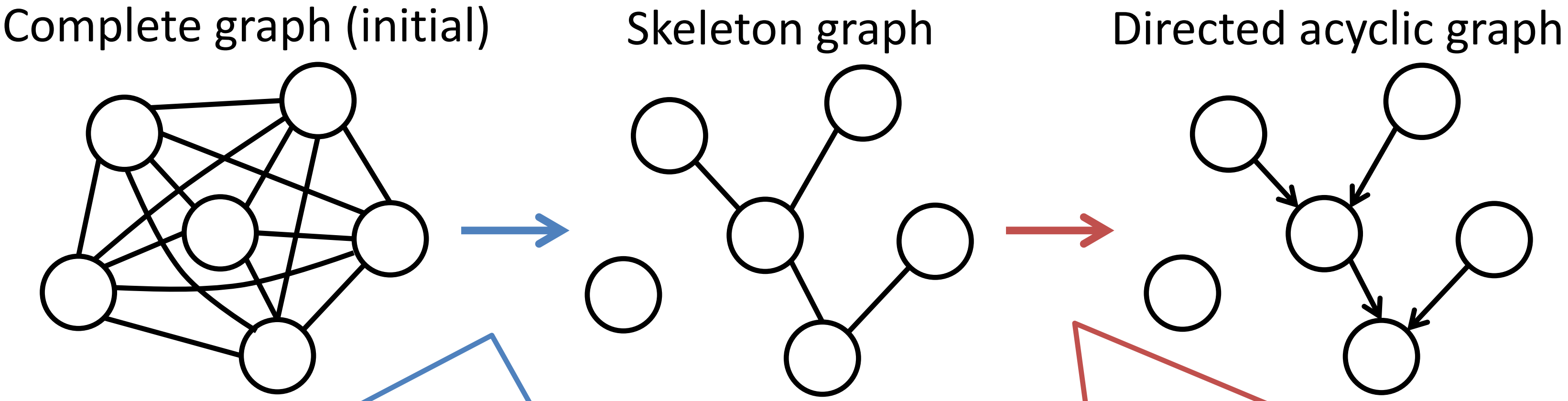
- Conditional Independence
  - A and B are independent if the effect of confounder C is excluded
  - A and B are conditionally independent given C
- PC algorithm [3]
  - Directed acyclic graph (DAG)
  - Explore conditional independence and remove false edges



$$P(A|C)P(B|C) = P(A,B|C)$$

[3] P. Spirtes et al. "An algorithm for fast recovery of sparse causal graphs”, Social science computer review, vol. 9, pp. 62–72, 1991.

Flow of PC algorithm



Complete graph (initial)

Skeleton graph

Directed acyclic graph

- Remove edges of conditional independence
- Statistical test for conditional independence
  - G2 test (for binary or multi-level data) [4]

ops) [5]

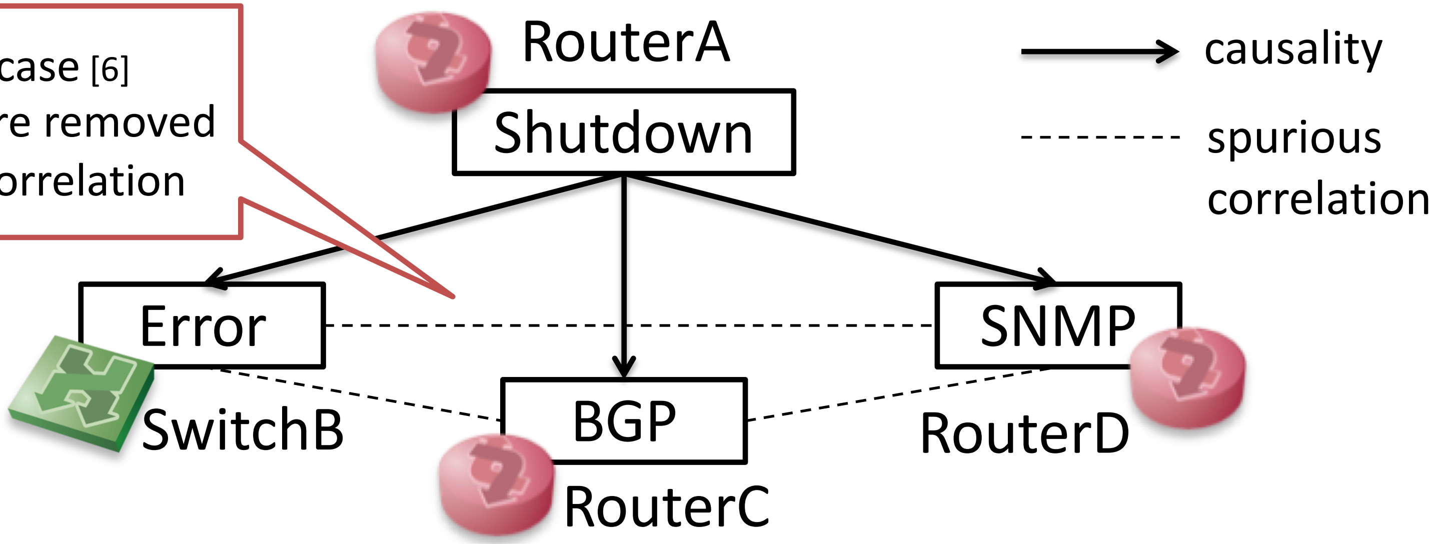
[4] R. E. Neapolitan. "Learning Bayesian Networks." Prentice Hall Upper Saddle River, 2004.

[5] T. Verma, et al. "An algorithm for deciding if a set of observed independencies has a causal explanation". In Proceedings of UAI'92, pp. 323–330, 1992.

