GeneTAG Named Entity Recognizer Using UIMA SDK

Shourabh Rawat (srawat@andrew.cmu.edu)

1. Design

We use the UIMA architecture to design the various components of our GENE Tag Named Entity Recognizer. Our system has the following 4 main components based on the UIMA architecture.

- Type System: [GeneTag]
  We are using a very simple type system in our implementation. We maintain a single annotation called GeneTAG of super type "tcas.uima.Annotation". We add an additional feature called the "SentenceID" to hold the sentence information where the annotation was found.

- Input Collection Reader [FileReader]
  The task of the input collection reader is to read in the input file and generate a collection of CAS objects(one for each sentences) for further processing.

- CAS Annotator [GeneAnnotator]
  This is the module which performs the task of annotating sentence text with GeneTAG annotations. For the annotation part we use a standard pre-trained HMM model from Ling Pipe toolkit and invoke it for each sentence. This involves extracting the raw sentences from the CAS object, running the models and then storing back the newly found annotations back into the CAS object.

- CAS Consumer [GeneConsumer]
  The task of our CAS consumer is to read in the annotations stores in the CAS objects and list them out in an output file in the following format.
  <SentenceID>|<start_span> <end_span>|<GeneTAG>

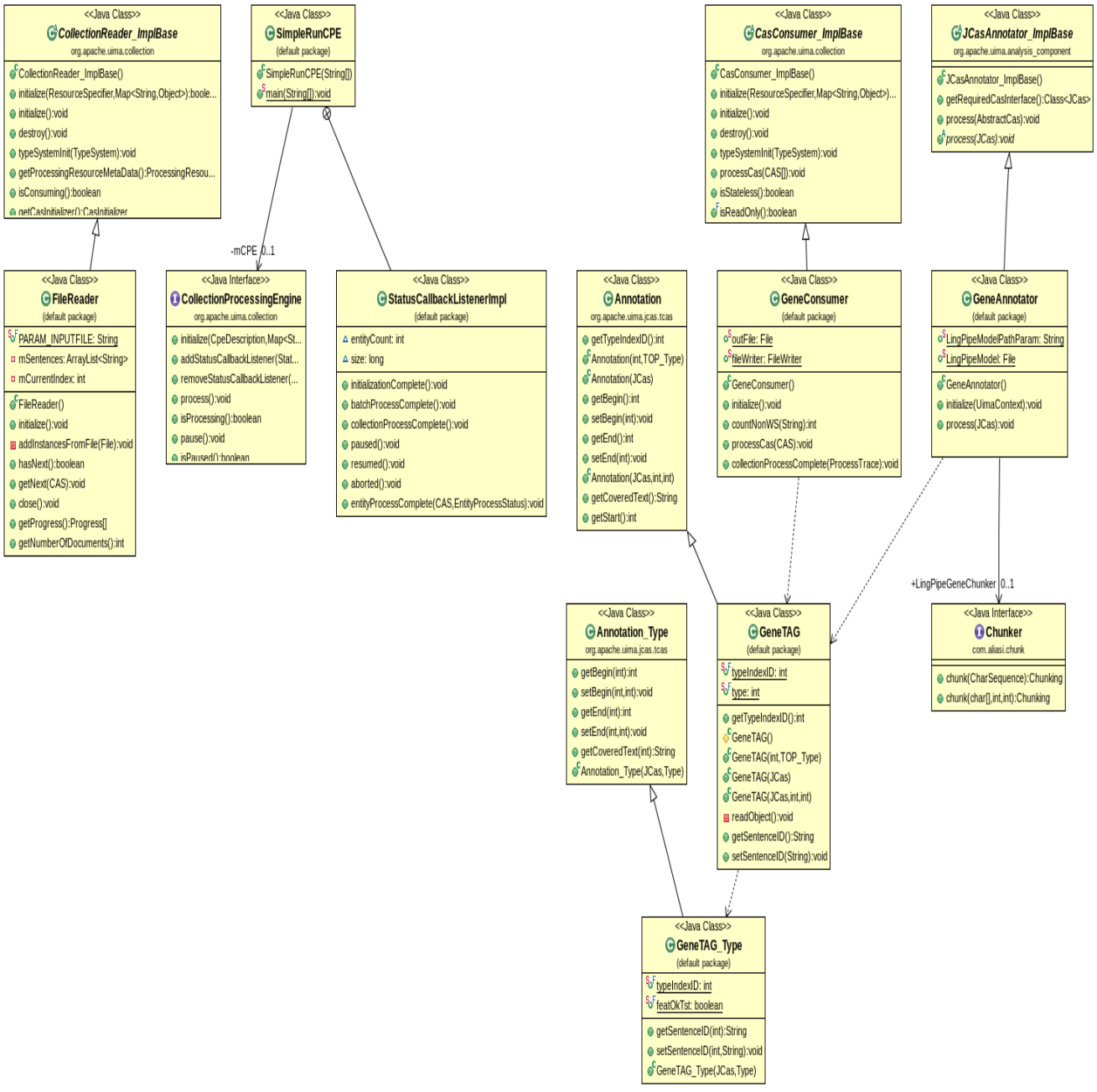
- CPE Engine
  We use the provided SimpleRunCPE.java for running our pipeline. We provide it the CpeDescriptor.xml as input which contains the information would the components of our pipeline including the FileReader (Collection Reader), GeneAnnotator(CAS Annotator or Analysis Engine), and GeneConsumer (CAS Consumer).

