Named Entity Recognition System using UIMA

Pradeep Prabakar Ravindran (ppravind)

October 17, 2012

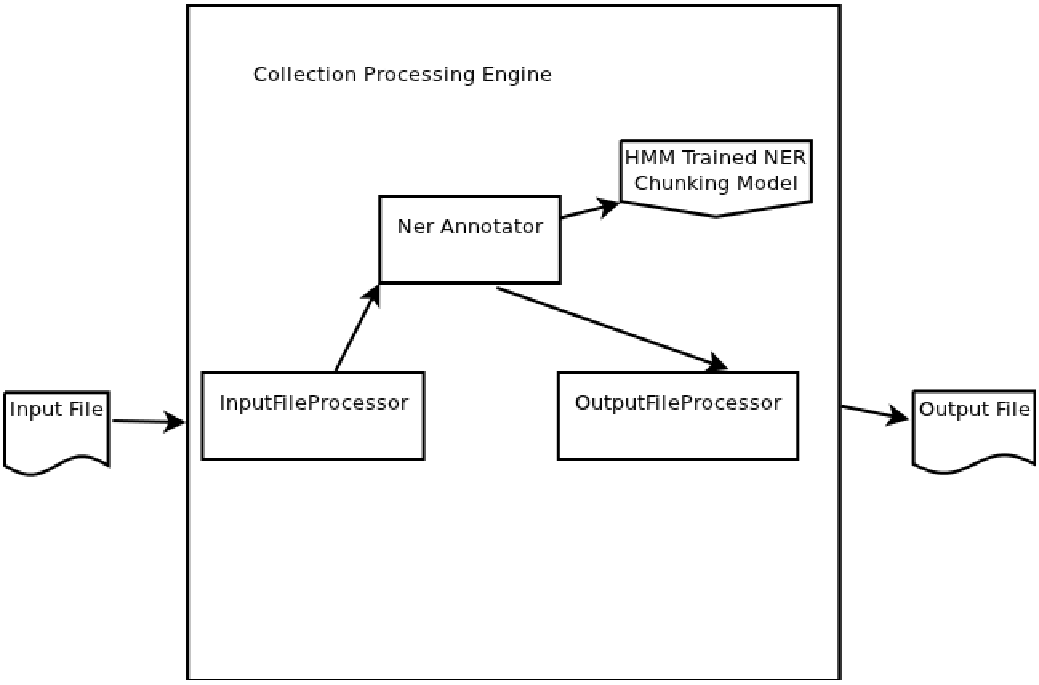
1 Introduction

This HomeWork report basically tries to describe a system that was built for the task of Named Entity recognition of Genes. The core of the system is basically built by using a Statistical Model for entity extraction that uses a Hidden Markov Model based chunking approach to identify named entities

The rest of the document basically concerns the details about the structural aspects of the system, models used for recognition and a preliminary perfomance evaluation of the system.

2 System Design

The system was basically built by using a UIMA Pipeline that is well suited for tasks related to Unstructured Information Management. At the heart of the system is the Analysis Engine, that is responsible for Named Entity Recognition of Genes.



The individual components of the system are described below

2.1 Input File Processor

The input handling is basically done by using a Collection reader approach used in UIMA. The purpose of the collection reader is to initialize the CAS (a document representation structure) with the sentences along with their corresponding IDs. The Input File Processor reads the input converts into a format required by the Named Entitiy Annotation engine and populates the CAS with this information.

The Collection reader also is responsible for several input management tasks such as

- Abstraction of the Input Sources to the Analysis engines
- Detecting the encoding of the language

2.2 Analysis Engine

The Analysis Engine forms the core component of our Analysis. The abstraction of using an Analysis Engine with several annotators helps in pipeling the system in an effecient manner. The NERExtractor does the job of using the statistical model trained with a corpus of annotated gene data to make predictions on Named Entities.

The NER Extractor also does several other tasks such as extracting the offsets required by the output format. The NERExtractor also populates the CAS with the Annotations that it has extracted. UIMA allows the annotations that are extracted to be handled by a rich Annotation Object Model.

2.3 Output File Processor

The Output File Processor is built using the CAS Consumer model specified in UIMA. The CAS Consumer model basically provides a neat interface for handling the results of our Analysis process. The CAS Consumer could be configued to perform different actions like writing to disk, populating a databse etc.

For our scenario, the output format required for evaluation needed to be in a specific file format. The Output File Processor does the job of retrieving all the annotation indexes that were populated by the Analysis Engine, processes the raw output format and writes to a pre-configured file specified prior.

3 Some Techniques and Algorithms Used

.A Statistical Named Entity Recognizer trained on a gene annotated dataset was used for this task. The model was trained using HMM's and used an algorithm called First-Best Named Entity Extractor was used to extract entities. The algorithm basically, decides if an annotation is a named entity based on a Confidence score that has been generated by the model.The following techniques/NLP approaches were used in this task and they are briefly described below

- Hidden Markov Models:A Markov model is basically a system which is assumed to be generated by a Markov Process with hidden states.
- Chunkers:Chunking is also called shallow parsing and it's basically the identification of parts of speech and short phrases (like noun phrases). Chunking has been widely used in applications like Named Entity recognition where the Named Entities are mostly Noun Phrases.
- Tokenization: Tokenization is the process of splitting a sentence into character or word tokens
- HMM Chunking: A HMM chunker uses Hidden Markov Models to perform chunking over tokenized sequences.

  The dataset on which the model was trained on was the GENETAG (2) corpus.

4 Preliminary Evaluation and Conclusion

The system was evaluated on the sample dataset provied for the task. The evaluation scores in terms of Precision,recall and F-Score are given below

- Precision : 0.748855774348
- Recall : 0.770380509171
- F-Measure: 0.759465659155

The problem was that there was too many noun phrases that were large and did not actually correspond to a gene. Better results could be obtained when better chunking models trained with a wider training set are used.

5 References

- LingPipe - (http://alias-i.com/lingpipe/)
- GENETAG - (http://www.biomedcentral.com/1471-2105/6/S1/S3)