1. System design
   1) Pipeline

      My pipeline consists of 1 collection reader, 3 annotators, and 1 Cas consumer.

      - In collection reader, program reads in sentences from document and generates a Cas for each sentence.

      - The first annotator is *simpleSentenceNumAndTextDetection*. The annotator separates *sentence ID* and *sentence text* of the input Cas.

      - The second annotator is *geneNameDetector*. In this annotator, I use a HMM models, which is trained by lingpipe, to annotate gene names in sentence text.

      - The third annotator is *geneNameFilter*. This annotator performs an online checking task. If confidence of a gene name is within a certain range ([0.3,0.6]), the program checks whether this gene is in bergman lab database. If in, update the confidence to 1 otherwise update the confidence to 0.

      - The Cas consumer output geneName with confidence larger than threshold (0.6).

   2) Type system

      There are 2 types in my type system. *RawSentence* and *GeneName*

      - RawSentence has 2 features, sentence ID and sentence text. The raw sentence type is the output of the first annotator and input of the second annotator.
      - GeneName is has 3 features, sentence ID, gene text, and confidence. Second annotator output GeneName and third annotator input GeneName and output an updated GeneName.

2. Algorithm
   1) Machine learning tools

      Use lingpipes *ne-en-bio-genetag.HmmChunker* model to detect the gene named entity.

   2) Knowledge sources

      Use bergman lab database (http://bergmanlab.smith.man.ac.uk/) to check if a geneName is really a gene name.

3. Evaluation result
   1) F-measure

      Recall:0.780016

      Precision:0.808695

      F-Measures: 0.794356

      Currently, I check those GeneName with confidence in [0.3, 0.6]. If they are in the gene name database, reset the confidence to 1 otherwise to 0. Then output all gene name with confidence greater than 0.6.