Sawmill: Extracting Data for Causal Diagnosis of Large Systems

Markos Markakis[1], Brit Youngmann[2], Trinity Gao[1], Ziyu Zhang[1], Rana Shahout[3], Peter Baile Chen[1], Chunwei Liu[1], Ibrahim Sabek[4], Michael Cafarella[1]

[1]MIT, [2]Technion, [3]Harvard University, [4]University of Southern California

MIT CSAIL

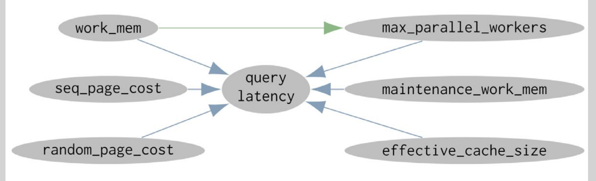
Finding Failure Causes from Logs is Hard!

- In large distributed systems, failures are ***common*** [1], and they must be resolved from ***observational data*** like system logs.
- Operations teams' goal is to ***most efficiently fix the problem***, which requires finding the ***strongest cause of a failure.***
- Ideal setting to apply ***causal reasoning*** and calculate ***Average Treatment Effects (ATEs)***.

```
20:24:44 INFO u0 q34 Running CREATE INDEX midx ON metrics (id);
20:32:25 INFO u0 q35 Running SELECT * FROM metrics WHERE id=562;
20:32:26 INFO u0 q35 Ran in 607.31ms
20:32:28 INFO u0 q36 Running SELECT * FROM metrics WHERE id=555;
20:33:28 INFO u0 q36 Query timed out
```

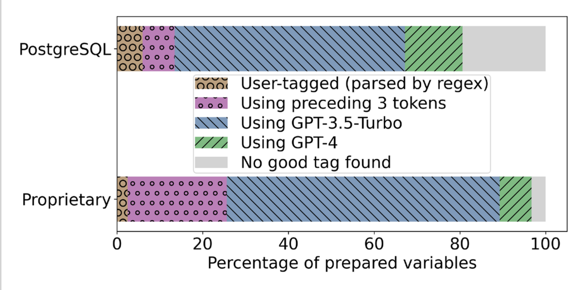
- However, we must bridge the ***available data*** and the ***requirements of causal reasoning*** using Pearl's model [2]:
  - ***Challenge A: Deriving the Schema***
    How can we derive a tabular, human-understandable dataset from log?
  - ***Challenge B: Distilling the Data***
    How can we distill useful features out of the log-derived tabular dataset?
  - ***Challenge C: Obtaining a Causal Model***
    How can we efficiently construct a causal model over the distilled features?

User	M: Free Memory	I:Index Presence	L:Latency Mean (ms)	D: Data Size (GB)	T: Timeouts per day
U_0	67.80 %	1	637.02	64.41	56852
U_1	80.96 %	0	372.60	38.07	29164
...	...	...	...	...	...



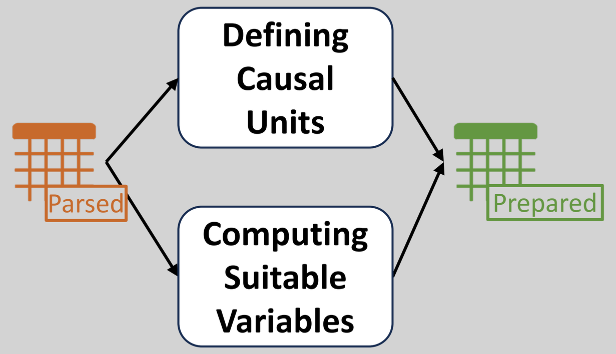
Challenge A: Turning Logs into Understandable Tables

