Summarize the design

1. Trained data

The data model I used comes from the website of **Lingpipe** where there are so many useful trained data. And the data model for genes based is called ne-en-bio-genetag.

2. Pipeline

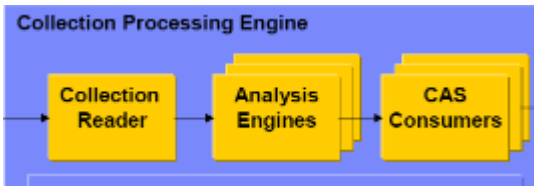

Diagram about pipeline

a) The first thing I should design is the **collection reader**, where we read data in the structure CAS. I use FileSystemCollectionReader as my collector in this program.
b) Second, I should put the **CAS consumer** as the input to conclude the result. There are several different consumers and I choose the Annotator_Printer as the consumer.
c) Finally, I should develop the key part of this pipeline, Analysis Engines and what should I do in it is just implement the process method which is public void process(JCas aJCas).

3. Some details about implement

Step1: Create the "TypeSystem.xml"
Using the UIMA descriptor editor, I can generate a type system xml file. In this homework, I create a new type: Sentence and subtype SentenceId, and genestext. They are all types of String. Then generate new 2 java class by pressing JCasGen.

Step2: Create the Analysis Engine like annotator
In this homework, I create a new annotator named . By importing the trained model in "lingpipe" as I suggest above, I can use this trained model to deal with data in "simple.in".

Step3: Import all the types into "annotator.xml"
Open the "SentenceAnnotator.xml" and select the import type system sub panel, I can import the defined "Desciptor.xml" as I mentioned above.

Step4: Implement the process method
In the method, I first declare the instance of annotator and put what are fit in the model into annotator indexes. During this process, I divide whole paragraph in

“simple.in” into sentence and analysis each sentence.

Step5: Create Collection reader

I use existed Collection reader FileCollectionReader.xml and its java.class in UIMA Frame Work.

Step6: Create Cas Consumer

I use existed Cas Consumer Annotator Printer.xml and its java.class in UIMA framework and make some little modification.

Step7: Generate the “CpeDescriptor.xml” using the UIMA tool “UIMA CPE GUI”

This has already mentioned in the UIMA tutorial handout book.

Step8: Import the Maven Dependency lingpipe4.0.0

By Changing the setting.xml in Maven, I can find the lingpipe-4.0.1.jar easily.