11791 Assignment 3 Report

Mengda Yang

mengday

1. Before everything started

I imported the project **`uimaj-examples`** into my eclipse to get all the Run Configurations.

I installed UIMA and UIMA-AS, and set the environmental variables accordingly.

2. Task 1.2

First, I wrote a Collection Reader descriptor **`FileSystemCollectionReader.xml`** in src/main/resources. In this file I set the value of parameter **`InputDirectory`** as this absolute path to meet the requirements of the UIMA CPE GUI:

`D:\Eclipse_Workspace\hw3-mengday\src\main\resources\inputData`

Then, I wrote a Consumer descriptor **`casConsumerDescriptor.xml`** in the same directory. I set the parameter **`OutputDirectory`** as:

`\outputData`

Both descriptors used the default java class provided by **`org.apache.uima.tools.components`**.

Next, I used the Run Configuration UIMA CPE GUI. With the GUI it is easy to generate the target xml for this task.

3. Task 2.2

By modifying the template we can easily create the UIMA-AS client descriptor **`scnlp-mengday-client.xml`**. Due to unstable network connection, I set the timeout to 100 seconds to avoid network failure. To integrate this client into my CPE pipeline, I modified my Aggregate Analysis Engine to call the functions provided by Stanford CoreNLP. By providing this

service the whole document as a CAS, I got the following Name Entity result.

```
Q John  s=0     e=6
John    s=2     e=6
Mary    s=13    e=17
A 1 John        s=20    e=28
1       s=22    e=23
John    s=24    e=28
Mary with all his heart s=35    e=58
Mary    s=35    e=39
his     s=49    e=52
A 1 Mary        s=61    e=69
1       s=63    e=64
Mary    s=65    e=69
John    s=89    e=93
A 0 Mary        s=96    e=104
0       s=98    e=99
Mary    s=100   e=104
John    s=118   e=122
A 0 John        s=125   e=133
0       s=127   e=128
John    s=129   e=133
Mary    s=147   e=151
A 1 John        s=154   e=162
1       s=156   e=157
John    s=158   e=162
Mary    s=169   e=173
```

4. Task 2.3

To create a deployment descriptor, we can right-click, new, and then find the Deployment Descriptor File in the wizard. Then we just load the AAE descriptor following the hints in the GUI, and then it's done.

To start a UIMA-AS broker, we have to call **`startbroker`** in the command lines.

To deploy my service is not very easy.

First I tried using **`org.apache.uima.adapter.jms.service.UIMA_Service`**, no problem. But when I tried to deploy it with command line, it took me a long time and effort to figure out how to correctly set the **`UIMA_CLASSPATH`** and to deploy.

If the deployment is successful, then we can get the following output in the console.

```
Service:QADescriptor Initialized. Ready To Process Messages From Queue:inQ_QADescriptor_1.1
Service:TokenDescriptor Initialized. Ready To Process Messages From Queue:inQ_TokenDescriptor_1.2
Service:NGramDescriptor Initialized. Ready To Process Messages From Queue:inQ_NGramDescriptor_1.3
Service:AnswerScoreDescriptor Initialized. Ready To Process Messages From Queue:inQ_AnswerScoreDescriptor_1.4
Service:hw2-mengday-aae Initialized. Ready To Process Messages From Queue:QAInputQueue
Press 'q'+'Enter' to quiesce and stop the service or 's'+'Enter' to stop it now.
Note: selected option is not echoed on the console.
```

Same as Task 2.2, I created the client descriptor by modifying the value of **`brokerURL`** to **`tcp://localhost:61616/`** and that of **`endpoint`** to the same as I set in the deployment descriptor ***`QAImputQueue`***.

By replacing the Analysis Engine in the CPE descriptor created in Task 1.2 to the client descriptor I just created, I generated the new CPE descriptor which is using a ‘remote’ service as its AE instead of just a local AE.