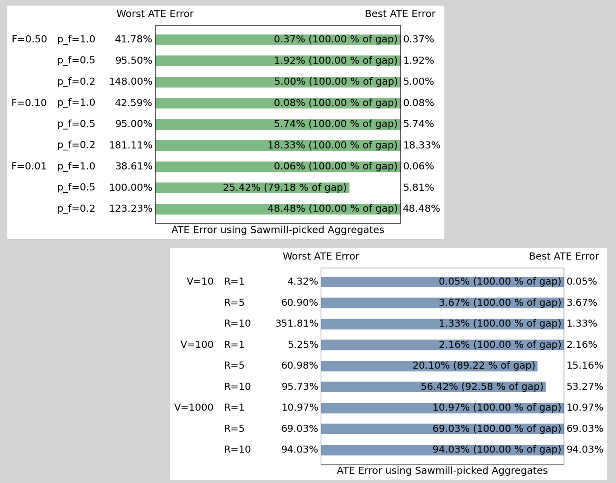
- The textual format of logs is ***unsuitable*** for automated analysis.
- Step 1A: ***Log Parsing***
  - Determine ***log template*** and ***parsed variables*** for each line.
  - Create the ***parsed table.***
  - Off-the-shelf algorithms for this part [3].
- Step 1B: ***Parsed Variable Tagging***
  - Assign ***human-understandable tag*** to each variable.
  - Leverage preceding log template tokens and GPT-3.5-Turbo/GPT-4 [4-5].



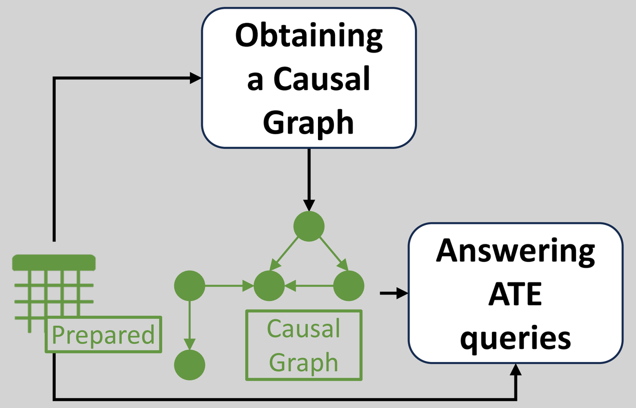
Challenge B: Summarizing Tables Usefully

- Log information is often ***too granular*** for the desired level of reasoning.
- Step 2A: ***Defining Causal Units***
  - User can specify granularity of analysis – e.g. per user, per region or per machine.
- Step 2B: ***Prepared Variable Computation***
  - The information in the parsed table is ***aggregated*** for each causal unit.
  - Appropriate aggregates are selected based on the ***variable type.***
- Step 2C: ***Prepared Variable Selection***
  - Only keep ***one*** aggregated prepared variable per parsed variable.
  - Maximize potential downstream usefulness by picking the variable that ***maximizes empirical entropy.***





Challenge C: Obtaining a Causal Graph



Dataset		PC	FCI	LiNGAM	GIN	GRaSP	GES	Exact Search	GPT-4
PostgreSQL		✓	✓	✗	▲	✓	✓	✗	▲
Proprietary		✓	✓	✗	▲	●	●	✗	▲
XYZ	$V = 10$	✓	✓	✓	✓	✓	✓	▲	▲
	$V = 100$	●	●	✓	▲	▲	▲	▲	▲
	$V = 1000$	▲	▲	✗	▲	▲	▲	▲	▲

✓: non-empty graph ▲: 30-minute timeout
●: empty graph ✗: error

- Causal analysis requires a model of variable interactions expressed as a ***causal graph.***
- Difficult to obtain over log variables:
  - Hand crafting it is daunting based on the ***large number of variables.***
  - Inferring it automatically using causal discovery is ***not reliably fast/correct enough*** because of variable dependencies [5-13].
- We instead propose ***Exploration-based Causal Discovery:***
  - User gives a ***variable of interest.***
  - Sawmill suggests ***candidate causes*** for it, based on the data in the prepared table.
  - User uses ***domain expertise*** to add real causes to the causal graph.
  - Repeat to increase ***exploration score.***

Evaluation

- We compared Sawmill against two baselines:
  - A simple ***Regression***-based approach that does not leverage causality.
  - An approach relying on ***GPT-4*** [5] to suggest candidate causes.
- We used three log datasets representing ***different tradeoffs between realism and ground-truth effect certainty***:
  - A dataset derived from ***real executions of TPC-DS on PostgreSQL*** with different parameter settings.
  - A ***real log dataset*** from an HTTP-based client-server application, with an ***injected causal relationship*** of varying magnitude and noisiness.
  - A ***synthetic log dataset*** with a varying number of variables and noisiness.

