

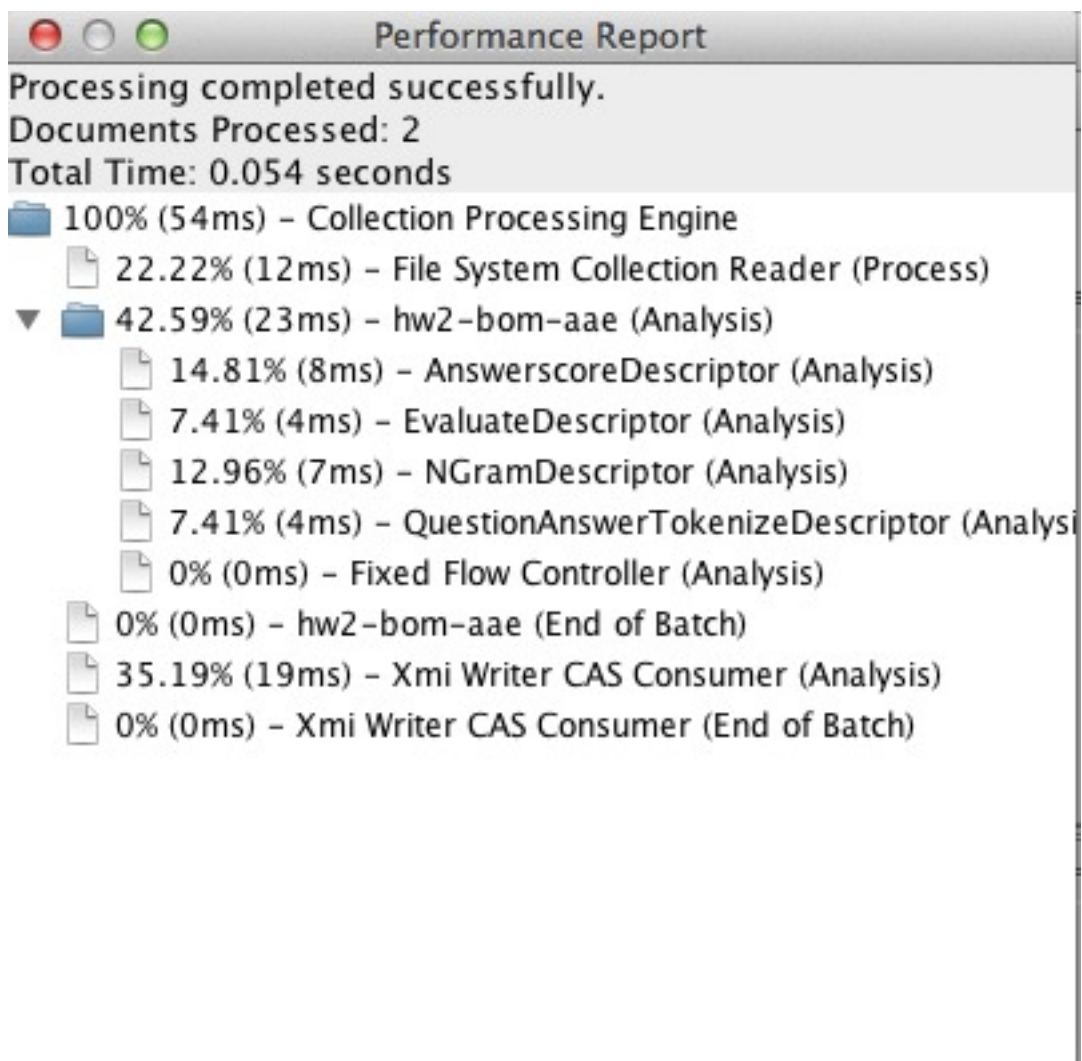
**Design and Engineering of Intelligent Information System*****Execution Architecture with CPE and Deployment Architecture with UIMA-AS*****Name:Bo Ma****Andrew Id: bom****Task 1****Execution Architecture with CPE****Task 1.2 Creating and Running your CPE**

figure 1 the result of the CPE

As you can see in the figure 1 above, I create and run my own CPE, I use the default Collection Reader with File System Collection Reader directly, my hw2-bom-aae as the analysis engine, and create my own Collection Consumer with CAS Consumer. My CAS consumer will output the score

and output all the annotation into the xml file. Then I create my own CPE “hw3-bom-CPE.xml”, it contains my configuration and all the components. I think the CPE is a descriptor file that control the whole pipeline combination and combine the reader, annotator and the consumer.

