Final Report

Name: Ningna Wang
AndrewID: ningnaw

Idea:

This project creates five descriptors: typeSystemDescriptor, collectionReaderDescriptor, aeDescriptor, casConsumerDescriptor, CPEDescriptor.

1. **typesystemdescriptor**: I define two types (classes) which are Type.gene and Type.sentence. There are two features in Type.sentence: ID and Text. And four features in Type.gene: ID, Text, GeneStart, GeneEnd.
2. **collectionReaderDescriptor**: I use *CollectionReader.java* to run this component and define one configuration parameter—InputDocument--to input the unprocessed document. Also, this descriptor output all features of Type.sentence.
3. **aeDescriptor**: I test two annotator to run this component: *StanfordCoreNLP.java* or *LingPipeAnnotator.java*. And I found that *LingPipeAnnotator.java* can perform better than *StanfordCoreNLP.java.* This descriptor input all features of Type.sentence and output all features of Type.gene.
4. **casConsumerDescriptor**: I use casConsumer.*java* to run this component. This descriptor define Type.gene as input.
5. **CPEDescriptor**: show out when Run as UIMA CPE GUI.

Comparation:

The idea using LingPipe performs much more better than using StanfordCoreNLP.

1. Comparing the results of StanfordCoreNLP with sample.out. The precision is 0.39402310654685496.
2. Comparing the results of StanfordCoreNLP with sample.out. The precision is 0.8959551898483196

The figure 1 is the flowchart of my project.