Accuracy: Sawmill's mean MRR is ***41.89%*** higher than that of the next best baseline (Regression), while Sawmill's mean ATE Error is ***10.99%*** lower than that of the next best baseline (Regression).

Dataset			Sawmill MRR	Regression MRR	AskGPT MRR
PostgreSQL			**0.5667**	0.0476	0.4815
Proprietary	F=0.5	p_f=1.0	**1.0000**	1.0000	0.0000
		p_f=0.5	**1.0000**	1.0000	0.3333
		p_f=0.2	**1.0000**	1.0000	1.0000
	F=0.1	p_f=1.0	**1.0000**	1.0000	0.0000
		p_f=0.5	**1.0000**	1.0000	0.0000
		p_f=0.2	**1.0000**	1.0000	0.0000
	F=0.01	p_f=1.0	**1.0000**	1.0000	0.0714
		p_f=0.5	**1.0000**	0.0667	0.0000
		p_f=0.2	**1.0000**	0.0667	0.0000
XYZ	V=10	R=1	**0.6667**	0.6667	0.4007
		R=5	0.6111	0.5556	**0.6667**
		R=10	**0.6667**	0.5833	0.0664
	V=100	R=1	**0.6667**	0.5476	0.5000
		R=5	**0.6667**	0.5370	0.0000
		R=10	0.3889	**0.6667**	0.5000
	V=1000	R=1	**0.6667**	0.0000	0.1667
		R=5	0.6667	0.0000	**0.8333**
		R=10	**0.6667**	0.0000	0.1667
Mean on Proprietary			**1.0000**	0.7926	0.1561
Mean on XYZ			**0.6296**	0.3952	0.3667
Mean			**0.8018**	0.5651	0.2730

Dataset			True ATE	Sawmill ATE	Regression ATE	AskGPT ATE
Proprietary	F=0.5	p_f=1.0	258.43	**257.47**	273.01	0.00
		p_f=0.5	114.86	**112.66**	118.04	112.66
		p_f=0.2	28.71	**27.28**	25.94	27.28
	F=0.1	p_f=1.0	258.43	**258.64**	256.01	0.00
		p_f=0.5	114.86	121.45	**119.38**	0.00
		p_f=0.2	28.71	**33.98**	35.30	0.00
	F=0.01	p_f=1.0	258.43	**258.57**	264.50	258.57
		p_f=0.5	114.86	**85.66**	84.79	0.0
		p_f=0.2	28.71	**42.64**	45.18	0.0
XYZ	V=10	R=1	2.00	**2.00**	2.00	2.00
		R=5	2.00	2.11	**2.10**	2.11
		R=10	2.00	1.97	**1.98**	1.97
	V=100	R=1	2.00	**1.96**	1.95	2.36
		R=5	2.00	**1.60**	1.58	0.00
		R=10	2.00	0.87	0.86	**0.97**
	V=1000	R=1	2.00	**1.78**	0.37	0.00
		R=5	2.00	**0.62**	-1.61	0.62
		R=10	2.00	0.12	**0.35**	0.00
Mean % Error on Proprietary				**11.72%**	14.64%	67.44%
Mean % Error on XYZ				**28.83%**	47.88%	49.50%
Mean % Error				**20.27%**	31.26%	58.47%