Log analysis and causal inference

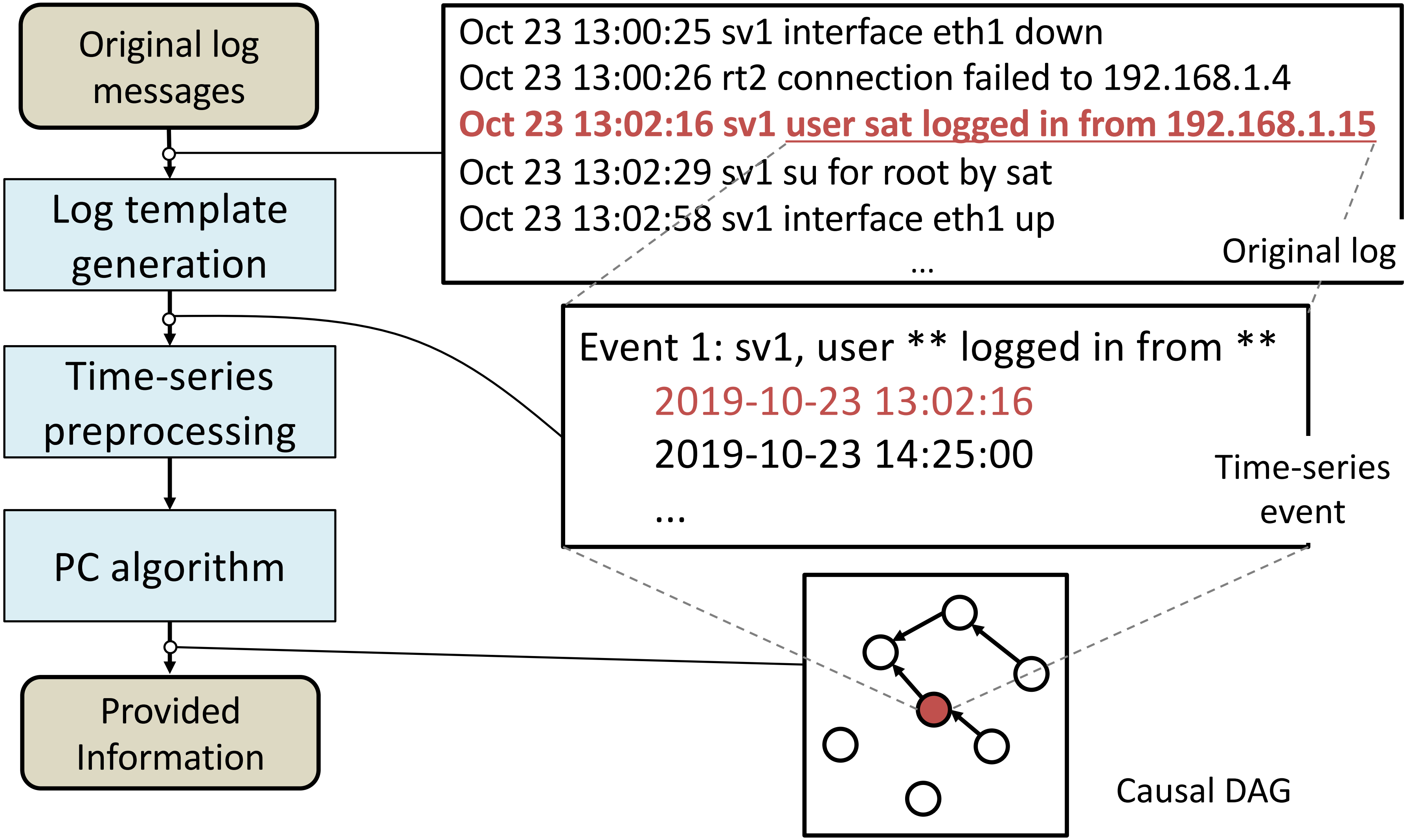
```
Feb 18 17:00:00 routerA System shutdown by root
Feb 18 17:00:05 switchB Error detected on eth0
Feb 18 17:00:15 routerC BGP state changed from Established to Idle
Feb 18 17:00:15 routerD SNMP trap sent to routerA
……
```

In SINET4 case [6]
99.1% edges are removed
as spurious correlation



[6] S. Kobayashi et al. "Mining causality of network events in log data", IEEE TNSM, vol. 15, no.1, pp. 53–67, 2018.

Causal analysis with network logs



Original log messages

Log template generation

Time-series preprocessing

PC algorithm

Provided Information

```
Oct 23 13:00:25 sv1 interface eth1 down
Oct 23 13:00:26 rt2 connection failed to 192.168.1.4
Oct 23 13:02:16 sv1 user sat logged in from 192.168.1.15
Oct 23 13:02:29 sv1 su for root by sat
Oct 23 13:02:58 sv1 interface eth1 up
                         …
```

Original log

```
Event 1: sv1, user ** logged in from **
        2019-10-23 13:02:16
        2019-10-23 14:25:00
        …
```

Time-series event

Causal DAG

Challenges for causal log analysis

- 3 main challenges
  1. How to generate time-series from log data?
     - ➢ Generate log templates as event classifier [7]
  2. How to decrease false positive edges?
     - ➢ Preprocessing periodic time series [6]
  3. How to obtain causality in reasonable time?
     - ➢ Use network domain knowledge for pruning [8]

