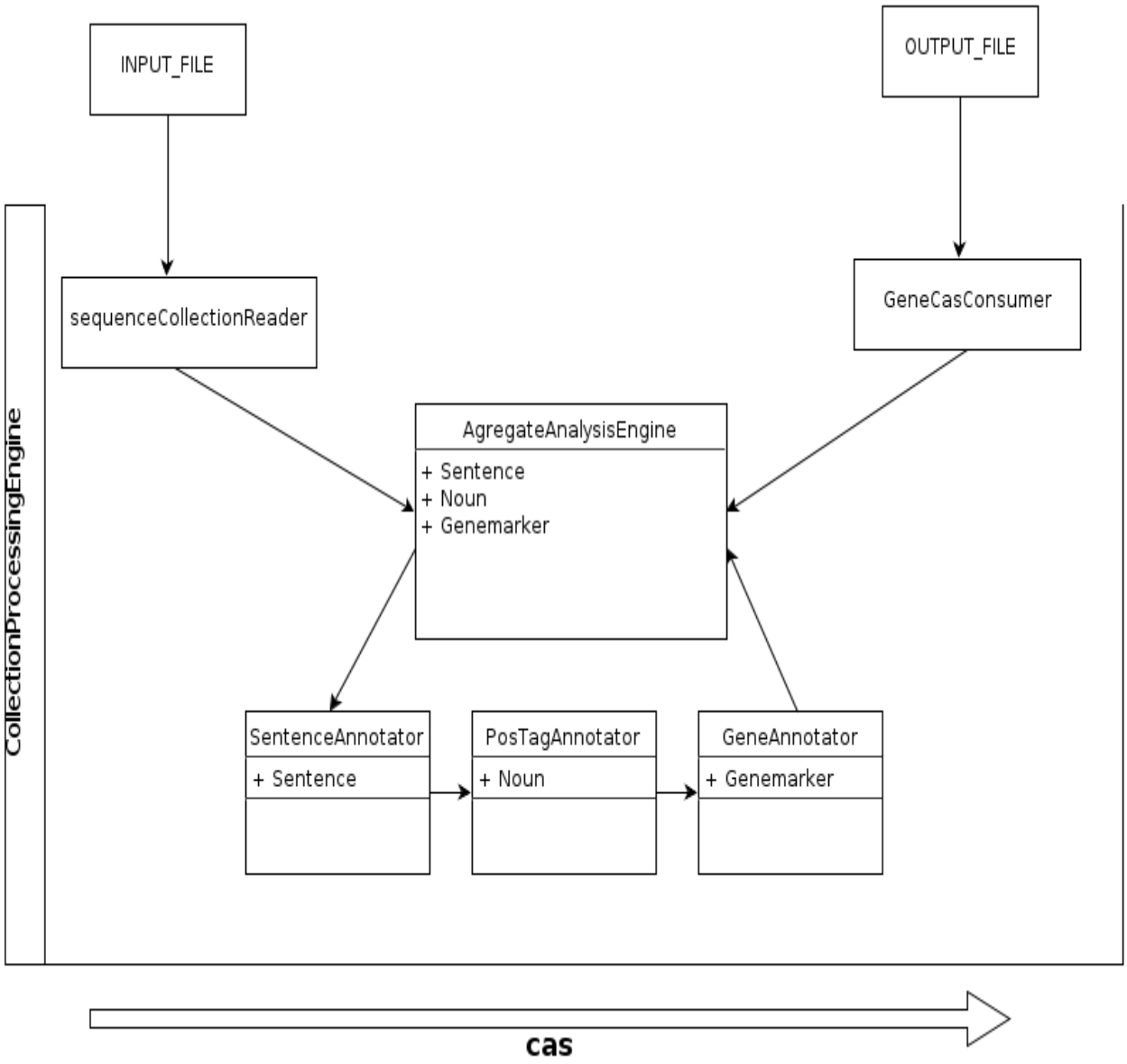
System Design

Varuni Gang
id: vgang

This diagram shows the way the pipeline designed for the task three of the assignment consisting of
a) sequenceCollectionReader
b) AgregateAnalysisEngine
c) GeneCasConsumer



a) sequenceCollectionReader :

This collection reader takes the file “sample.in” as input and reads it line by line and uses getnext method to generate cas. These cas are further passed to the annotators for annotations.

b) AgregateAnalysisEngine :

It includes three analysis engine :

1) SentenceAnnotator :
This annotator is used to annotate sentence and its id. It accomplishes such task through Sentence type system.

2) PosTagAnnotator :
This annotator is used to annotate nouns from sentences and its id. It accomplishes such task through Noun type system. It uses Stanford CoreNLP tool for recognising nouns from the sentence.

3) GeneAnnotator :
This annotator is used to annotate genes from the nouns previously identified and its id. It accomplishes such task through geneMarker type system. The annotator searches database downloaded from NCBI genebank i.e dictionary lookup is performed.

c) GeneCasConsumer :

This cas consumer collects all the annotations and outputs them to a file.

While development of the correct pipeline for this task , I prefered this pipeline due to less time complexity. Some of the other techniques that I tried to apply for the task are listed below, these include the reason behind there unsuccessful as being a good pipeline for the assignment :

a) Online database search :

- Searching the correct gene name from the external database takes a long time, It takes approximately 20 min to search for the limited number of sentence tags provided in the assignment.
- Due to imperfection in the database search there were many erroneous gene tagging

b) Gene Named Entity Recognizer :

- Implementation of the dictionary search on the data set made from the biomedical literature, gave almost similar results as the gold-output provided in the assignment but the major drawback of it not being submitted is the large size of the dataset generated.

- The size of the data set exceeds more than 110 mb and thus could not be submitted for the task.

Other details asked in the question set :

- Due to unrelated background with respect to natural language processing, I wasn't able to incorporate any of the techniques in my pipeline.
- Some of the machine learning techniques that could be implemented for such tasks are Hidden Markov Model (HMM) and an HMM-based chunk tagger through the LingPipe API.
- Rule sets based on GENETAG Annotation Guidelines could be applied to increase the accuracy of the annotation.