# Project Individual 2: UIMA Type System

Leah Nicolich-Henkin: lnicolic

September 14, 2015

## Overview

This report outlines the UIMA type system for a task which evaluates whether answers to a question are correct. The system involves a series of annotators, which read the input file and decompose it on different levels. After annotations are complete, it ranks the answers according to how well they match the question, chooses the top $N$, where $N$ is the number that are actually correct, and then evaluates the output.

## Type System
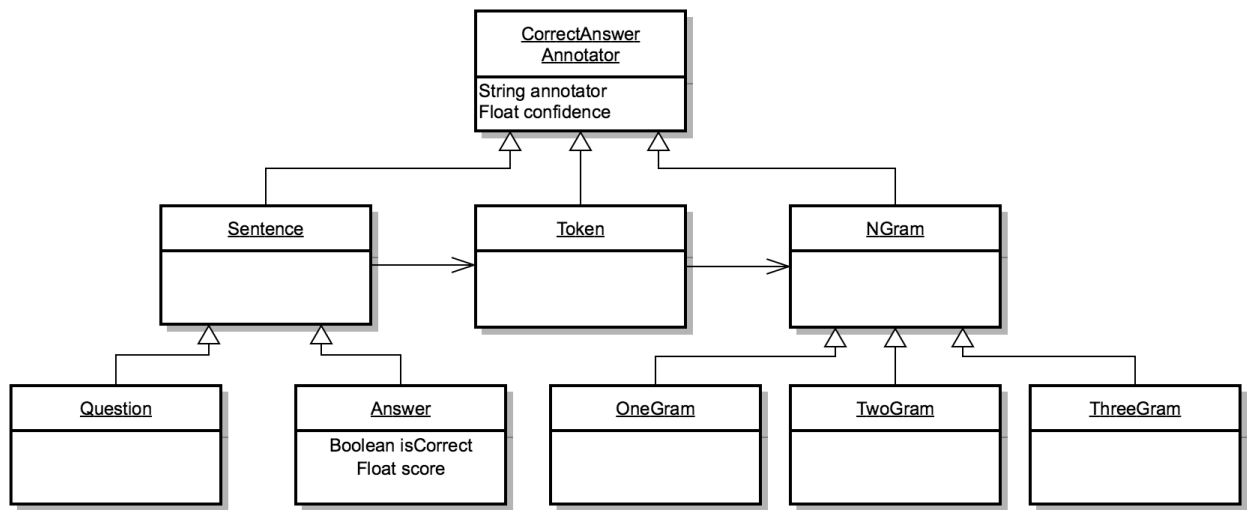


All the Annotations in this type system extend the PI2Annotation. The PI2Annotation has two features – a `confidence` Float, which provides the confidence that an annotation is accurate, and an `annotator` String that records which component created the Annotation –that are used by all Annotations in this Analysis Enginge. Having all the Annotations extend PI2Annotation ensures that they will all reliably follow this pattern, and eliminates the repetion that would otherwise be necessary if each Annotation had to implement those features independently.

There are three main types of PI2Annotations: the Sentence, the Token, and the NGram. There are two types of Sentences: Question, and Answer. When the input is first read by the system, the Test Element Annotator creates one Question and multiple Answer Annotations. The next annotator is the Token Annotator, which annotates Sentences with Tokens. These are in turn are used by the N-Gram Annotator to annotate the

N-Grams in the Sentence. The NGram Annotation is extended by three different Annotations: OneGram, TwoGram, and ThreeGram. Both Questions and Answers are annotated with tokens and n-grams, but Answers additionally have a Boolean feature `correct`, determined based on the input file, and a Float feature `score`, which is calculated with reference to the NGram Annotations of both the Answer itself and the associated Question.

Once the input has been fully annotated and the scores calculated, the Evaluation component can compare the scores of the different Answers, rank them, choose the top $N$ as correct, and then compare them to their `correct` values.

## Methodology

The organizing principle to this type system is that Question and Answer, by extending a Sentence annotation, are treated as essentially the same type, with the same annotations. This makes it easy to view them as analygous types and to compare them. Similarly, the Annotations used for comparison are all different NGrams, and can, by and large, be manipulated and compared in the same ways. The input flows through the different annotations, being broken into increasingly small and organized pieces, before those pieces are incorporated to evaluate the sentences.