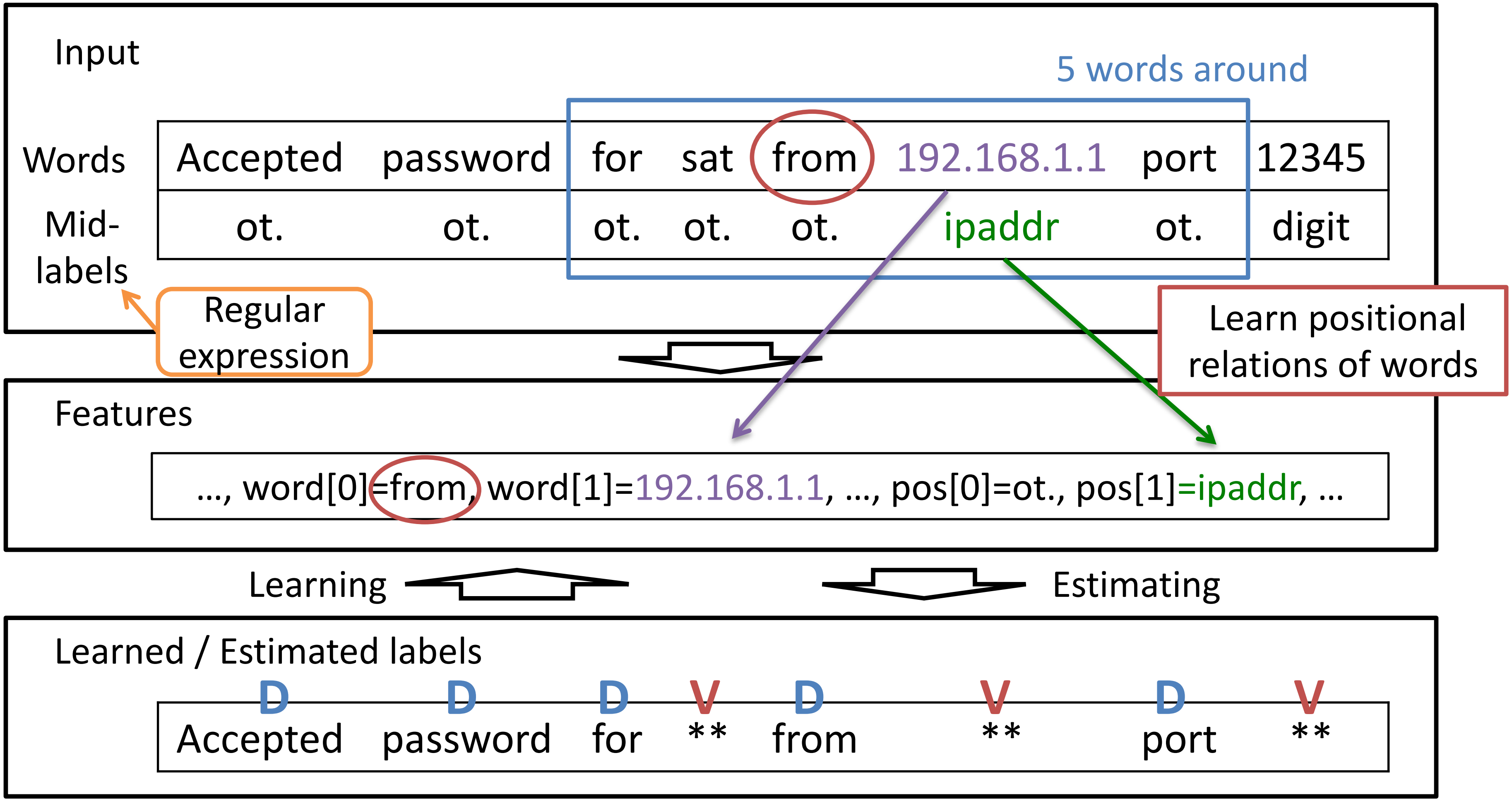
[7] S. Kobayashi et al. “Towards an NLP-based Log Template Generation Algorithm for System Log Analysis”, CFI, 2014.
[6] S. Kobayashi et al. “Mining causality of network events in log data”, IEEE TNSM, vol. 15, no.1, pp. 53–67, 2018.
[8] S. Kobayashi et al. ”Causal analysis of network logs with layered protocols and topology knowledge”, CNSM, 2019

1. Log template generation

- Existing approaches [9,10]
  - – Clustering log messages
  - – Largely depends on log appearance distribution
    - Fails in minor log events (i.e., troubles and anomalies)
- Proposed method [7]
  - – Supervised learning of log template structure with Conditional Random Fields (CRF)

[7] S. Kobayashi et al. “Towards an NLP-based Log Template Generation Algorithm for System Log Analysis”, CFI, 2014.
[9] R. Vaarandi. “A data clustering algorithm for mining patterns from event logs”. In IEEE IPOM , pp.119-126, 2003.
[10] M. Mizutani. “Incremental mining of system log format”. In IEEE SCC’13, pp. 595–602, 2013.

CRF-based log template estimation



Input

5 words around

Words	Accepted	password	for	sat	from	192.168.1.1	port	12345
Mid-labels	ot.	ot.	ot.	ot.	ot.	ipaddr	ot.	digit

Regular expression

Learn positional relations of words

Features

..., word[0]=from, word[1]=192.168.1.1, ..., pos[0]=ot., pos[1]=ipaddr, ...