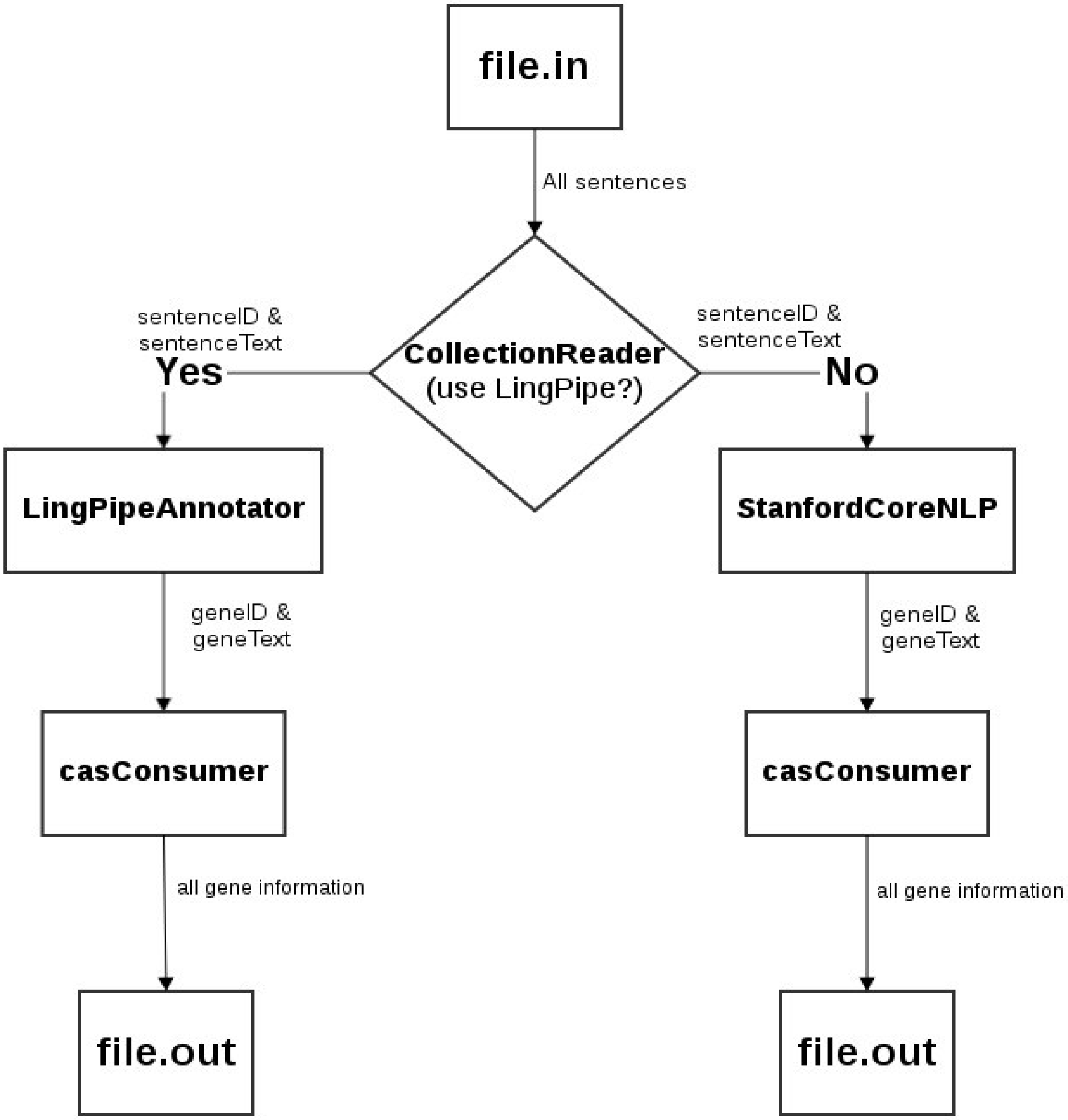

Figure 1: Flowchart of my project

The figure 2 is the UML diagram of my project using *StanfordCoreNLP.java.*

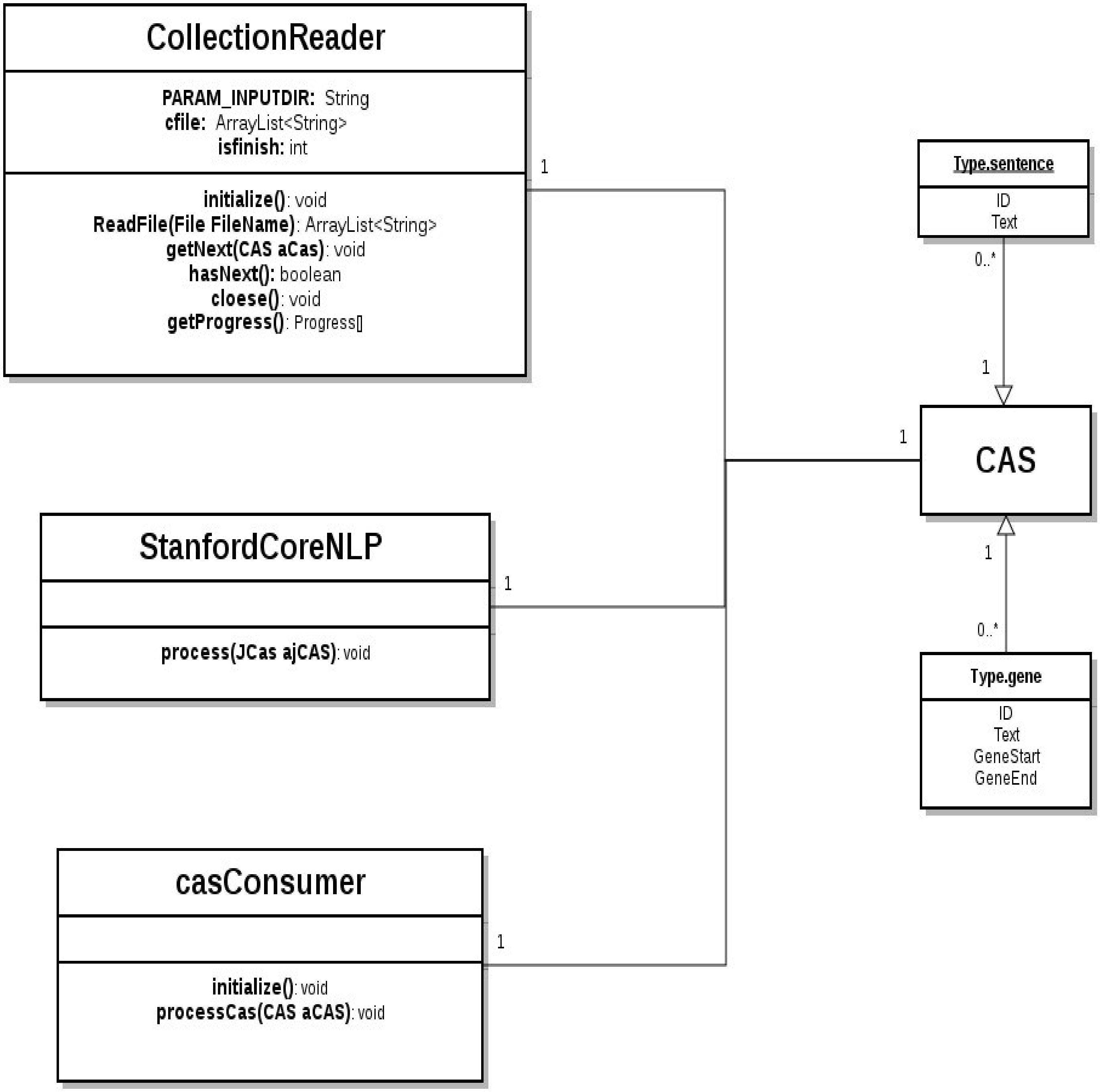

