Hound: Causal Learning for Datacenter-scale Straggler Diagnosis

Pengfei Zheng

Duke University

Department of Computer Science

pfzheng@cs.duke.edu

Benjamin C. Lee

Duke University

Department of Electrical and Computer Engineering

benjamin.c.lee@duke.edu

ABSTRACT

Stragglers are exceptionally slow tasks within a job that delay its completion. Stragglers, which are uncommon within a single job, are pervasive in datacenters with many jobs. We present Hound, a statistical machine learning framework that infers the causes of stragglers from traces of datacenter-scale jobs. Hound is designed to achieve several objectives: datacenter-scale diagnosis, unbiased inference, interpretable models, and computational efficiency. We demonstrate Hound's capabilities for a production trace from Google's warehouse-scale datacenters and two Spark traces from Amazon EC2 clusters. (This work is accepted as a full paper for oral presentation at *SIGMETRICS'18* and published in *POMACS, Volume2, Issue 1, March 2018* [8].)

KEYWORDS

datacenter; distributed system; performance modeling; performance diagnosis; machine learning; causal reasoning; topic modeling

ACM Reference Format:

Pengfei Zheng and Benjamin C. Lee. 2018. Hound: Causal Learning for Datacenter-scale Straggler Diagnosis. In *SIGMETRICS '18 Abstracts: ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems Abstracts, June 18–22, 2018, Irvine, CA, USA*. ACM, New York, NY, USA, 3 pages. <https://doi.org/10.1145/3219617.3219641>

1 INTRODUCTION

Stragglers threaten performance from datacenter-scale parallelism. Datacenters split a computational job into many tasks, execute them in parallel on many machines, and aggregate results when the last task completes.¹ Stragglers are exceptionally slow tasks within a job that significantly delay its completion. Unfortunately, stragglers' effects increase with the number of tasks and scale of the system.

Existing straggler mitigation strategies are incomplete solutions because they address symptoms rather than diagnose causes. Although profilers support straggler diagnosis by producing massive datacenter traces, deriving causal explanations from complex

¹A job contains one or more tasks that execute a single program on multiple data in parallel. When frameworks, such as Apache Spark, organize tasks into stages, each stage corresponds to a job.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SIGMETRICS '18 Abstracts, June 18–22, 2018, Irvine, CA, USA

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-5846-0/18/06.

<https://doi.org/10.1145/3219617.3219641>

datasets is difficult. Existing diagnosis procedures rely heavily on human expertise in systems and application of best practice, which are laborious and fail to scale. Thus, understanding stragglers' causes is a prerequisite for efficient countermeasures and large datasets motivate methods for systematic causal analysis.

We present a statistical machine learning framework – Hound – that infers stragglers' causes with three a three-stage (base, meta and ensemble) learning process. We demonstrate Hound on two representative systems. The first is a month-long trace from a production Google datacenter with 12K machines running 650K jobs and 25M tasks [6]. The second is a pair of traces from Amazon EC2 clusters running Spark data analytics [5]. Hound produces interpretable topics that describe stragglers' causes. Each topic is a set of system conditions that combine to explain a major source of stragglers that affects many jobs. Results show that Hound's inferred causes consistent with those from expert analysis.

2 SYSTEM OBJECTIVES

Datacenter-scale Diagnosis. Datacenter operators benefit from a broad view of stragglers. A datacenter-scale perspective reveals recurring problems that cause stragglers in many jobs instead of minor problems that impact just a few. Hound uses base and meta learners to identify datacenter-scale patterns. First, Hound's base learner for each job constructs latency models and reveals system conditions associated with poor task performance. Second, from base learners' models, Hound's meta learner discovers recurring, straggler-inducing conditions shared by many jobs.

Unbiased Inference. A statistical model makes assumptions that prevent it from performing well on all datasets. Relying on one method is risky when analyzing large, heterogeneous datasets such as datacenter traces. Hound uses ensemble learning to reduce risk of biased conclusions. An ensemble combines responses from multiple, independent learners with a majority rule that amplifies correct responses and avoids erroneous ones. We design an ensemble that combines distinct but complementary methods for diagnosing stragglers.

Interpretable Models. Hound ensures interpretability with topic models. Latent Dirichlet Allocation identifies features that often appear together in jobs' latency models [2]. A cluster of features correspond to a topic, which reveals clear and concise system conditions that are associated with atypically large latencies. When jobs suffer from diverse causes of stragglers, Hound reports a mix of relevant topics and assesses their relative contributions to system performance.

Computational Efficiency. Statistical inference can be computationally expensive and even intractable. The computational complexity of constructing a model for straggler diagnosis depends

on three factors: number of profiled metrics per task, number of tasks per job, and number of jobs. Datacenter traces can include millions of jobs and tasks, each with tens of profiles. Hound relies on learning methods that have polynomial complexity and are amenable to parallelization in a distributed system.

3 MODEL DESIGN

Figure 1 shows Hound’s learning process. Base learning captures relationships between latency and system conditions. Meta learning reveals recurring topics. Ensemble learning integrates results from disparate types of models.