## Task 2

### Deployment Architecture with UIMA-AS

#### Task 2.2 Creating an UIMA-AS client

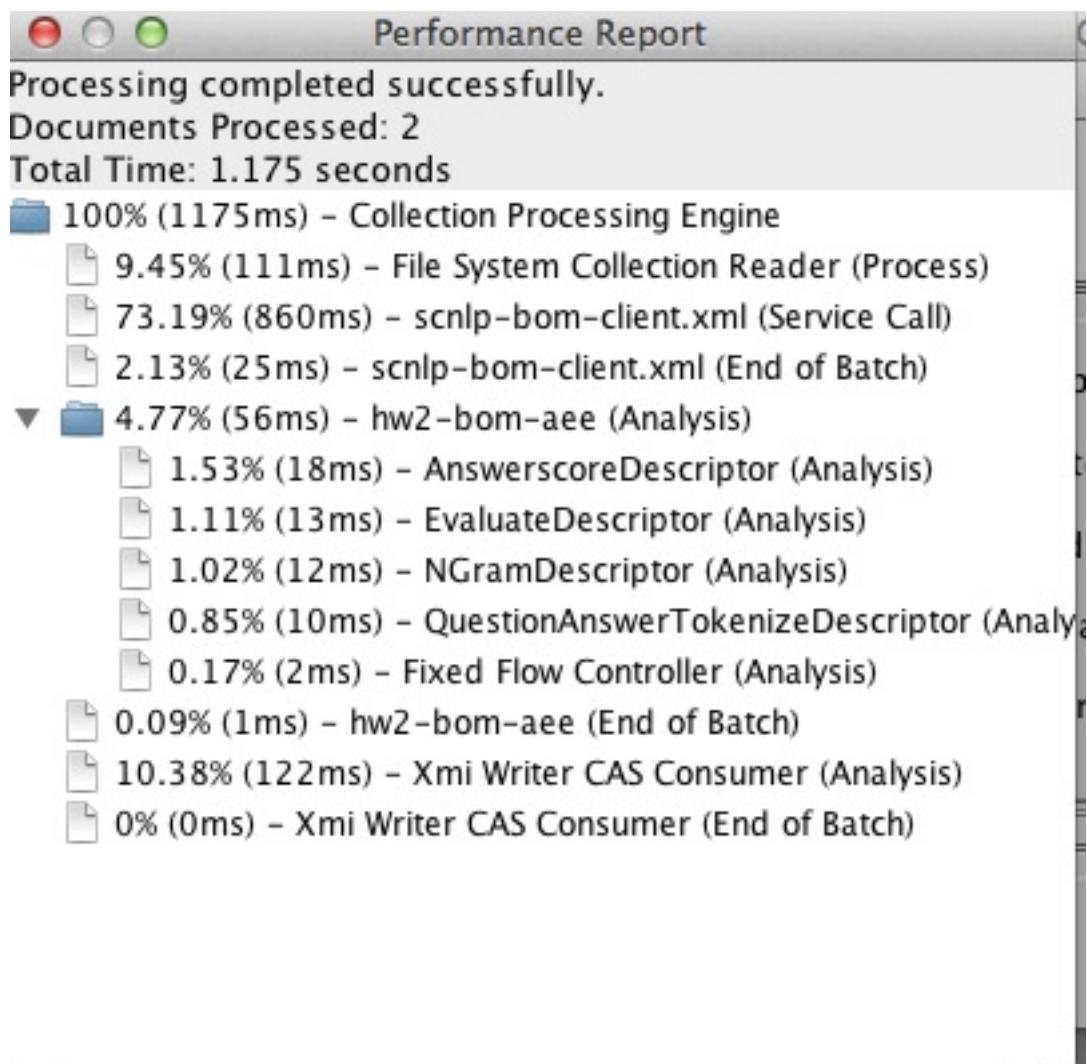


figure 2 the result of adding a UIMA-as client

**The output score of using the Name Entity annotation from Stanford CoreNLP service:**

Question: Booth shot Lincoln?

- + 1.00 Booth shot Lincoln
- 0.63 Lincoln shot Booth
- 0.50 Lincoln assassinated Booth
- + 0.43 Booth assassinated Lincoln
- + 0.36 Lincoln was shot by Booth
- 0.36 Booth was shot by Lincoln
- + 0.29 Lincoln was assassinated by Booth
- 0.29 Booth was assassinated by Lincoln

Precision at 4.0: 0.50

Question: John loves Mary?

