Programming on Cloud (Fall 2022) - Assignment 1

Individual or Group of 2 Assignment

Due by October 30th 23:59

Problem Statement

This assignment aims to practice the concepts, and techniques for data models and the communications for resources represented by data models.

The data set is from a Github project, under the **directory of Workload Data.**

https://github.com/haniehalipour/Online-Machine-Learning-for-Cloud-Resource-Provisioning-of-Microservice-Backend-Systems

The workload data contains the workload generated from two industrial benchmarks NDBench from Netflix and Dell DVD store from Dell. Both benchmarks are deployed on a cluster of cloud VMs on AWS and Azure clouds. The workload has been split to training sets and testing sets for machine learning purpose.

In each of the workload file, the first 4 columns contain the following attributes.

CPUUtilization_Average,NetworkIn_Average,NetworkOut_Average, MemoryUtilization_Average

In this assignment, please develop a client/server program to serve a "workload query" scenario. In this scenario, a client sends a 'Request For Workload (RFW)', and the server replies an 'Response for Data (RFD)' for each conversation.

The client's RFW includes:

1. RFW ID
2. Benchmark Type (such as DVD store or NDBench)

3. Workload Metric (such as CPU or NetworkIn or NetworkOut or Memory)
4. Batch Unit (the number of samples contained in each batch, such as 100)
5. Batch ID (such as the 1st or 2nd or… 5th Batch)
6. Batch Size (such as the how many batches to return, 5 means 5 batches to return)
7. Data Type (training data or testing data)
8. Data analytics (10p, 50p, 95p, 99p, avg, std, max, min), for example 50p means 50th percentile

The server's RFD reply includes:

1. RFW ID
2. The last Batch ID
3. The data samples requested
4. The data analytics

This assignment is responsible for the design of the data model, and implementation of the data communication. There is no need to develop a full-fledged database system. Data can be stored in files or any kinds of storage, such as relational databases or nosql databases.

Technical Requirement

1. Data Communication

The data should be communicated between the client and server through data serialization/deserialization in **two methods**, namely **text based (de)-serialization and binary (de)-serialization.** For example,

(1) XML or JSON can be used for text based (de)-serialization.(2) Protocol Buf or Thrift can be used for binary (de)-serialization.

For each method, your program should be able to retrieve the samples requested for each RFW.

2. Programming Language

You can program this application in any language.

3. Application

Your client/server can be a standalone program or you build on any software framework that supports client/server. You can choose the protocol your prefer TCP, or HTTP.

5. Testing and Deployment (updated on October 4th 2022)

*due to requests from the class for the difficulties of finding team members, the revision is as follows. For those who have single member or two-member team. There is no additional requirements. For those who have three members, there is extra workload.

5.1 Single member solution can use localhost for communication; if 5.2 is developed, it will considered as bonus.

5.2 Two-member solution should run your server program on a cloud instance (e.g. AWS instance) or with in a cloud platform (e.g. Google App Engine).

5.3 Three-member solution should following 5.1 and in addition to develop unit testing cases for both the client and server.

Other options can be discussed with the lecturer.

Submission

The deliverables include the following artifacts and they should be submit to moodle site

. 1) Pack all your source code in a single zip file. .gz .tar or .zip are acceptable. Please do NOT use .rar file. The file should have this naming convention **[STUDENT1 ID_STUDENT2_ID]_A1_source.zip.**

. 2) The complete data model files for each method (XML, JSON, Proto and etc). Please follow the naming convention **[STUDENT1 ID_STUDENT2_ID]_data.zip.**

. 3) A report in PDF with the naming convention **[STUDENT**

ID]_A1_report.pdf that includes the following sections. The report should follow the format of IEEE publication. https://www.ieee.org/conferences_events/conferences/publishing/templates.html You can either use Word or Latex template. Make your report within 4 pages for the sections below.

Section Structure of Report

i. Data model design

ii. Data serialization/de-serialization method

iii. Technical implementation of response, request and (de)serialization

For example, libraries or software packages you choose to deal with request/response, and data serializations (e.g. pros or cons given your experience)

iv. Cloud deployment of your server code

v. Instructions on how to run both the client and server applications

vi. Screenshots of running your application with SUCCESSFUL results.

Marking Criteria

. 1) Quality of the design of data models (20 marks)

. 2) Executable applications that fulfills the function of data query. [35 marks]

. 3) Cloud deployment (25 marks)

. 3) Quality of the report –The required items are addressed in clear description with detailed information provided. [20 Marks]