Inputs – Job and Task Profiles. Hound requires profiles from jobs that include stragglers. Profiles collected throughout the datacenter and across time comprise a dataset. Hound uses the dataset to infer models that take latency as the dependent variable and system conditions as independent variables.

Outputs – Causal Topics. Hound’s topics are concise causal explanations for stragglers. Each topic is a set of abnormal profile metrics associated with stragglers across many jobs. For example, the topic [CACHE_MISS(+), CPI(+)] indicates some stragglers arise from poor cache behavior, measured in terms of the number of cache misses and average cycles per instruction. Suffixes (-) or (+) indicate whether metric’s values are significantly lower or higher for stragglers than those for normal tasks. Hound identifies relevant topics for each job and estimates each topic’s significance across the datacenter’s tasks and jobs. Administrators can use these outputs to easily identify significant causes and triage anomalies.

Base Learning. Base learners infer relationships between task latency and profiled metrics for each job. Because relying on a single learner could induce bias and produce false causes, Hound uses an ensemble of heterogeneous learners to discover predictive, dependence, and causal relationships.

- **Predictive (PR).** Hound’s PR learner constructs regression models with Bagging Augmented ElasticNet (BAE). ElasticNet is a regularization method that automatically selects significant metrics [9]. Bagging [3] is a machine learning method that improves model accuracy and stability. Hound combines these methods to fit latency models using profiled metrics. It encodes resulting regression models as vectors of coefficients to quantify metrics’ predictive power and summarize potential causes of stragglers.
- **Dependence (DP).** Hound’ DP learner uses statistical dependence to assess the association between latency and profiled metrics. We survey state-of-the-art dependence measures and favor Schweizer-Wolff Dependence (SWD) [1], but this measure is unsigned and computationally expensive. We address SWD’s limitations, creating the Signed Schweizer-Wolff Dependence (SSW) with the Fréchet-Hoeffding Bound [4] and reducing its computational cost with sampling.
- **Causal (CA).** Hound’s CA learner constructs Rubin Causal Models (RCM) [7]. RCM estimates the causal effect of one metric on latency while eliminating bias induced by other metrics. In RCM, causal effect is usually estimated with logistic regression, but this approach is vulnerable to collinearity, over-fitting, and outliers. We address these challenges by introducing a more accurate and robust causal effect estimator, AdaBoost Inverse Probability Weighting (AdaBoost-IPW).

Figure 1: The Hound Framework

Meta Learning. Hound’s meta learner uses topic models (Latent Dirichlet Allocation [2]) to identify recurring straggler causes from jobs’ numerous models and causality profiles. First, Hound populates a dictionary with metrics and identifies metrics that often appear together in jobs’ causality profiles. Metrics that are more prominent in these profiles produce corresponding words that appear more frequently in documents. Second, Hound identifies a mix of relevant topics for each job.

Ensemble Learning. Hound’s ensemble learner reconciles potentially divergent topics from multiple learners. The ensemble emphasizes topics found by multiple learners and drops those found by only one. First, Hierarchical Agglomerative Clustering (HAC) identifies similarities in prediction, dependence, and causation topics. Clusters’ centroids define the ensemble’s topics. Second, Hound identifies relevant ensemble topics for each job. Weights for ensemble topics are averages of those for prediction, dependence, and causation topics.

4 EVALUATION AND RESULTS

Experiment. We implement Hound in Apache Spark and evaluate two systems. First, we use a month-long trace from a production Google cluster that contains 12K servers and computes for 650K diverse jobs [6]. Second, we evaluate Hound on a pair of Spark

Topic	Keywords	Weights	Interpretation
E_0	MEM_ASSIGN(+), MEAN_MEM(+),	0.5, 0.25,	Data Skew
	PEAK_MEM(+)	0.25	
E_1	PAGE_CACHE(+), MEM_ASSIGN(+),	0.45, 0.17,	Data Skew
	PAGE_CACHE_UM(+),	0.38	
E_2	DISK_SPACE(+)	1.0	Data Skew
E_3	MEAN_CPU(+), PEAK_CPU(+)	0.52, 0.48	Computation Skew
E_4	PEAK_IO(+), MEAN_IO(+)	0.51, 0.49	I/O Skew
E_5	MEAN_CPU(-), PEAK_CPU(-)	0.8, 0.2	Limited Processor
E_6	MEAN_MEM(-), PEAK_MEM(-)	0.83, 0.17	Limited Memory
E_7	MEAN_IO(-)	1.0	Limited I/O
E_8	PEAK_IO(-), MEAN_IO(-)	0.83, 0.17	Limited I/O
E_9	CACHE_MISS(+), CPI(+)	0.54, 0.46	Cache Bottleneck
E_{10}	SCHED_DELAY(+)	1.0	Queueing Delay
E_{11}	EVICT(+)	1.0	Eviction Delay
P_{12}	FAIL(+)	1.0	✖
C_{12}	MACHINE_RAM(+)	1.0	✖

Table 1: Causal topics revealed from an ensemble of predictive, dependent, causal models. Topics with “✖” are revealed by individual models but rejected by the ensemble.

