Distributed applications are composed on many services, each of which is complex. The applications can have millions of users using them leading to billions of in-progress interactions in the system at any time. This scale and complexity makes identifying runtime errors and debugging them hard. To tackle this, engineers collect data about an application's execution called a trace. OpenTelemetry is a community-driven project to define a common format and tooling to collect trace data.

Trace data is collected by instrumenting an application at interesting points. Network calls, exceptions, conditional statements are examples of interesting points. Historically, the instrumentation would be manually inserted by the programmer. However, as application grow more complex, automatic instrumentation is gaining popularity [Mace et al.(2018)]. The OpenTelemetry project has auto-matic instrumentation for many open-source libraries commonly used by developers. In this project, you will evaluate the overhead of this instrumentation for python.

要求

Assemble a set of benchmark programs to evaluate the instrumentation overhead. Evaluate the instrumentation on these programs using different sampling ratios. Analyze the results.

Implement an advanced sampling technique technique such as tail sampling [Zhang et al.(2023)] for one language's instrumentation. Evaluate the performance.

对使用 opentelemetry 和不使用的 overhead 进行对比，可以比较响应时间，CPU 占用率，内存占用率等指标

用 python 写，检测的程序在 GitHub 上找。