COMP0235 Coursework challenge

Introduction

In the field of biochemistry there have recently been large advances in predicting protein structure. A group of biochemist researchers at UCL Medical school would like to make predictions of possible 3D structures of proteins in the Human Genome. The have written a small data analysis pipeline, comprising 4 steps that is capable of making such predictions. A prediction for each protein in their pipeline takes 13 minutes and they have 41,339 proteins they could analyse. They calculate that on their computer with just one CPU core, it would take over a year to make all the predictions they want.

They have approached you to help turn their data analysis program in to a distributed analysis system which can run their predictions in a timely fashion. They can provide you with their python scripts. Their pipeline makes use of two machine learning predictors; S4Pred and HHSearch. To test the system with the researchers give you a list of 6,000 proteins they are most interested in and they they would like you to analyse first.

The researcher's pipeline and data

The researchers provide you with two files. One file containing all the proteins in the human proteome name **uniprotkb_proteome_UP000005640_2023_10_04.fasta.gz** and a second file with their pipeline python code **code.tar.gz**.

The code file includes 3 items. Two python scripts and a text file with 6,000 protein IDs. The python script, **pipeline.script.py** runs the data analysis pipeline they have written. The python script **results_parser.py** is a short piece of code that their pipeline script requires. The **experiment_ids.txt** contains the list of protein IDs for the subset of 6,000 proteins they would like you to run predictions for.

If you open **pipeline.script.py** you can follow the logic of their analysis. At the start of the process the script reads in all the proteins from a fasta file. Then for each file it runs the following 4 steps.

1. Run the s4pred ML tool
2. Rewrite the input sequence to include the s3pred predictions
3. Run the HHSearch ML tool
4. Parse the HHSearch output

The 4th step outputs the results that the researchers want to capture.

Coursework task

In this coursework you are required to build a distributed pipeline across your cloud machines that will run the 4 steps in the `pipeline_script.py` it should accomplish this in distributed fashon across your mini-cluster of 7 machines (one host and 6 clients). You are free to accomplish this as you see fit. Your solution should include the following features

1. Should use an appropriate configuration system. We have covered Ansible and Salt but others are available.
2. Make use of an apporpriate datastore for the complete human proteome contained in file **uniprotkb_proteome_UP000005640_2023_10_04.fasta.gz**. This should be able to return appropriate records from a list of arbitrary protein IDs
3. Make use of appropriate monitoring and logging of your mini-cluster and your data analysis pipeline
4. Should collate the results calculated on the client machines and make them available to the researchers.

You need to collate the following information from step 4 for the researchers, preserving these file formats:

1. A csv file that contains a list of proteins and the identity of the best hit calculated by HHSearch. You can find an example such file in **coursework_example_output/example_hits_output.csv**
2. A file containing the mean Standard Deviation and mean Geometric means for all 6,000 HHSearch runs you calculate (i.e. capture the STD and Gmean values for each pipeline run and take the average across 6,000 runs). You can find an example such file in **coursework_example_output/example_profile_output.csv**

Challenges/hints

1. On your 2 AWS machines we estimate it should take about one to two days to run all the calculations.
2. The host instances are too small to run the calculations
3. EC2 client instances have 4 CPUs
4. You need to be able to understand how to install and run the s4pred and hhsearch programs
5. You need to be able to understand how to fetch the required datasets for s4pred and hhsearch
6. You will need to be able to understand the FASTA data format
7. You should ensure you can successfully run **pipeline.script.py**. This could be on either your own machine or on one of the cloud machines you have access to. In the directory `pipeline_example`. You can find an example input sequence `test.fa`. If you run the script successfully you should produce a number of intermediary files, example of these can be found in the directory. And a final output file

hhr_parse.out. The file you produce should be equivalent (though some figures may have some minor differences)

8. A runtime the Load Average for a Client machine should not exceed 3

Deliverables

1. A link to a github repository that contains all the code you used to accomplish the coursework task. This code should be able install everything needed on your mini-clsuter and run the data analysis
2. Your code repository must include instructions on how to use your code to install AND run your data analysis system. Instructions should assume that someone is starting with brand new host and n (n=6) client machines that have nothing installed. The persona marking you coursework should be able to check out your github to a fresh set of cloud machines, run your setup and analysis processes and produce the two required files on the host machine.
3. A short report (no more than 2,000 words) that explains why you have designed your data analysis system/pipeline the way you have. You should explain the logic of the system and why you have made the decisions you have. For example if you have chosen to use Ansible for machine configuration you should, explain why you have chosen this instead of the many other possibilities (i.e. chef, salt, puppet, etc…)

Pipeline Dependencies

Executables

1. S4Pred - https://github.com/psipred/s4pred
2. HHSuite - https://github.com/soedinglab/hh-suite
3. pipeline_script.py
4. results_parser.py

Data sets

1. pdb70 protein structure sequence dataset for HHSearch - **https://wwwuser.gwdg.de/~compbiol/data/hhsuite/databases/hhsuite_dbs/pdb70_from_mmcif_latest.tar.gz**
2. uniprot human proteome set - **uniprotkb_proteome_UP000005640_2023_10_04**
3. List of fasta IDs to analyse - **experiment_ids.txt**

Python dependencies

1. biopython
2. torch
3. numpy
4. scipy

Background Reading

1. https://en.wikipedia.org/wiki/Human_genome

2. https://en.wikipedia.org/wiki/Protein
3. https://en.wikipedia.org/wiki/Proteome
4. https://en.wikipedia.org/wiki/FASTA_format
5. https://en.wikipedia.org/wiki/Protein_Data_Bank_(file_format)