- Class Structure
  Type System: GeneTAG.java GeneTAG_Type.java
  Input Collection Reader: FileReader.java
  CAS Analysis Engine: GeneAnnotator.java

CAS Consumer Engine: GeneConsumer.java
CPE: SimpleRunCPE.java

- Architecture:

UML Class Diagram:



To get a better version of the class diagram one can look in the docs folder for the file docs/GeneTAGClassDiagram.png.

2. Algorithm Details:

- Machine Learning Techniques Used: We use a pre-trained Model from Ling Pipe for our Analysis Engine. The model uses Hidden Markov Models for training a chunker over the GeneTag dataset. Though the Ling Pipe toolkit provides several different models for the GeneaTag annotation, we use the "First-Best Named Entity Chunking" model which only the most confident prediction. The implementation could be found here in the following package: *com.aliasi.chunk.HmmChunker* in the Ling Pipe source code.Also one can download the pre-trained model from here(ne-en-bio-genetag.HmmChunker). The model has been trained on the GeneTAG Dataset.
- Machine Learing Components: Ling Pipe HMM Gene Chunker
- External Training DataSet Used: GeneTAG Dataset
- External Lexical Resources: None
- Rule Sets Used: None
- Biological Databases Used: None

3. Evaluation:

We report our performance based on Precision, Recall and FMeasure metric. This is done by comparing the output of our chunker "hw1-srawat.out"[A] with the "sample.out"[B] on the input data "sample.in" provided as part of the assignment.

$$\text{Precision} = \frac{(\#Items\ in\ A - \#Items\ in\ A\ Not\ in\ B)}{\#Items\ in\ A}$$

$$\text{Recall} = \frac{(\#Items\ in\ B - \#Items\ in\ B\ Not\ in\ A)}{\#Items\ in\ B}$$

$$\text{F-Measure} = \frac{2 * Precision * Recall}{Precision + Recall}$$

Precision measures the percentage of False Alarms while recall measures the percentage of Missed Detections.

Our Evaluation gave the following results:
Precision: 15504 of 20174 Correct (**76.85%)**
Recall: 15502 of 18265 Correct (**84.87%)**
F-Measure: **80.66%**

4. Performance:

Here are the timing results from the SimpleRunCPE.java on the "sample.in" dataset. We observe that annotation engine [GeneAnnotaor] or Analysis Engine is responsible for bulk of the processing time, supposedly so because it contains the HMM model that identifier GENE entities in the text.

Total Time Elapsed: 12918 ms
Initialization Time: 1167 ms
Processing Time: 11751 ms

Component Name: FileReaderDescriptor
Event Type: Process
Duration: 865ms (10.48%)
Result: success
Component Name: Gene Annotator
Event Type: Analysis
Duration: 6612ms (80.12%)
Component Name: Gene Annotator
Event Type: End of Batch
Duration: 64ms (0.78%)
Component Name: Gene Consumer
Event Type: Analysis
Duration: 630ms (7.63%)
Component Name: Gene Consumer
Event Type: End of Batch
Duration: 82ms (0.99%)