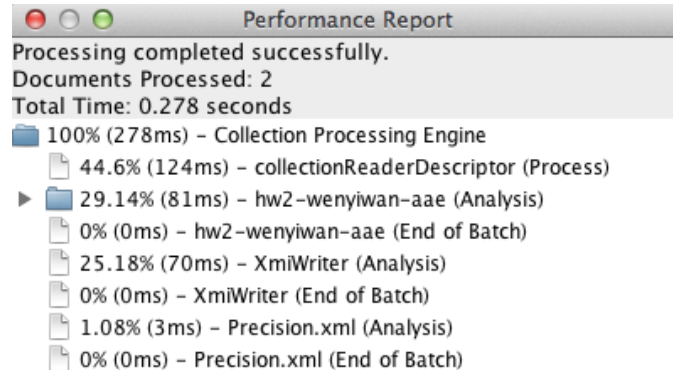# 11-791 Homework 3

Wenyi Wang (Andrew ID: wenyiwan)

Task 1. Execution Architecture with CPE

According to the requirement in the homework, based on homework 2, I added a collection reader and created a CAS Consumer based on the evaluator component, and included it in my CPE pipeline.

The figure below shows the execution result of my CPE on the input data.


This is part is trivial in that we can refer to the existing code in the uima-examples to generate proper code for my own program.

Task 2. Deployment Architecture with UIMA-AS

1. Creating an UIMA-AS client

According to the requirement of the homework, what I did was as follows:

•   create an UIMA-AS client descriptor (scnlp-ID-client.xml) for a remote UIMA-AS service (Stanford CoreNLP), and integrate my client with my CPE pipeline (myCPE2.xml).

•   integrated the Name Entity annotations from the Stanford CoreNLP service into the answer scoring component, and compare the accuracy and the speed with my pipeline in the previous homework.
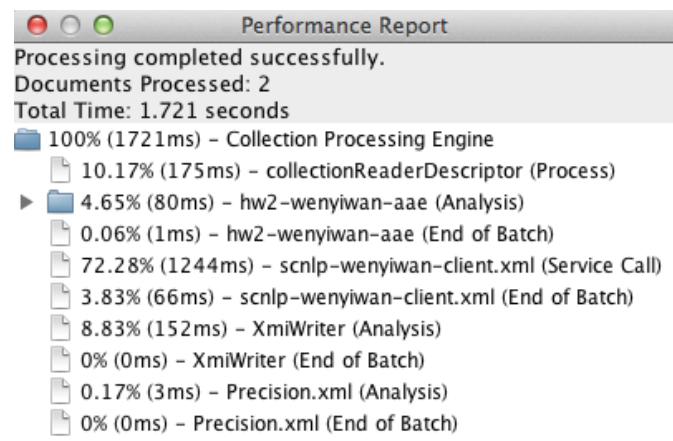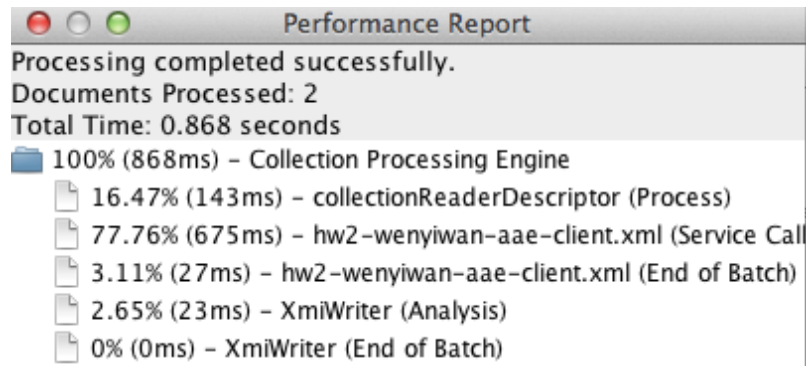

The accuracy stayed the same as that in homework 2 and the speed gets lower. It seems that the named entity module doesn't perform well in this case.

2. Deploying your own UIMA-AS service

According to the instructions in the homework, what I did was as follows:

- created a deployment descriptor and pointed to the AAE in homework;
- started a UIMA-AS broker locally, based on the instruction in the README file, as suggested by Di Wang;
- created a client descriptor for my service, which is similar to the one in the previous section, except that the broker URL is changed to local;
- created a CPE descriptor to test the service by calling the client, which is similar to the previous CPE descriptors.


```
⬤ ◯ ⬤                    Performance Report
Processing completed successfully.
Documents Processed: 2
Total Time: 0.868 seconds
📁 100% (868ms) – Collection Processing Engine
    📄 16.47% (143ms) – collectionReaderDescriptor (Process)
    📄 77.76% (675ms) – hw2-wenyiwan-aae-client.xml (Service Call
    📄 3.11% (27ms) – hw2-wenyiwan-aae-client.xml (End of Batch)
    📄 2.65% (23ms) – XmiWriter (Analysis)
    📄 0% (0ms) – XmiWriter (End of Batch)
```

However, the performance stayed the same and the speed gets higher than the previous case.

Here, I have to say that the in-class recitation provided by TAs really helps with this part. Another important resource for this problem is the solution to the following question on Piazza, by Zhengzhong Liu. The solution really helps with setting the $UIMA_CLASSPATH and assuring the program run correctly.

https://piazza.com/class/hkzplg21ib9t0?cid=115

Bonus

I tried to the Stanford CoreNLP annotator locally, and the speed is much lower than that of the remote one. It basically shows that the remote machine has better computing capability than my local machine. This indicates the significance of this homework, namely, telling us it is important to collaborate, share, and leverage remote resources.

Conclusion

This homework really guides me through the process of using CPE to run a pipeline and using UIMA-AS architecture to call a remote service. It is a good example and preview of how work can be done in collaboration.