Learning

Estimating

Learned / Estimated labels

D	D	D	V	D	V	D	V
Accepted	password	for	**	from	**	port	**

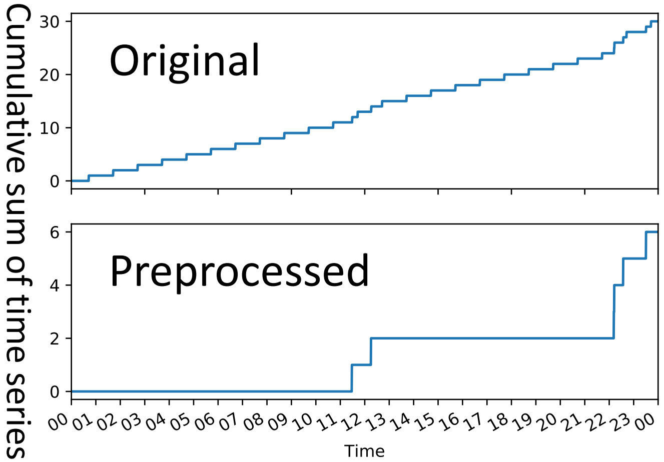
Description / Variable

2. Time-series preprocessing

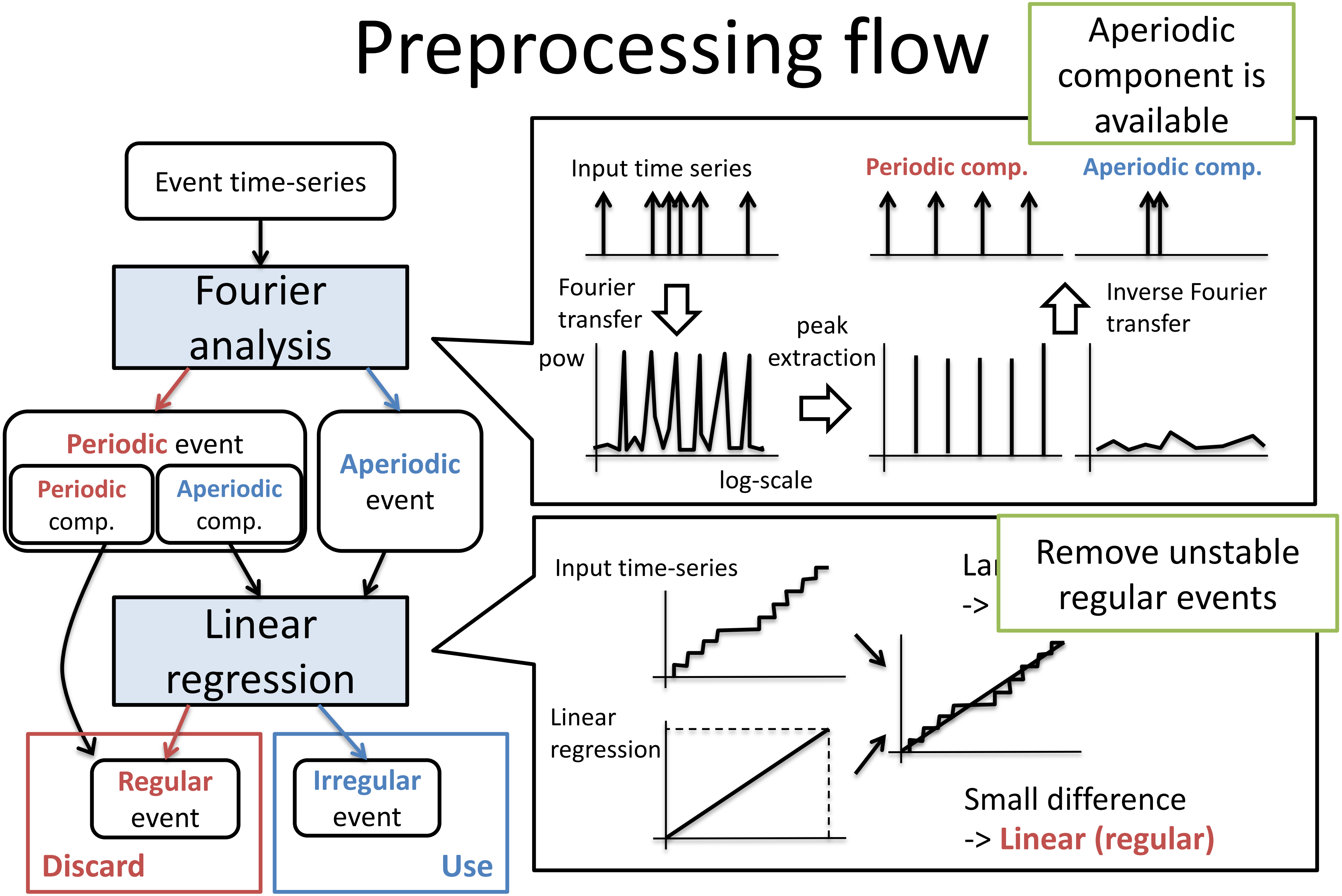
- Periodic time-series (e.g., CRON log) cause spurious correlation that cannot be removed with conditional independence
  - – Events with same interval <- All correlated

➢ Extract aperiodic components from periodic time-series

  - – Focus on irregular behaviors
  - – Use Fourier analysis



Preprocessing flow



Aperiodic component is available

Event time-series

Fourier analysis

Periodic event

Periodic comp.

Aperiodic comp.

