11-791
Design and Engineering of Intelligent Information System

Homework 1

Report

Hao Zhang
AndrewID: haoz1
haoz1@andrew.cmu.edu

Architecture Design

1. Type system

For this task I designed the following types:

model.type.base: Inherit from uima.tcas.Annotation, has feature source (String v alue) and confidence (Float value)

mode.type.SentenceID: Inherit from model.type.base, has feature ID (String value)

mode.type.NERannotatior: Inherit from model.type.base, has feature NameEntity (String valu e) and SourceSentenceID (String value).

The most important type is *mode.type.NERannotatior*, which is used to annotate the start and end index of recognized name entity.

2. Input interface, collection reader design

Input data is stored in one single file and each line consists one instance. I open the file inp ut stream as the initialization in initialize method.

In the getNext method, I simply read a line from input file and use setdocText method from CAS to generate the CAS to processed.

The hasNext method checks if the File input stream is ready or come to an end.

3. Analysis Engine, name entity recognition design.
In the initialize method, I load the model file to be used for NER and create an instance of "Chunker" class, where the core algorithm lies in.

In the process method, I extract sentence from CAS text, separate sentence ID and put the sentence content to "chunk" method which is called by the "Chunker" class which is initialized in the beginning. The "chunk" method will return an instance of "Chunking" class, where contains the name entity recognition information. Finally I set the start and end index with its corresponding name entity to NERannotator object.

4. Output interface, collection consumer design.
In the initialize method, I open the output file stream and prepare to write.

The processCas method simply write the name entity recognition result in the output format . After process each CAS, the output stream will call flush method.

The collectionProcessComplete method will close output file stream at the end.

Machine learning/Natural language processing Algorithm design:

In this task, I import Lingpipe package for bio-name entity recognition. I incorporate the machine learning based algorithm in UIMA framework to recognize name entities. There are three generic, trainable chunkers which could be used for name entity recognition.
I used the CharLmHmmChunker for our task. The algorithm treats name entity recognition as a sequence labelling problem. Given the tokens (the HMM observations), it tries to find the most possible state sequence, the state here could be a binary variable representing if a token is name entity. The decoding problem could be solved with Viterbi algorithm.

I download a trained model: "English Genes: Gene Tag" from the Lingpipe website:
http://alias-i.com/lingpipe/web/models.html
This model is trained on bio-information text, which is suitable for our task.

Result

Using the sample.in as input file. The output file contains 20174 name entity entries. The gold truth file sample.out contains 18265 name entity entries.

Precision = 0.4083
Recall = 0.4510
F-score = 0.428