- + 1.00 John loves Mary
- + 0.40 John loves Mary with all his heart
- 0.36 Mary doesn't love John
- 0.36 John doesn't love Mary
- + 0.24 Mary is dearly loved by John

Precision at 3.0: 0.67

Average Precision: 0.58

**The output score without using the Name Entity annotation from Stanford CoreNLP service:**

Question: Booth shot Lincoln?

- + 1.00 Booth shot Lincoln
- 0.50 Lincoln shot Booth
- + 0.33 Booth assassinated Lincoln
- 0.33 Lincoln assassinated Booth
- + 0.25 Lincoln was shot by Booth
- 0.25 Booth was shot by Lincoln
- + 0.17 Lincoln was assassinated by Booth
- 0.17 Booth was assassinated by Lincoln

Precision at 4.0: 0.50

Question: John loves Mary?

- + 1.00 John loves Mary
- + 0.33 John loves Mary with all his heart
- 0.22 Mary doesn't love John
- 0.22 John doesn't love Mary
- + 0.13 Mary is dearly loved by John

Precision at 3.0: 0.67

Average Precision: 0.58

In the figure2, we can see , I write my own UIMA as client---”scnlp-bom-client”, In the client I use the Name Entity annotations from the Stanford CoreNLP service, I think it might not very will. In my opinion it will find the name of the person, However, I find some wrong name. Furthermore, I cannot find the name in the question. So I just count the number of the NamedEntity, and judge whether this NamedEntity is in our answer’s begin and end. After than I add the number of the count to Numerator of the Ngram score. Moreover, I count the total number the NamedEntity and add it to denominator of the Ngram score. This make my score higher than the Ngram score.

However, the precision is the same as before and not changed. As you can see the compare result above.

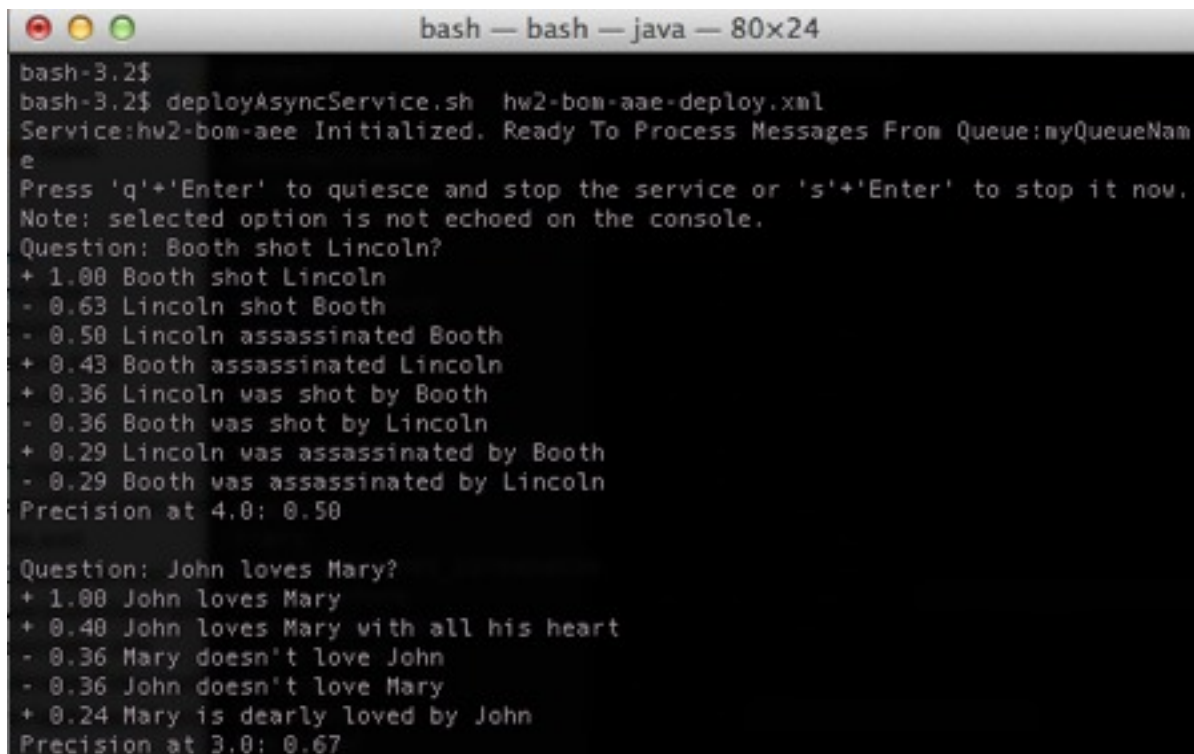
### Task 2.3 Deploying your own UIMA-AS service