Aperiodic event

Linear regression

Regular event

Discard

Irregular event

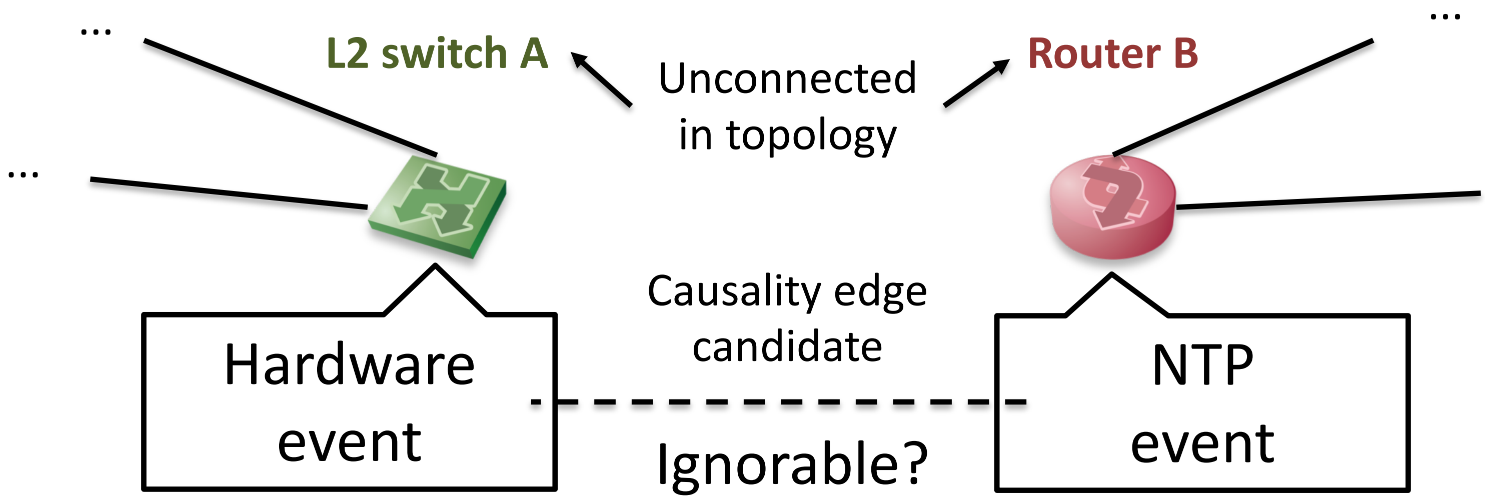
Use

Remove unstable regular events

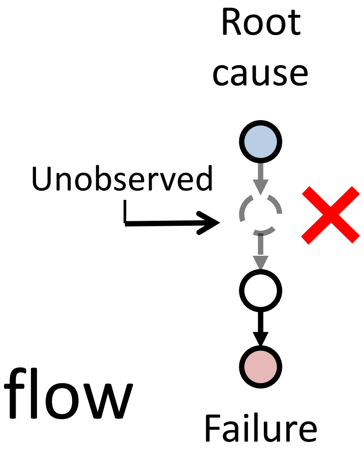
Small difference
-> **Linear (regular)**

3. Graph pruning with network domain knowledge

- Causal analysis takes large processing time
  - Combination explosion of confounders
- Pruning causal edge candidates that is clearly not causality according to domain knowledge

