Ryan Carlson (rcarlson)
Software Engineering – Homework 3
October 7, 2013

I'll take each task in turn, discussing how my approach.

Task 1: CPE

The CPE was fairly straightforward. I copied over the FileSystemCollectionReader and XmiWriterCasConsumer verbatim from the UIMA examples. For the cas consumer, I modified it to keep a running total of the precision and the number of elements that have been evaluated, and then I implemented the collectionProcessComplete to report the average P@N score to standard out. Then I just created a CPE that involved all those components with hw2-rcarlson-aae as the analysis engine.

Task 2: UIMA-AS

Here I created a custom resource descriptor to contact the server and get the Stanford Core NLP annotations. Then I needed a way to integrate those results into my scoring system. So I created a new descriptor called StanfordNameEntityScoreAnnotator. This descriptor takes question / answer pairs and assigns a score based on the overlap of named entities contained in each.

At this point, we also need a way to combine scores from multiple sources (Stanford and CosineSimilarity, in my case), so I created a CollapseAnswerScoreAnnotator that looks at the AnswerScore instances that cover the same span and sums their scores together into a new AnswerScore instance. This then gets passed to the evaluator. The evaluator only looks at these collapsed AnswerScore instances

The accuracy remained identical to the results from the second assignment. In both cases, the hw2-rcarlson-aae step took under a half second (between 300 and 400 milliseconds). The scnlp-rcarlson-client step took just over one second (about 1250 milliseconds). The entire CPE takes 2 seconds, which is about one second slower than without the Stanford annotations.

Note that I modified hw3-rcarlson-CPE to add the call to the Stanford service.

The last step was fairly straightforward once CLASSPATH issues were sorted out. The deployment descriptor for hw2-rcarlson-aae was created, and then I made a client (hw2-rcarlson-client) to point to the service with the queue name of “rcarlson-hw2-aae-client-queue.” Then I just created a CPE that uses the service, started the broker, deployed the service, and ran the CPE.

Note that I couldn’t figure out how to use relative paths, so I only have absolute paths in my CPEs. This might cause a problem for automatic testing and whatnot. Also, if it’s interesting, my CLASSPATH was set to

/Users/rcarlson/local/eclipse-workspaces/software-engineering/cmu11791-hw3-rcarlson/**hw3-rcarlson/target/dependency/**:/Users/rcarlson/local/eclipse-workspaces/software-engineering/cmu11791-hw3-rcarlson/**hw3-rcarlson/target/classes/**

I highlighted the sort of “relative” parts in bold.