I start my broker:

```
.jdk/Contents/Home
Heap sizes: current=83888k free=81617k max=126912k
JVM args: -Dactivemq.classpath=amq/conf:/Users/mabodx/Desktop/school/791/hw/apache-uima-as-2.4.8/apache-activemq-5.4.1/conf; -Dactivemq.home=/Users/mabodx/Desktop/school/791/hw/apache-uima-as-2.4.8/apache-activemq-5.4.1 -Dactivemq.base=amq
ACTIVEMQ_HOME: /Users/mabodx/Desktop/school/791/hw/apache-uima-as-2.4.8/apache-activemq-5.4.1
ACTIVEMQ_BASE: amq
Loading message broker from: xbean:file:amq/conf/activemq-nojournal.xml
INFO BrokerService - Using Persistence Adapter: MemoryPersistenceAdapter
INFO BrokerService - ActiveMQ 5.4.1 JMS Message Broker (localhost) is starting
INFO BrokerService - For help or more information please see: http://activemq.apache.org/
INFO ManagementContext - JMX consoles can connect to service:jmx:rmi:///jndi/rmi://localhost:1099/jmxrmi
INFO TransportServerThreadSupport - Listening for connections at: tcp://cmu-876485.wv.cc.cmu.edu:61616
INFO TransportConnector - Connector openwire Started
INFO BrokerService - ActiveMQ JMS Message Broker (localhost, D:cmu-876485.wv.cc.cmu.edu-58327-1380946131204-0:0) started
```

figure 3 The information after I am successful to start a broker

I create my own deployment descriptor --"hw2-bom-deploy.xml" for my aggregate analysis engine (hw2-bom-aae.xml);



```
bash — bash — java — 80x24
bash-3.2$
bash-3.2$ deployAsyncService.sh hw2-bom-aae-deploy.xml
Service:hw2-bom-aae Initialized. Ready To Process Messages From Queue:myQueueName
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.
Question: Booth shot Lincoln?
+ 1.00 Booth shot Lincoln
- 0.63 Lincoln shot Booth
- 0.50 Lincoln assassinated Booth
+ 0.43 Booth assassinated Lincoln
+ 0.36 Lincoln was shot by Booth
- 0.36 Booth was shot by Lincoln
+ 0.29 Lincoln was assassinated by Booth
- 0.29 Booth was assassinated by Lincoln
Precision at 4.0: 0.50

Question: John loves Mary?
+ 1.00 John loves Mary
+ 0.40 John loves Mary with all his heart
- 0.36 Mary doesn't love John
- 0.36 John doesn't love Mary
+ 0.24 Mary is dearly loved by John
Precision at 3.0: 0.67
```

figure 4 the result that I succeed deploying my "hw2-bom-aae-deploy"

I meet some problem here, when we want to deploy our service, we should be careful that we have add the dependency:copy-dependencies and check the UIMA\_CLASSPATH is right. we should put dependencies in our UIMA\_CLASSPATH. After that we should put out class code which is packaged into a jar also into our UIMA\_CLASSPATH. So after this, we can run deploy our service locally.

The broken Url is set to "tcp://localhost:61616"

endpoint is set to "myQueueName"

The another problem that I meet is the path. I rewrite all the path in the XML file to the Relative path. Because the server do not know the path directed by "name". so I build the path directed by location and use the relative path.

After that, I will run my client to connect to the local server.