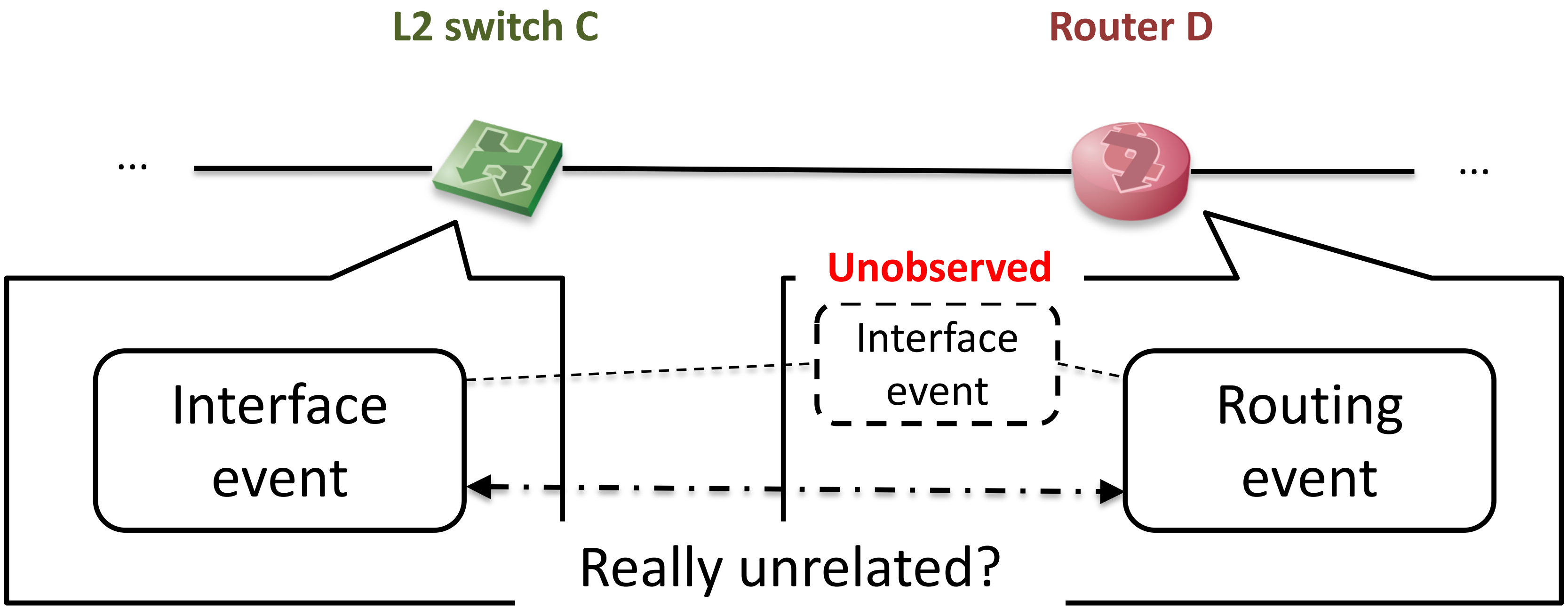


Difficulty in pruning



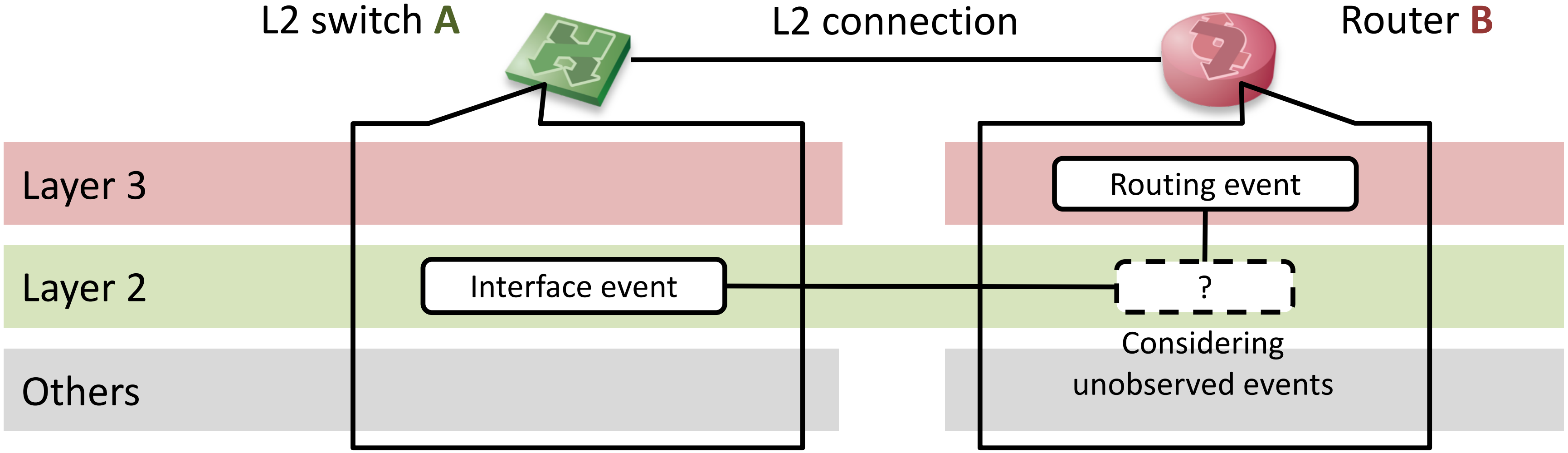
- **Unobserved events** mediate causality
  - – Pruning mediated causality breaks causal flow

-> How to determine the criteria?



