Paper Review

“Correlating instrumentation data to system states: A building block for automated diagnosis and control”

1. Summary

This paper mainly talked about TAN (Tree-Augmented Bayesian Networks). The paper used TAN to implement a classifier to determine whether the system state is low performant or high performant. A service-level objective (SLO) is a critical element of a service-level agreement (SLA) between service providers and customers. With the more complex design of systems, a naive Bayesian classifier can help maintain large systems.

This paper chooses several system metrics, such as memory, network, CPU time, user level, IO, and database performance, different from previous works that only use CPU usage. The naive Bayesian classifier can calculate a joint probability distribution by the metrics input. Because the metrics of the system can have an interval impact on each other, it is feasible to use the Bayesian model for these random variables. In the experiment part of this paper, the author tried hundreds of different metrics, and the result is that a few precise metrics is enough and can produce better results.

Since I have little knowledge and experience in building a Bayesian model, I focus on the process of confounding variables. This paper made a priori fix to the model to reveal and exclude some relationships between metrics and variables. After the experiment, the paper has filtered the top related variables to the SLO state.

This paper builds a test platform with Apache and adds a payload step by step to test the results and iterate TAN. From my point of view, we can introduce chaos testing instead, which can add more randomness and diversity to the system.

From the results, we learned that a TAN consisting of several variables can precisely predict the state of SLO online with 87% to 94% accuracy and can flexibly adjust by changing the SLO threshold.

2. Advantages

+ TAN can provide a precise prediction.
+ TAN is flexible to port to other systems and introduce different metrics.

3. Disadvantages

– They are collecting data on a real machine. Using a VM can provide a more comfortable environment and provide more precise and more data.
– Nowadays, stateless and serverless services run on a large cluster and have the self-recover ability. We need to consider the communication metrics when building TAN in nowadays services.

4. Brainstorming

We have a new idea for performance scheduling and scaling. We want to create a scheduler powered by reinforcement learning. Our scheduler can time-efficiently or cost-efficiently make the program run on specific cluster nodes by collecting traces from historical data. Our future work is to collect specific data from our university's cluster and use a reinforcement learning program to find the correlation and in what node can a distributed task run most efficiently with other existing tasks.

We are at one's wit's end with which data we need to collect and whether this metric will work in the reinforcement learning algorithm. After reading this paper, we may list many metrics first and collect. We can find our answers with prior processing, rules, and upcoming analysis.