Dataset			Parse (s)	Prepare (s)	ExploreCandidateCauses (s)	Total Time (s)	Total Time over Log Size (s/MiB)
PostgreSQL			37.65	4.41	4.85	46.91	2.39
Proprietary	F=0.5	p_f=1.0	189.19	48.58	2.62	240.39	1.06
		p_f=0.5	189.51	48.83	3.20	241.54	1.07
		p_f=0.2	195.58	49.32	3.18	248.08	1.10
	F=0.1	p_f=1.0	194.50	49.11	3.22	246.83	1.09
		p_f=0.5	189.53	49.18	3.24	241.95	1.07
		p_f=0.2	189.58	49.50	3.28	242.36	1.07
	F=0.01	p_f=1.0	190.70	49.99	3.31	244.00	1.08
		p_f=0.5	187.09	49.47	3.38	239.94	1.06
		p_f=0.2	191.18	49.30	3.43	243.91	1.08
XYZ	V=10	R=1	72.33	4.26	2.40	78.99	1.32
		R=5	80.04	5.27	2.98	88.29	1.48
		R=10	79.69	4.70	3.09	87.48	1.47
	V=100	R=1	145.67	33.24	3.59	182.50	2.95
		R=5	144.14	33.87	3.87	181.88	2.94
		R=10	144.67	33.79	3.97	182.43	2.95
	V=1000	R=1	853.14	326.83	8.38	1188.35	18.90
		R=5	822.43	334.47	8.91	1165.81	18.54
		R=10	849.08	335.59	15.09	1199.76	19.08
Mean			260.30	82.09	4.53	346.92	4.30
Breakdown (%)			75.03%	23.66%	1.30%		

Computational Efficiency: Sawmill only requires an average of ***346.92 s*** to go from a log to an ATE, ***75.03%*** of which is required for log parsing. Sawmill's performance scales linearly with log complexity.

Human Efficiency: Sawmill only requires ***6-10 user interactions*** to leverage causality, up to 5 more than the best baseline, Regression.

Dataset	System	Parse	Separate	SetCausalUnit	Prepare	Explore CandidateCauses	Accept	GetATE	Regress	GPT-Explore CandidateCauses	Total
PostgreSQL	Sawmill	1	1	1	1	2	3	1	0	0	10
	Regression	1	1	1	1	0	0	0	1	0	**5**
	AskGPT	1	1	1	1	0	3	1	0	2	10
Proprietary	Sawmill	1	0	1	1	1	1	1	0	0	6
	Regression	1	0	1	1	0	0	0	1	0	**4**
	AskGPT	1	0	1	1	0	1	1	0	1	6
XYZ	Sawmill	1	0	1	1	2	3	1	0	0	9
	Regression	1	0	1	1	0	0	0	1	0	**4**
	AskGPT	1	0	1	1	0	3	1	0	2	9

References

[1] Gupta, Saurabh, et al. "Failures in large scale systems: long-term measurement, analysis, and implications." *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. 2017.
[2] Pearl, Judea. *Causality*. Cambridge university press, 2009.
[3] He, Pinjia, et al. "Drain: An online log parsing approach with fixed depth tree." *2017 IEEE international conference on web services (ICWS)*. IEEE, 2017.
[4] OpenAI. "Models". *https://platform.openai.com/docs/models/* , 2024.
[5] Achiam, Josh, et al. "Gpt-4 technical report." *arXiv preprint arXiv:2303.08774* (2023).
[6] Spirtes, Peter, and Clark Glymour. "An algorithm for fast recovery of sparse causal graphs." *Social science computer review* 9.1 (1991): 62-72.
[7] Spirtes, Peter, Clark N. Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2000.
[8] Shimizu, Shohei, et al. "A linear non-Gaussian acyclic model for causal discovery." *Journal of Machine Learning Research* 7.10 (2006).
[9] Xie, Feng, et al. "Generalized independent noise condition for estimating latent variable causal graphs." *Advances in neural information processing systems* 33 (2020): 14891-14902.
[10] Lam, Wai-Yin, Bryan Andrews, and Joseph Ramsey. "Greedy relaxations of the sparsest permutation algorithm." *Uncertainty in Artificial Intelligence*. PMLR, 2022.
[11] Chickering, David Maxwell. "Optimal structure identification with greedy search." *Journal of machine learning research* 3.Nov (2002): 507-554.
[12] Silander, Tomi, and Petri Myllymaki. "A simple approach for finding the globally optimal Bayesian network structure." *arXiv preprint arXiv:1206.6875* (2012).
[13] Kıcıman, Emre, et al. "Causal reasoning and large language models: Opening a new frontier for causality." *arXiv preprint arXiv:2305.00050* (2023).