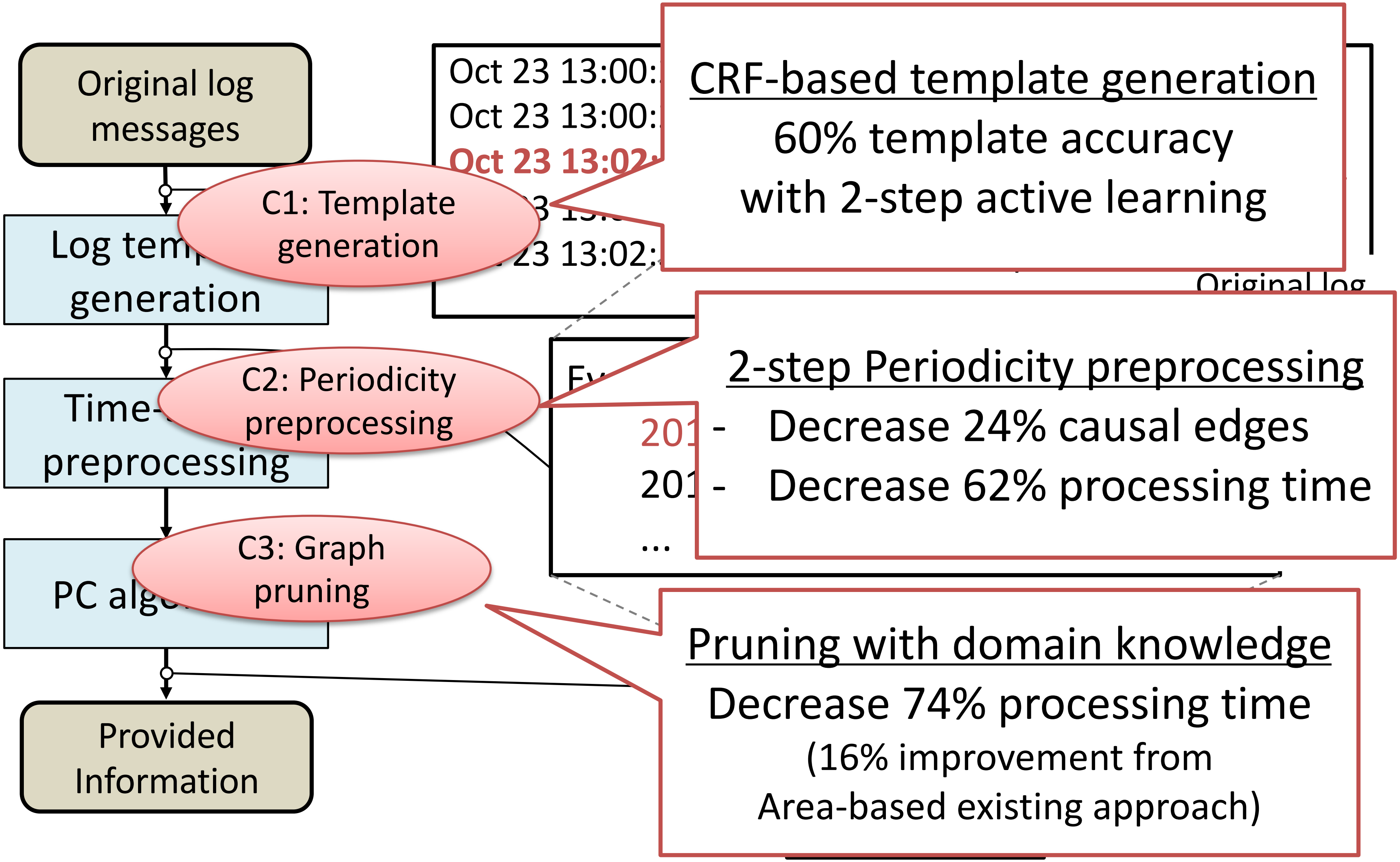
Proposed method of edge pruning

- 2 hueristic rules for edge pruning
  1. Caused events are in same device, or in same functional layer and in connected devices
  2. 1 (or 0) unobserved event can mediate causality



-> Rules satisfied, remaining

Causal analysis with network logs



Original log messages

Log temp generation

Time- preprocessing

PC algo

Provided Information

C1: Template generation

C2: Periodicity preprocessing

C3: Graph pruning

Oct 23 13:00:

Oct 23 13:00:

Oct 23 13:02:

3 13:

23 13:02:

Original log

Ev

201

201

...

CRF-based template generation

60% template accuracy with 2-step active learning

2-step Periodicity preprocessing

- Decrease 24% causal edges
- Decrease 62% processing time

Pruning with domain knowledge

Decrease 74% processing time

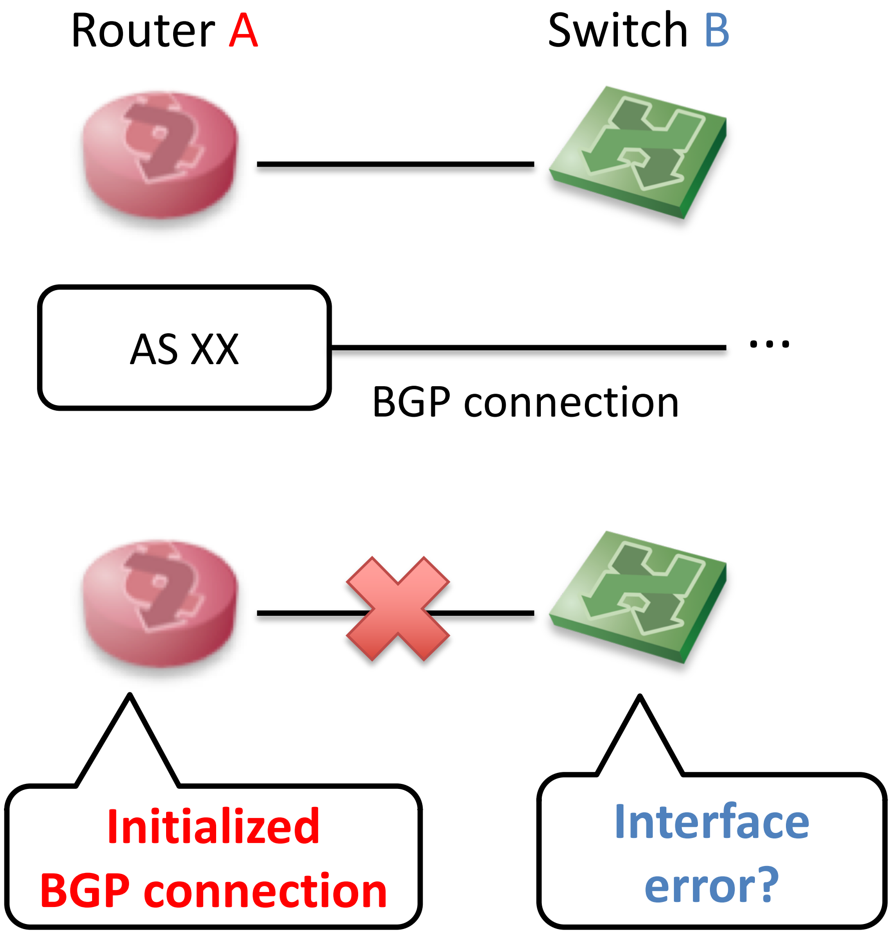
(16% improvement from Area-based existing approach)

Evaluation

- 15 months log data for causal analysis
  - 3.5 million lines (1,789 Log templates, 131 hosts)
- Generate causal DAG for every 1-day data
  - ➢27,668 causal edges in total [8]
- Investigate detected causal edges
  - Case study
  - Comparison with trouble ticket

