11-791 HW3 Report

Zhiyu Li

Oct. 6th, 2013

Name: Zhiyu Li
Andrew ID: zhiyul

1 Summarize

This report includes the CPE and UIMA-AS Architecture design specified in the homework 3 handout.

2 Task 1.2 Creating and Running your CPE

The pipeline is the same with the pipeline in homework 2. Instead of using Document Analyzer as we did in homework 2, we need to use a Collection Reader in this part. I used the org.apache.uima.tools.components.FileSystemCollectionReader, and wrote a Collection Reader descriptor "FileSystemCollectionReader.xml" to wrap it into the CPE. Also, I took the Evaluator Component out of the Analysis Engine and turned it into a CAS Consumer "EvaluatorCasConsumer.xml". When I run the CPE, the average accuracy for the two QAs was 0.733. Running time was 106ms.

3 Task 2.2 Creating an UIMA-AS client

As TA agreed on the piazza (`https://piazza.com/class/hkzplg21ib9t0?cid=119`), we can use NamedEntityMention in this homework to meet the requirement, so in this part I use NamedEntityMention.

I saved the CPE descriptor for this task into task_2_2.xml. You can open the UIMA CPE GUI, load it and run. If you run into any problem running it, please refer to the troubleshooting section below.

The architecture is similar to the previous task. We still use CPE to run this task, the Collection Reader and CAS Consumer parts stay the same, and the only thing we need to change is the Analysis Engine part. I wrote a client descriptor "scnlp-zhiyul-client.xml" to call the remote service and get the data(NamedEntityMention annotation) back. Also, I wrote another Aggregate Analysis Engine "scnlp-aae-client.xml" to process the NamedEntityMention we

got from the server and assign scores to answers accordingly. In the Analysis Engine layer of the CPE task_2_2.xml, I first put in the client descriptor, and then the new Aggregate Analysis Engine. After running the new CPE, the average accuracy I got for the two QAs is 0.675, lower than the pipeline in homework 2. Running time is 1497ms (1293ms were on the service call), much slower than the homework 2 pipeline but acceptable.

4 Task 2.3 Deploying your own UIMA-AS service

Most parts of this task is still similar to the previous tasks. For the CPE, the Collection Reader part and the CAS Consumer part are the same. In the Analysis Engine part, I created a client descriptor to call the service I created locally, and that is it. All we need now is to deploy the Analysis Engine we used in task 1.2 to a local UIMA-AS service.

To deploy our own UIMA-AS service, some prepare work need to be done first. I copied the maven dependency files to a single folder(target) using "copy-dependencies" plugins, then add that folder to the UIMA_CLASSPATH (Of course, if you have not set the UIMA_HOME, JAVA_HOME and PATH environment variables, you need to set them also). After that, I created a deployment descriptor (hw2-zhiyul-aae-deploy.xml) for my aggregate analysis engine (hw2-zhiyul-aae.xml), then start a UIMA-AS broker locally(startBroker.sh), and deploy my service to the local broker(deployAsyncService.sh classes/descriptors/hw2-zhiyul-aae-deploy.xml -brokerURL tcp://ZhiyumatoMacBook-Air.local:61616). After running the CPE, I got the same accuracy as in task 1.2, and the running time is a little slower(231ms).

5 Troubleshooting

When you run the task 2.2 using the UIMA CPE GUI, if you encountered such an error message as in Figure 1, please go to the "Run Configuration" of the UIMA CPE GUI. Under the "Classpath" tab, click the "User Entries" then "Add External JARs". Find where the "cleartk-type-system-1.2.0.jar" is and add it, as shown in Figure 2. Then the error should go away and you can run the task with no problem.