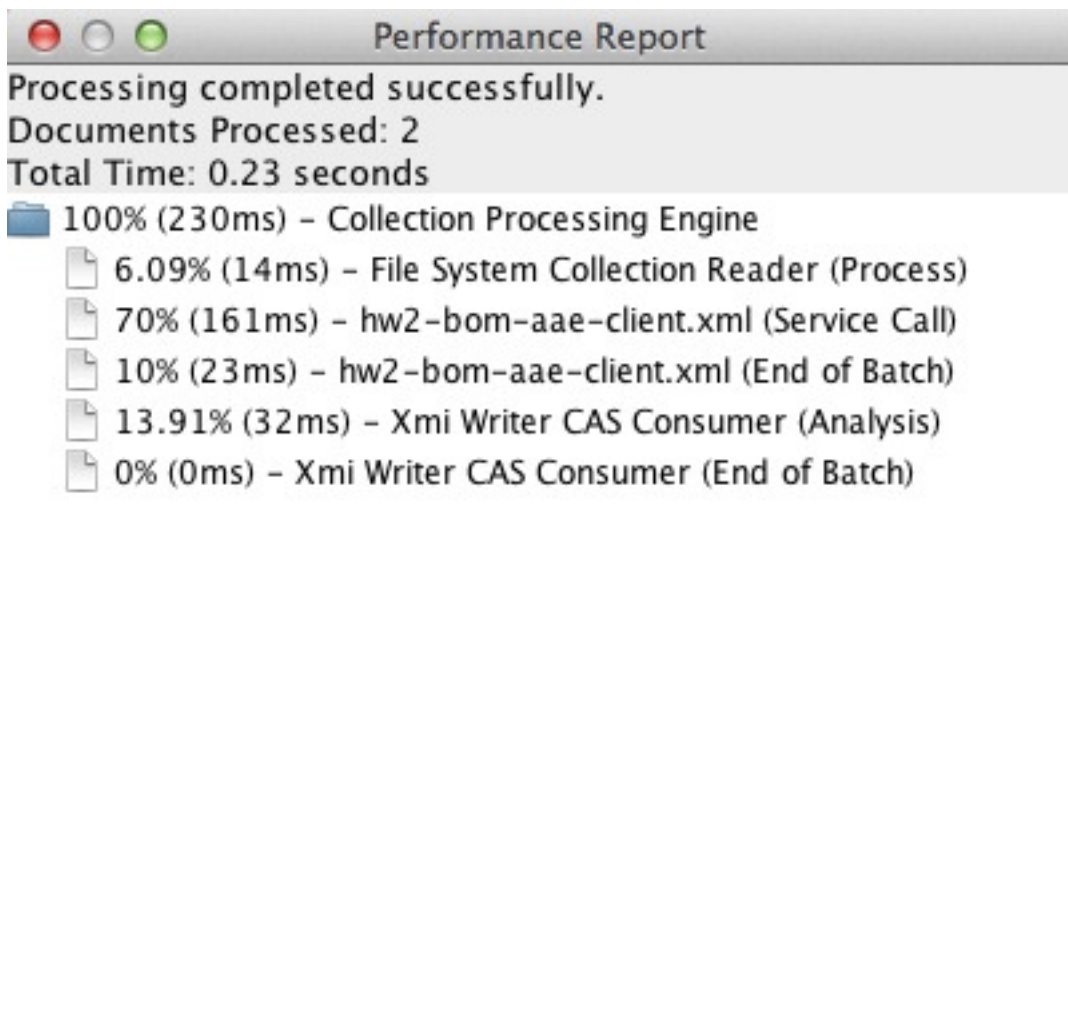


figure 5 the result that I run my “hw2-bom-aae-client.xml” to connect the local server

In the above picture, you can see I have also created a CPE descriptor (hw3-bom-aae-as-CPE.xml) to test my local service by calling the client.

## Task 2.4 Bonus

**1、 run a Stanford CoreNLP annotator locally, and compare the speed with the re- mote one;**

I build a StanfordCoreNLP Descriptor.xml to run the Stanford CoreNLP annotator locally, and compare the speed of it with the speed of run the Stanford CoreNLP remotely.



As you can see in the following picture, the speed for the locally Stanford CoreNLP annotator is much slower, it cost 3.268 to run locally, but just 1.175 to run remotely.

So the UIMA AS has an effective scale out and compute in remote sever which is dispatched is faster.

