11791 HW1 Report

Anurag Kumar
Andrew id: alnu@andrew.cmu.edu

September 24, 2014

1 NER Report

I implemented the system through two types in type systems. The first one is "Annotated" with "Sentence" and "Sentence Id" as attributes. The second one is "stringN" which stores the output features. Features/Attributes in "stringN" are Begin, End, and Sentence. Sentence attribute contains the gene output. Begin and End represents start-offset and end-offset. For final output the sentence id is accessed through the "Annotated" type. The collection reader (NewCollectionReader.java) reads input file and splits sentence and sentence id. This is done line by line. The CAS annotator (NewJcasAnnotator) uses lingpipe to identify genes in the input sentences. All chunks of gene names are returned by lingpipe chunker. It's based on Hidden Markov Models. Finally CAS consumer (CasConsumer.java) writes the output to the output file. Some of the important aspects of the implementation are

- Line by Line reading input file
- The sentence id and text are separated during the collection reader process.
- Lingpipe based pre-trained model is used for gene type recognition
- The type system description is already described above

Answers to the specific questions.

- Machine Learning Techniques used - I used Hidden Markov Models(HMM) based named entity recognizer. This was done through lingpipe.

- I used lingpipe which is based on HMM. So all NLP techniques are done through lingpipe. The exact model from linpipe used is
- No external training data were used.
- Apart from lingpipe genetype tag no other resources were used.
- No interaction with external biological database
- We used lingpipe based model for gene recognizer.
- A brief description of overall flow of system has already been given above. The system was tested on the provided sample.in file. The performance is better than the provided sample.out.