Figure 2: UML diagram using StanfordCoreNLP

The figure 3 is the UML diagram of my project using *LingPipeAnnotator.java.*

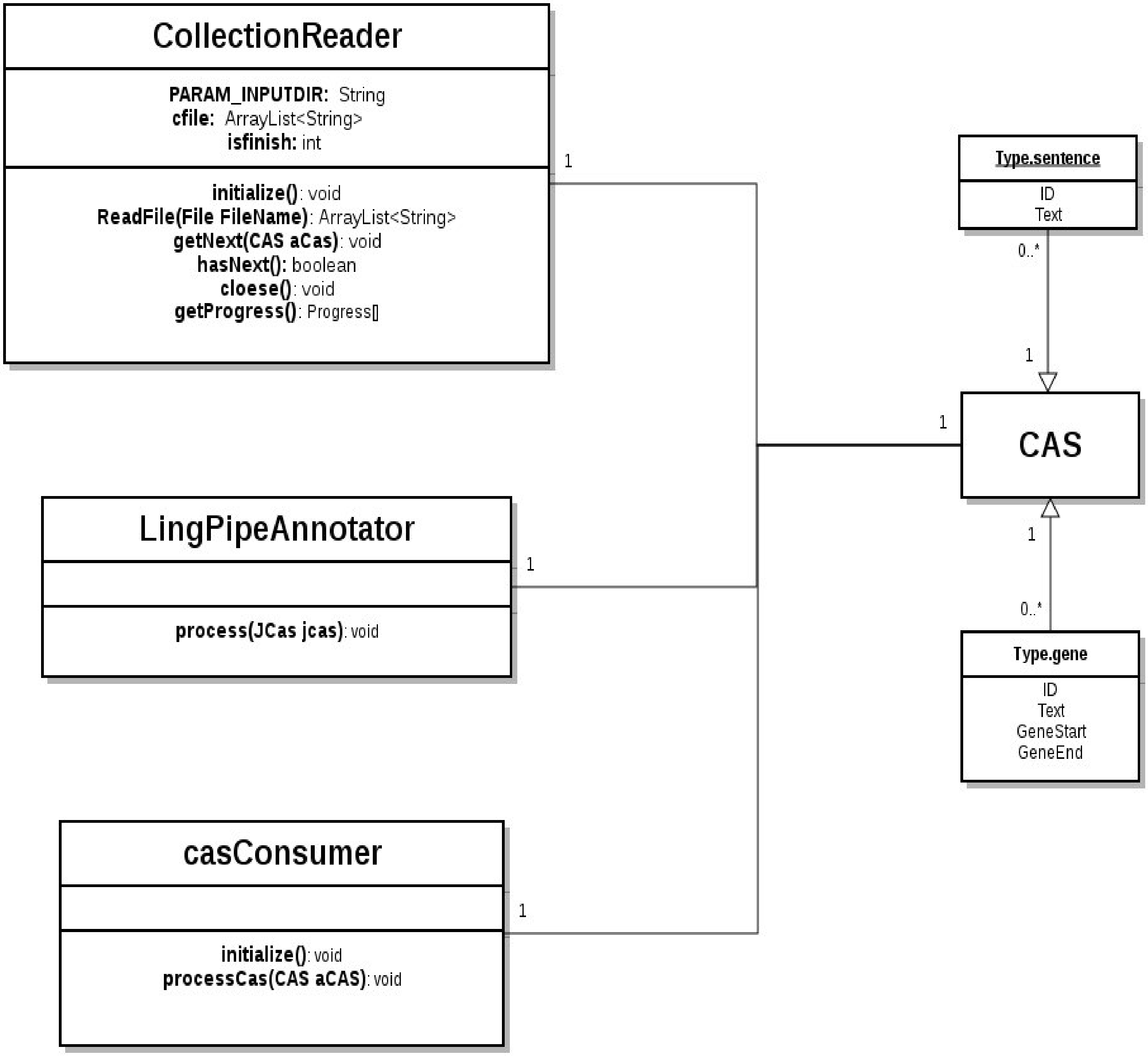

Figure 3: UML diagram using LingPipeAnnotator