11791-HW2

rgoutam

October 2014

1 Architecture

This system extracts named entities from biomedical text. It consists of three main components - collection reader, analysis engine and cas consumer.

The function of the collection reader is to read one sentence at a time from the text document, initialize the CAS with the sentence to be analyzed and send the CAS to the analysis engine.

The analysis engine contains the meat of the IIS. In this case, the analysis engine has a two components. One of them uses the lingpipe toolkit for named entity extraction using a statistical model trained the GeneTag dataset. The other analysis engine uses the ABNER toolkit for biomedical named entity extraction using a model trained on the NLPBA dataset. The specification for aggregating the two analysis engines are present in the aggregate analysis engine descriptor file. The analysis engine receives the CAS from the collection reader, processes it to tag named entities, adds the named entity data to cas and sends it to the CAS consumer.

The CAS consumer receives the CAS with analysis data from analysis engine, extracts relevant data from CAS and writes to file.

The analysis engine contains the Named Entity Analysis component. One of them uses lingpipe. Lingpipe has a statistically trained model that has been trained on the GENETAG corpus. GENETAG corpus is a tagged corpus for gene/protein named entities. The model is basically a hidden markov model (HMM) that sequentially tags words into named entity categories. Each named entity can span across multiple contiguous. A word in the beginning of the named entity is tagged as B-NER and word in the middle of the named entity is tagged as I-NER where NER stands for the symbol of the specific named entity. I have used the get n-best chunk option in lingpipe to get 10 best chunks per sentence. Lingpipe also gives me the confidence value (between 0 and 1) of each chunk. There are certain heuristics that I have used to filter out NEs that, according to my study of the data, are efficient in filtering out false positives.These heuristics are :

1. The confidence score should be greater than 0.5
2. The string length should be greater than 2.

3. The NE should not contain special characters like parentheses.

The analysis engine also contains ABNER, which is a biomedical named entity recognizer. ABNER has a model trained for named entity extraction on the NLPBA corpora with a performance F1 score of 0.705. I have added some simple rules on top of the named entities extracted by ABNER based on my study of the data. Such rules filter out NEs that, according to my heuristics, are not really NEs. These heuristics are

1. The length of the NE should be greater than 8
2. The number of words in the NE should be less than 3
3. The NE should not contain special characters like parentheses.

1.1 Aggregation of Analysis engines

Aggregation of the two analysis engines is really simple. I simply take the union of the NEs given by the two analysis engines. Other options for aggregation included the intersection of the NEs given by the two analysis engines, voting by weight to give an overall confidence score to NEs, etc. I did not choose intersection because that reduced my recall too much and hence, my F1 score was lower than when I took union. Voting by weight was not viable because ABNER does not really give a confidence value to the extracted NEs.

1.2 Type System

A type system has already been provided to us. I have created NEAnnotation type that inherits from Annotation type provided to us. It has another feature, called NamedEntity, to store the named entity as a string. I also store the confidence of the NE chunk given by the analysis engine and the CAS processorid in the same type.