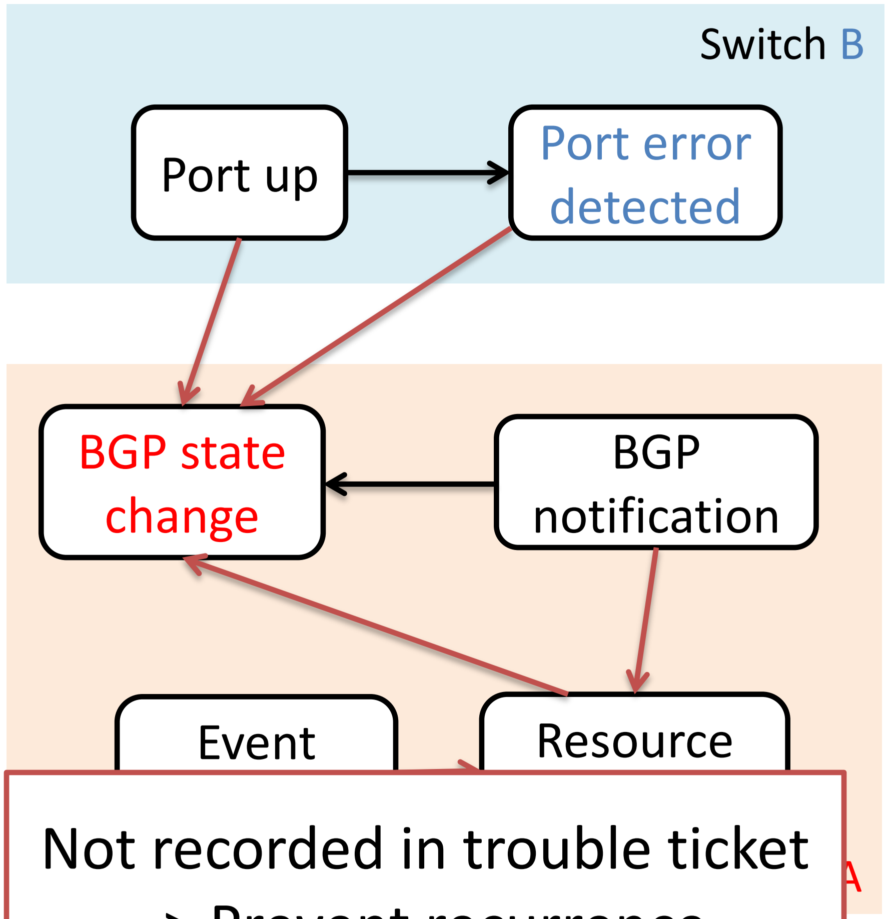
[8] S. Kobayashi et al. ”Causal analysis of network logs with layered protocols and topology knowledge”, CNSM, 2019

Example of detected DAGs



Initialized BGP connection

Interface error?



Not recorded in trouble ticket
-> Prevent recurrence

→ Anomalous

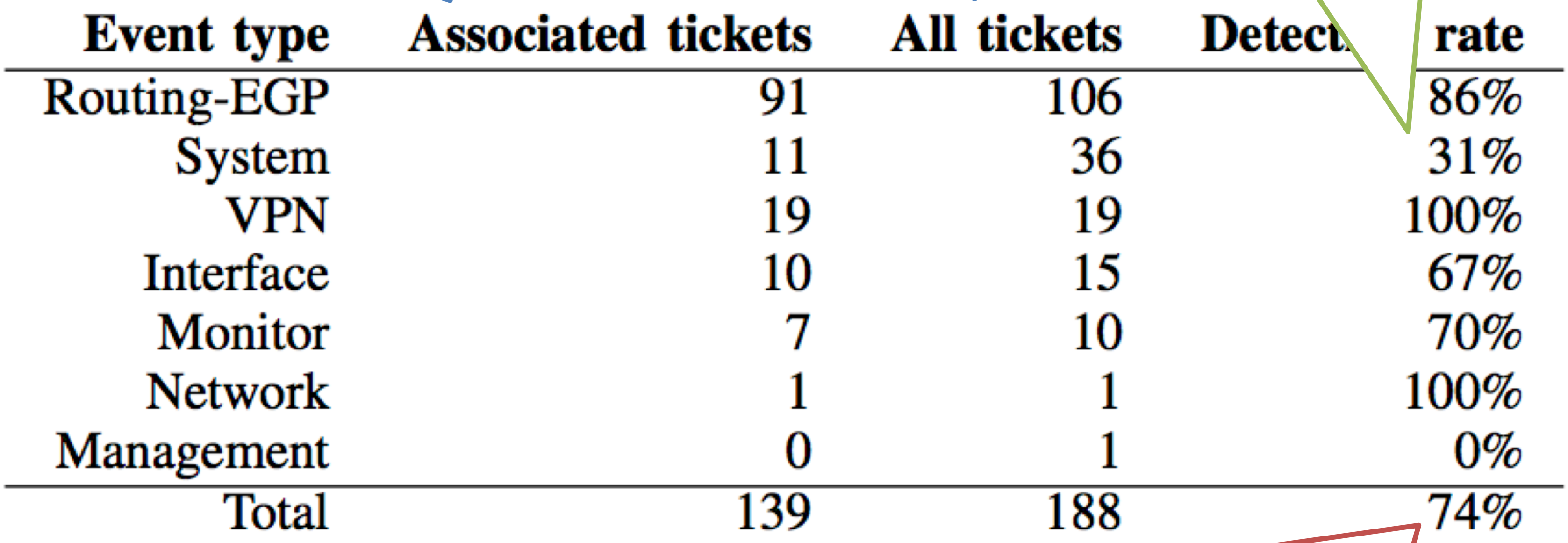
Comparison with trouble tickets

- Detectability of causality related to tickets

Tickets with related causal edges

Tickets with related log messages

One-off events
-> difficult to detect



Event type	Associated tickets	All tickets	Detect rate
Routing-EGP	91	106	86%
System	11	36	31%
VPN	19	19	100%
Interface	10	15	67%
Monitor	7	10	70%
Network	1	1	100%
Management	0	1	0%
Total	139	188	74%

Manually labeled event type

Provide valuable information in major parts of troubles

Conclusion

- Causation mining in network logs
  - Estimate causal DAG with PC algorithm
  - Consider 3 challenges of log causal analysis
- Evaluation with large-scale network logs
  - Detect useful information for troubleshooting
- Future works
  - Consider semantic information in log messages
- https://github.com/cpflat/logdag