11-791: HW1 Report

Pengtao Xie (pxie1)
pengtaox@cs.cmu.edu

September 23, 2014

1 Overview

In this homework, we use UIMA SDK to deploy an gene named entity recognizer. The analysis engine consists of three components: collection reader, annotator, CAS consumer. The type system has two types: Sentence and EntityMention. The collection reader reads each line from the input file and parses the line into a Sentence object. The annotator takes each Sentence object and recognizes the entity mentions in this object. The CAS consumer saves each entity mention to the output file. Figure 1 show the flowchart of the analysis engine.

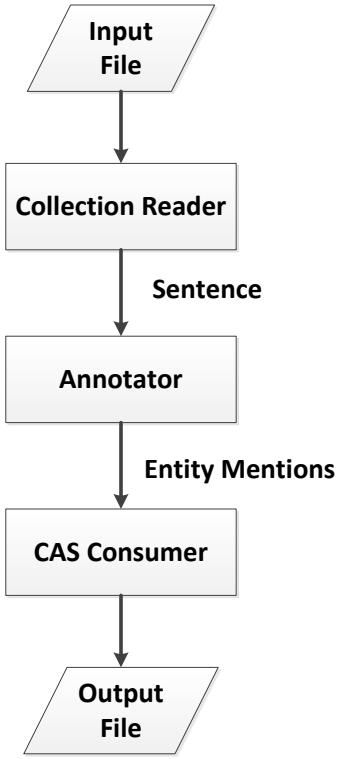

Figure 1: Flowchart of the NER analysis engine

2 Type System

We define two types in the type system. Figure 2 shows the UML diagram of the two types.

- Sentence: represent a sentence (each line in the input file). It has two members: 1, id: identifier of the sentence; 2, content: content of the sentence.
- EntityMention: represent a mention of the gene/protein entity. It has four members: 1, id: identifier of the sentence which contains the entity mention; 2, start: start position of this entity mention; 3, end: end position of this entity mention; 4, content: content of this entity mention.

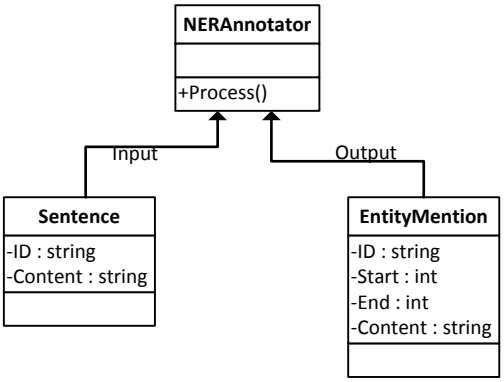

Figure 2: UML of the type system

3 Annotator

We use one annotator in this task: NERAnnotator. It takes a Sentence object as input and recognizes and outputs all entity mentions in this sentence.

4 Algorithm

We use the Stanford NLP Tools[1] to do this task. The NLP techniques used include:

- **Tokenization**. Tokenization is the process of breaking a stream of text up into words, phrases, symbols, or other meaningful elements called tokens. The list of tokens becomes input for further processing such as parsing or text mining. Tokenization is useful both in linguistics (where it is a form of text segmentation), and in computer science, where it forms part of lexical analysis.
- **Part-of-speech tagging**. In corpus linguistics, part-of-speech tagging (POS tagging or POST), also called grammatical tagging or word-category disambiguation, is the process of marking up a word in a text (corpus) as corresponding to a particular part of speech,

[1] http://nlp.stanford.edu/software/index.shtml

based on both its definition, as well as its contexti.e. relationship with adjacent and related words in a phrase, sentence, or paragraph. A simplified form of this is commonly taught to school-age children, in the identification of words as nouns, verbs, adjectives, adverbs, etc. Once performed by hand, POS tagging is now done in the context of computational linguistics, using algorithms which associate discrete terms, as well as hidden parts of speech, in accordance with a set of descriptive tags. POS-tagging algorithms fall into two distinctive groups: rule-based and stochastic. E. Brill's tagger, one of the first and most widely used English POS-taggers, employs rule-based algorithms.

- **Named entity recognition**. Named-entity recognition (NER) (also known as entity identification, entity chunking and entity extraction) is a subtask of information extraction that seeks to locate and classify elements in text into pre-defined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc.

The machine learning techniques used include:

- **Conditional Random Field**. Conditional random fields (CRFs) are a class of statistical modelling method often applied in pattern recognition and machine learning, where they are used for structured prediction. Whereas an ordinary classifier predicts a label for a single sample without regard to "neighboring" samples, a CRF can take context into account; e.g., the linear chain CRF popular in natural language processing predicts sequences of labels for sequences of input samples. CRFs are a type of discriminative undirected probabilistic graphical model. It is used to encode known relationships between observations and construct consistent interpretations. It is often used for labeling or parsing of sequential data, such as natural language text or biological sequences and in computer vision. Specifically, CRFs find applications in shallow parsing, named entity recognition and gene finding, among other tasks, being an alternative to the related hidden Markov models. In computer vision, CRFs are often used for object recognition and image segmentation.