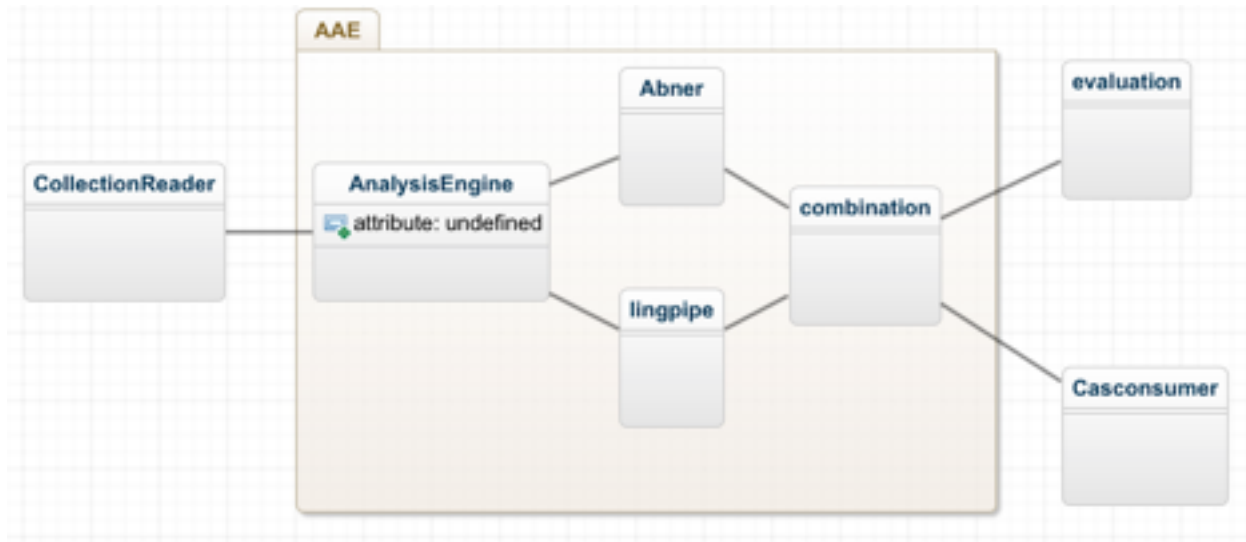


# HW2-Report

Name: Jingwei Li Andrew id: jingwei2

## 1. System Architecture



I have three main part in my CPE system including the **CollectionReader** which read files, the **AAE** that aggregate 4 annotators and consumer to evaluate and output the results.

**CollectionReader:** The Collection Reader read the hw2.in file line by line and put the content in JCAS that AAE could use the data for analysis.

**AAE:** The AAE is a aggregation of my annotators. I use two main annotators in the procedure including the Abner and the lingpipe which the tool of Abner and Lingpipe respectively.

Firstly, the data collected by the **CollectionReader** is read in a **AnalysisEngine** which split the key and the content of the data. Then save the key and content separately in annotation.

Then, two parallel annotators are used to analysis the data by pick the name of the genes and their starting and ending position in the sentence. The results of the two annotators are wrote into two type systems separately.

The combination is used to combine the data of lingpipe and Abner by finding the overlap of their data and pick one of them in the output data.

**Consumer:** The consumer is composed of two part: the evaluation part and the casconsumer part. The evaluation part is a consumer that evaluate the output data with the sample out data to get the precision, recall and F score of the output data. The casconsumer part writes the start and ending position of gene name in the content and the gene name according to the form of the instruction.

## **2.Type System**

**My type system design is as follows:**

**uima.types.Abnerdata**

**gene**

**uima.types.InputData**

**content**

**key**

**uima.types.Lingpipedata**

**gene**

**uima.types.OutputGene**

**contnet**

## **3. Design Pattern**

Trough testing the performance of Abner, it turns out to be as follows:

precision: 0.4670550263717232

recall: 0.34292815444193975

F score: 0.3954806025863218

Trough testing the performance of Lingpipe, it turns out to be as follows:

precision: 0.8062231870574249

recall: 0.9015663834460436

F score: 0.8512333551626282

According to the instruction, I aggregate the two tools together and combine the results of the two annotators to pursue a higher performance. During the combination, the two results are added together and the overlap contents are found. Then I added the contents with those contents that were not overlapped together to aggregate the final result.

#### **4. Performance Evaluate**

precision : 0.6804915514592934

recall: 0.8705855152881337

F score: 0.7638899454799037