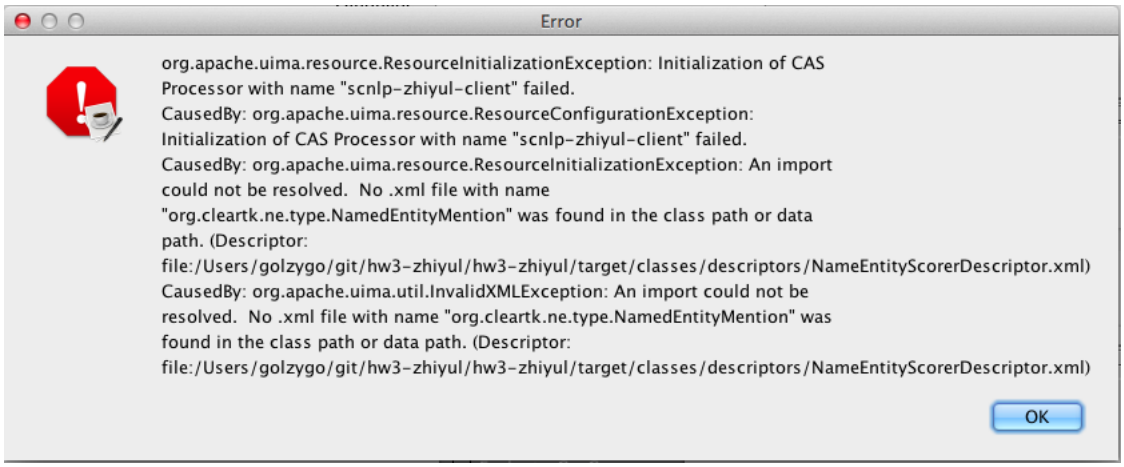

Figure 1: Type system error message.

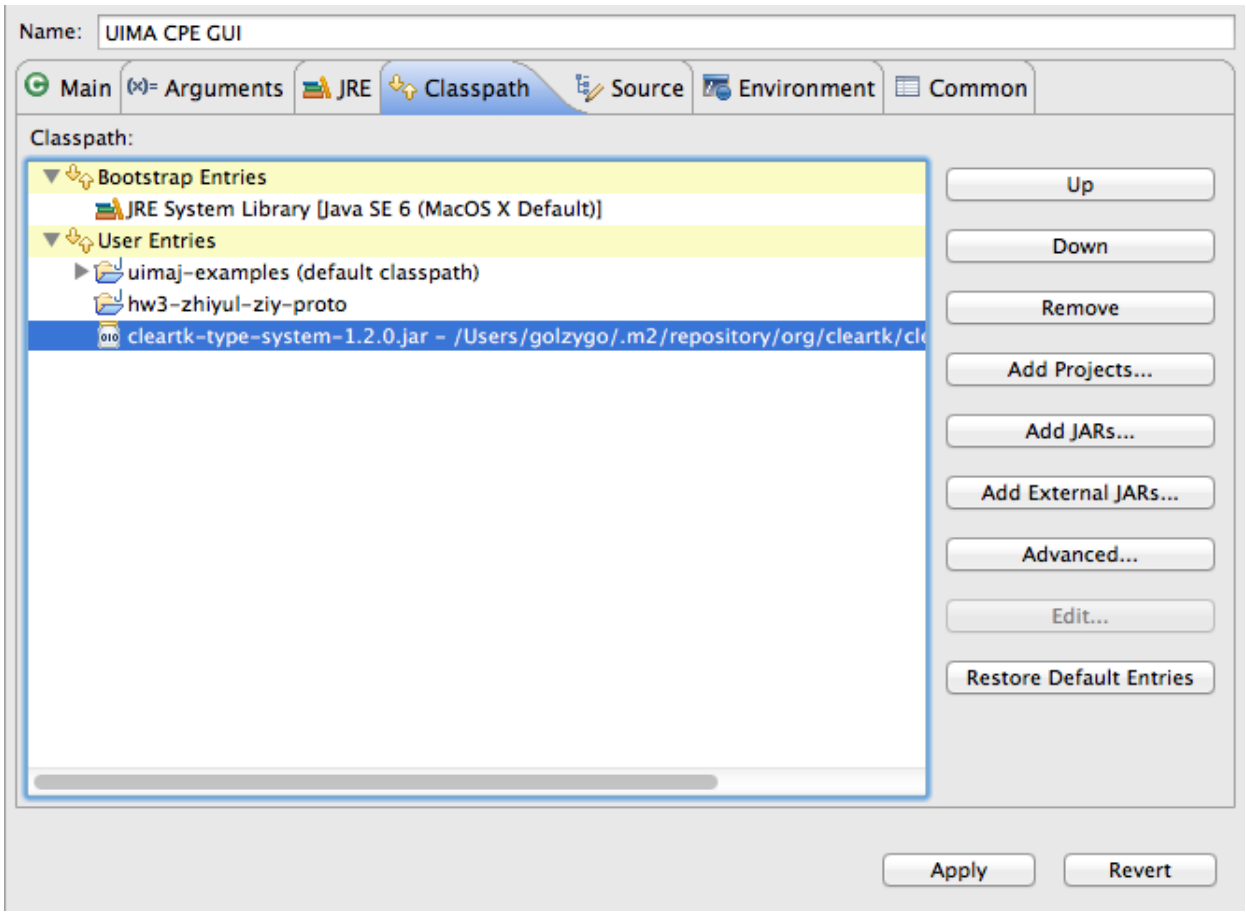

Figure 2: Add the type system jar to the CPE GUI.