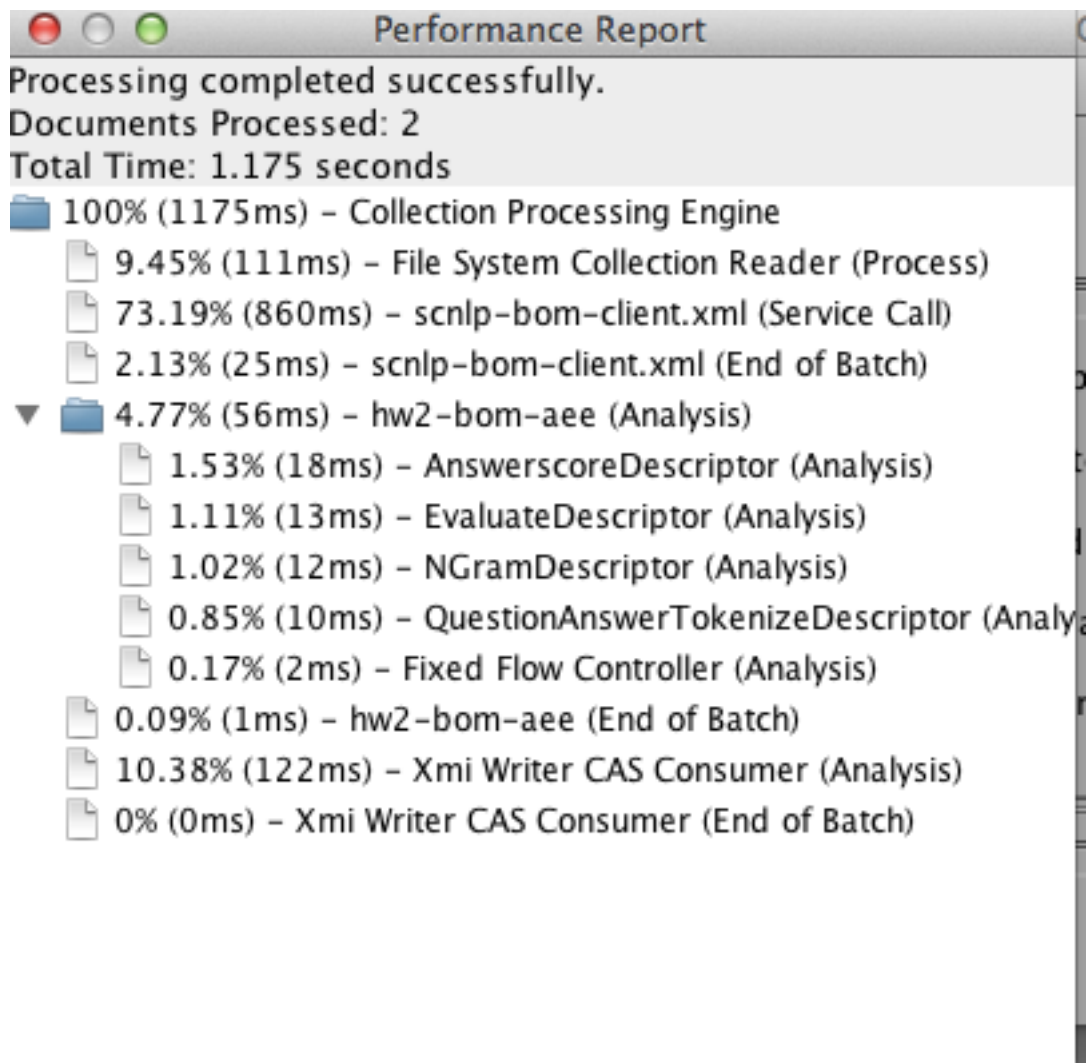


figure 6 run a Stanford CoreNLP annotator remotely,

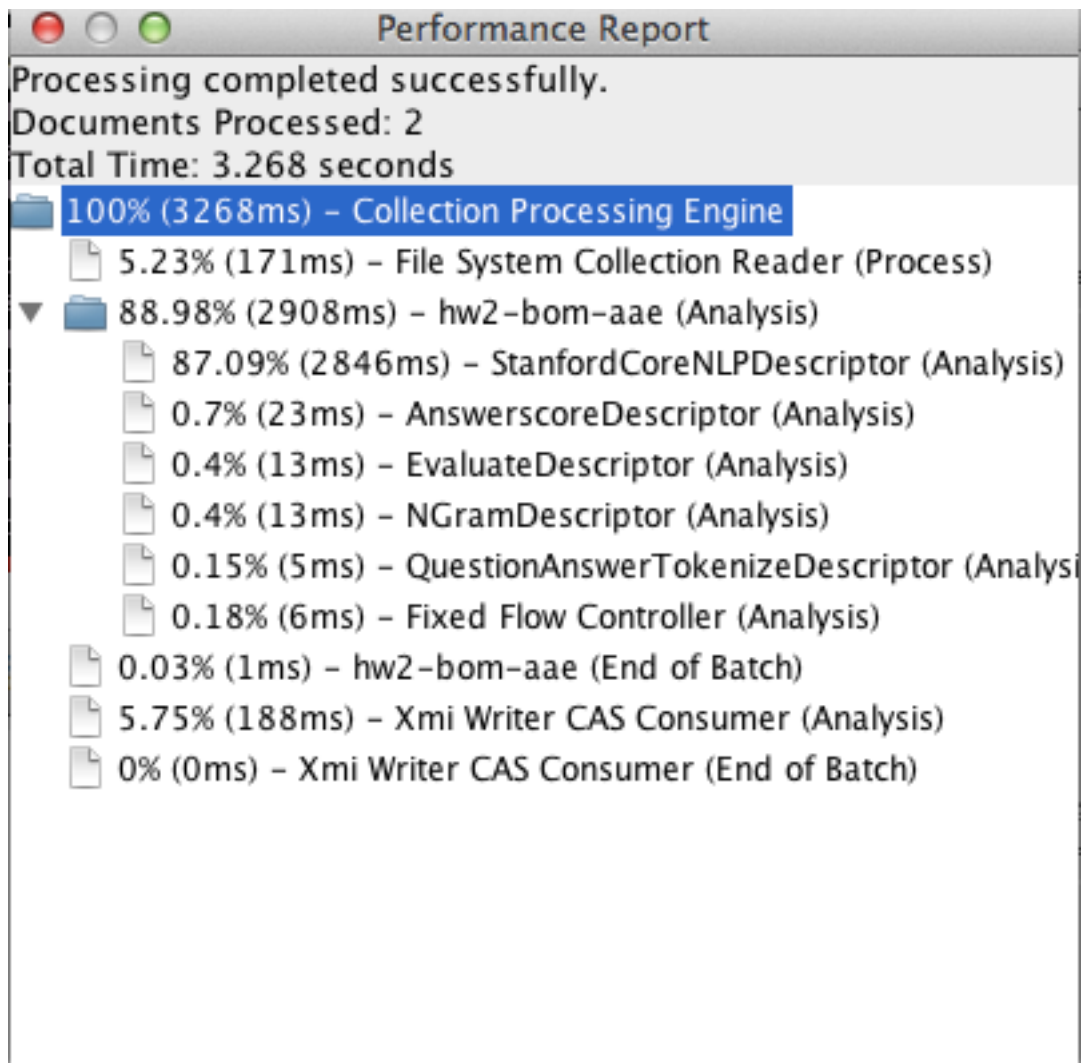


figure 7. run a Stanford CoreNLP annotator locally,

## 2、incorporate other annotations from Stanford CoreNLP such as POS-tagging, lemma, and parsing;

I rewrite my AnswerscoreAnnotator . I incorporate the POS-tagging and Lemma annotation in my software. I also add the name entity. This really helps me improve my precision.

POS-tagging : I use the pos-tag to help me find the Noun the Verb

A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads text in some language and assigns parts of speech to each word (and other token), such as noun, verb.

So I use this Tag find the Noun which is better than the namedentity.



Lemma: I use Lemma to generate the word lemmas for all tokens in the corpus.

This is very helpful for me to judge whether the sentence is passive or positive. Then I can combine the score of the POS-tagging and lemma annotation with my NGram.

For each pair of question and answer, I first find whether they have the same number of NamedEntity. Then I find their verb, I judge whether this verb is positive or passive, if passive judge the order of the NamedEntity. If the logic of the sentence is the same I will give them 1 score, if not, I will decrease 1 score from them.

Then I also find the negation sentence from the “not” phase. And also consider the logic of the sentence. If both of question and answer are the affirmative or negative I will add 1 score to the answer. If they have different affirmative or negative, I add decrease 1 score.

Through this way, I get a very good accuracy which is 0.75. You can see the result in the following.

The result that I get after incorporating the POS-tagging and Lemma.

Question: Booth shot Lincoln?

- + 1.11 Lincoln was shot by Booth
- 1.11 Booth was shot by Lincoln
- + 1.09 Lincoln was assassinated by Booth
- 1.09 Booth was assassinated by Lincoln
- + 0.18 Booth shot Lincoln
- 0.13 Lincoln shot Booth
- 0.11 Lincoln assassinated Booth
- + 0.08 Booth assassinated Lincoln

Precision at 4.0: 0.50

Question: John loves Mary?