Cause	Coverage	Dominant Coverage
Data Skew (E_0, E_1, E_2)	73.6%	55.0%
Limited Processor (E_5)	39.2%	12.1%
Cache Misses (E_9)	32.6%	7.0%
Limited I/O (E_7, E_8)	36.7%	6.6%
Queueing Delay (E_{10})	20.0%	5.1%
Limited Memory (E_6)	13.6%	2.7%
Computation Skew (E_3)	31.2%	2.2%
Eviction Delay (E_{11})	3.80%	0.90%
I/O Skew (E_4)	5.60%	0.60%

Table 2: Coverage statistics for causes

traces collected from Amazon EC2 clusters and compare Hound’s inferred causes to those from expert diagnoses in a prior study of this data [5]. We summarize Google findings and refer the reader to our extended article for Spark findings [8].

Causal Discovery. Table 1 presents Hound’s inferred topics that explain stragglers in Google’s datacenter. Each topic is a weighted combination of keywords, which are measured metrics followed by a suffix, drawn from models. The suffix “(+)” or “(-)” means the metric is significantly larger or smaller for stragglers than for normal tasks. Keywords and topics produce natural interpretations and reveal causes of stragglers. Hound’s ensemble guards against false conclusions, which may be consistent with operator intuition but arise from biased models. The following causes are concise and consistent with domain-specific expertise.

- (1) *Load Imbalance.* Tasks with more work are more likely to be stragglers. Tasks that use more memory (E_0), larger page caches (E_1), or more local storage (E_2) may be computing on more data. Tasks that require more processor time (E_3) or I/O transactions (E_4) may have more work.
- (2) *Resource Constraints.* Tasks that use fewer resources are likely to perform poorly. Atypically low processor and memory use suggest poor progress (E_5, E_6), and atypically low I/O rates suggest constrained data supply (E_7, E_8). Atypically

low and high processor usage are both causes for stragglers (E_3, E_5), but these conditions do not arise simultaneously.

- (3) *Microarchitectural Activity.* Cache misses increase the average number of cycles required per instruction (E_9).
- (4) *Scheduling Problems.* The cluster manager’s decisions affect task completion. Queueing delays cause stragglers by extending tasks’ end-to-end latency (E_{10}). Eviction delays cause stragglers as tasks halt computation on overloaded machines, re-launch on another machine, and lose progress (E_{11}).

Mixtures of Causes. Hound determines which jobs suffer from which causes. Hound assigns each job a weighted mix of causes. Weights estimate the fraction of stragglers attributed to each cause. We found that 12% of jobs require one cause, 33% of jobs require two causes, 36% of jobs require three causes, 16% of jobs requires four causes and 3% of jobs require five causes.

We propose a series of measures to understand mixtures of causes. We say a cause is *dominant* when it explains the majority of a job’s stragglers. We define a cause’s *coverage* as how often it explains some of a job’s stragglers. We define a cause’s *dominant coverage* as how often it explains the majority of a job’s stragglers, which is a more selective measure and smaller than coverage. Topics with greater coverage are relevant for more jobs and provide greater utility to system operators. We find that when multiple causes are required, one is usually dominant.

Table 2 summarizes coverage and dominant coverage. The top three causes—data skew, limited processor utilization, cache misses—explain the majority of stragglers in 74.1% of jobs. 73.6% of jobs experience some form of data skew. Many jobs also suffer from low resource usage or cache behavior. Relatively minor causes include memory availability, task eviction and I/O skew.

ACKNOWLEDGMENTS

This work is supported by the National Science Foundation under grants CCF-1149252 (CAREER), CCF-1337215 (XPS-CLCCA), SHF-1527610, and AF-1408784 as well as by STARnet, a Semiconductor Research Corporation program, sponsored by MARCO and DARPA.

REFERENCES

- [1] E. F. Wolff B. Schweizer. 1981. On Nonparametric Measures of Dependence for Random Variables. *The Annals of Statistics* 9, 4 (1981), 879–885. <http://www.jstor.org/stable/2240856>
- [2] David M Blei. 2012. Probabilistic topic models. *CACM* 55, 4 (2012), 77–84.
- [3] Leo Breiman. 1996. Bagging predictors. *Machine learning* 24, 2 (1996), 123–140.
- [4] Roger B Nelsen. 2007. *An Introduction to Copulas*. Springer Science & Business Media.
- [5] Kay Ousterhout, Ryan Rasti, Sylvia Ratnasamy, Scott Shenker, and Byung-Gon Chun. 2015. Making Sense of Performance in Data Analytics Frameworks. In *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*. USENIX Association, 293–307.
- [6] Charles Reiss and John Wilkes. 2011. Google cluster-usage traces: format+ schema. *Technical Report* (2011).
- [7] Paul R Rosenbaum and Donald B Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70, 1 (1983), 41–55.
- [8] Pengfei Zheng and Benjamin C. Lee. 2018. Hound: Causal Learning for Datacenter-scale Straggler Diagnosis. *Proc. ACM Meas. Anal. Comput. Syst.* 2, 1, Article 17 (April 2018), 36 pages. <https://doi.org/10.1145/3179420>
- [9] Hui Zou and Trevor Hastie. 2005. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society* 67, 2 (2005), 301–320.