- + 1.10 Mary is dearly loved by John
- + 0.26 John loves Mary
- + 0.19 John loves Mary with all his heart

- 0.12 Mary doesn't love John  
- 0.12 John doesn't love Mary  
Precision at 3.0: 1.00

Average Precision: 0.75

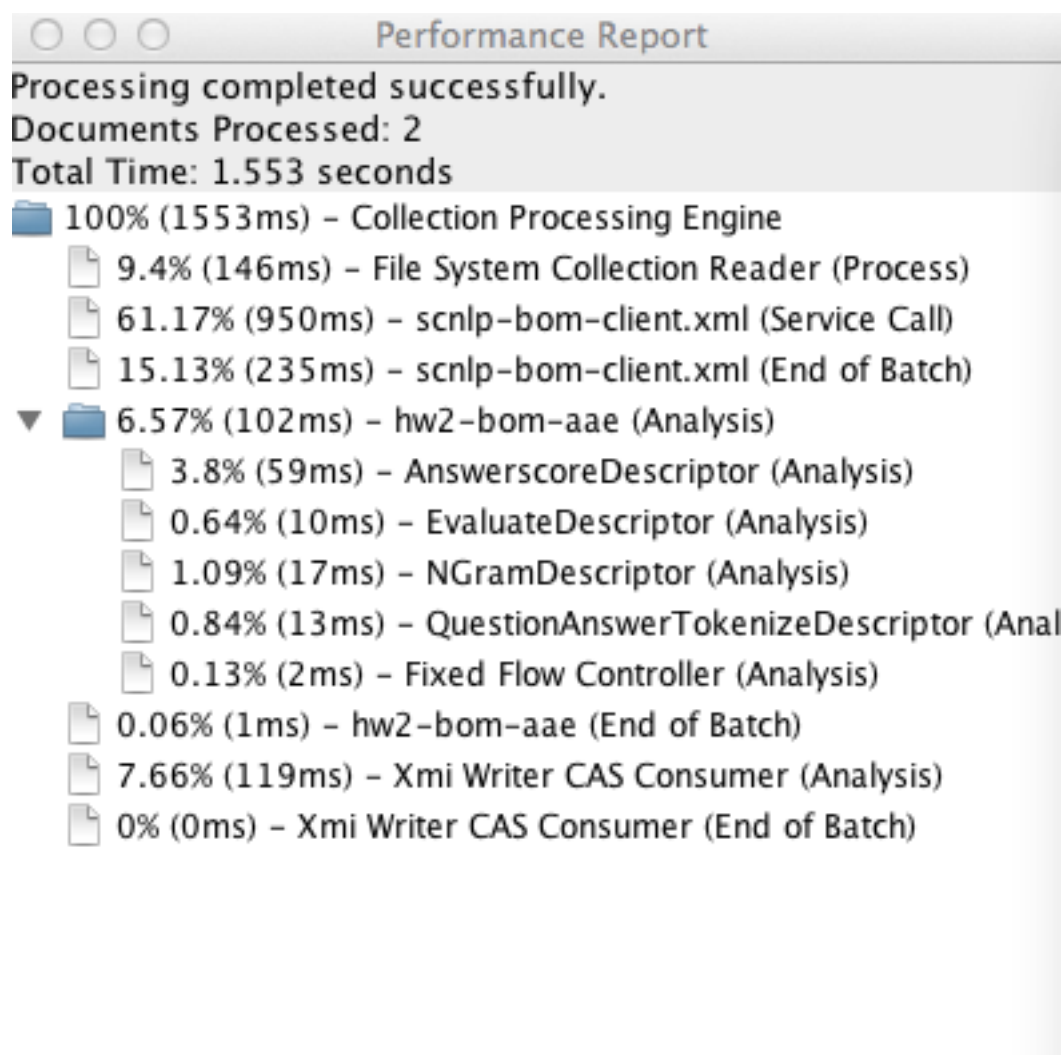


figure 7. Performance of running my incorporating the POS-Tag and Lemma annotator into my software

The time it cost to run the code with incorporating the POS-Tag and Lemma annotator is 1.553 seconds.

The time it cost to run my original code is 1.175 seconds.

As you can see in the above picture, the time of running incorporating the POS-Tag and Lemma annotator into my software is much longer than the just running my original software.

## **Summary**

After this homework, I really understand the UIMA AS structure, and how the UIMA CPE work and how to analyze large volumes of unstructured information in order to discover knowledge that is relevant to an end user. I get the idea of how to use some tool such as the Stanford CoreNLP to solve the problems.