

HW3-Report

TypeSystemExtensions

1)edu.cmu.deiis.subTypes.AnnotatedToken

AnnotatedTokenType extends the edu.cmu.deiis.types.Annotation by adding the:-
tokenText: stores the text which was annotated.
pos: (part of speech) which part of speech the words belong to.

2)edu.cmu.deiis.subTypes.AnnotatedNGram

The type extends the edu.cmu.deiis.types.NGram and adds :-
nGramToken:it is the nGramText

3)edu.cmu.deiis.subTypes.AnnotatedAnswer

It extends the answer score and adds the following :-
text: it is the sentence text
answerId:A unique ID associated with the answer
sentiment:Whether the sentence is negative or positive.

4)edu.cmu.deiis.subTypes.AnnotatedQuestion

It extends the question type and adds the following feature:-
text: it is the sentence text
sentiment:Whether the sentence is negative or positive.

5)edu.cmu.deiis.subTypes.Document

It contains:-
question:An AnnotatedQuestion to store details of the question in the document.
Answers:An Array to hold the answers
threshold:A confidence level associated with a document. Below this answers will be marked as false else true.

6)edu.cmu.deiis.subTypes.TokenizedSentence

This type extends the UIMA Annotation and adds the following:
annotatedtokens:An array to hold tokens

7)edu.cmu.deiis.subTypes.TokenizedDocument

This type extends the UIMA Annotation and adds the following features:
tokenizedQuestion: This if of type AnnotatedQuestion to hold the question.
tokenizedAnswers:This is an array to hold answers

8)edu.cmu.deiis.subTypes.NGramMatrix

This type extends the UIMA Annotation and adds the following
matrix: An $X \times N$ array. Where X represents the the number of answers and N represents size of Ngram.
It stores all the Ngrams associated with an answer,

Annotators

1)TestElementAnnotator(I/p: Document O/p Question Answer Spans)

It takes in a document as CAS type.

It separates the question and answers

To each answer it assigns whether it is true or false.

It assigns sentiment to the question and each answer i.e. -1 for -ve & +1 for +ve.

It creates a AnnotatedQuestion and an AnnotatedAnswerArray which are associated with a edu.cmu.deiis.subTypes.Document.

2)TokenAnnotator(I/p Spans O/p Annotated Tokens)

It will receive the edu.cmu.deiis.subTypes.Document and from it extract the AnnotatedQuestion and AnnotatedAnswerArray

It will split them into tokens.

These tokens will be assigned additional metadata.

The generated tokens will be stored in edu.cmu.deiis.subTypes.TokenizedDocument.

3)NamedEntityAnnotator(I/p Document O/p Named Entity Annotations)

This annotator is a remote implementation.

The aggregate analysis engine calls it via customResourceSpecifier.

When the service is called the pipeline is suspended.

On finishing the remote computation the process flow continues.

4)NGramAnnotator(I/p token O/p NgramTokens)

This annotator will receive the edu.cmu.deiis.subTypes.TokenizedDocument.

It will take the tokens and combine it into NGrams tokens.

i.e. For a tokens of a given sentence it will generate the unigram, bigram,... ngram tokens associated with the sentence.

5)AnswerScorer(I/p Ngrams O/p scored Answers)

It will check the sentiment of question and answer.

If they are incompatible i.e one has a sentiment score of 1 and other -1 it will arbitrarily assign it a confidence and score of 0.

If they are of compatible sentiments, it will take all the nGram for a given question and answer and try to match them with question. It will average the number of NGrams matched.

In case the answer text is longer than the question it will multiply the answer scores by a factor of (Alength/Qlength). This is done so that in a case where answer has more tokens in a question, the mismatches which occur due to data being absent in question should not reduce the confidence.

If NamedEntities are matched confidence is increased else there is a penalty and score is deducted.

The above steps give the confidence of each answer.

6)Evaluator(I/p AnswerSet O/p SortedAnswerSet)

It will sort the answers based upon their confidence scores using a comparator.

It will display the answers in decreasing orders of their confidence.

Along with the answer text it will display the answer confidence, the score(1 or 0) assigned to the answer assigned by the system and the gold standard score.

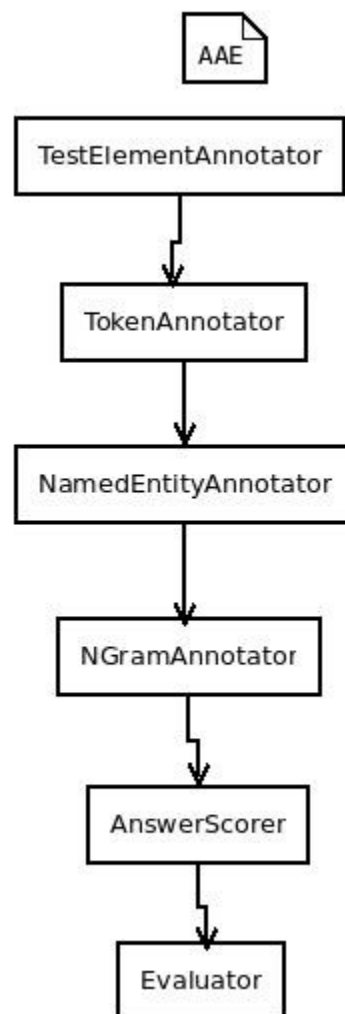
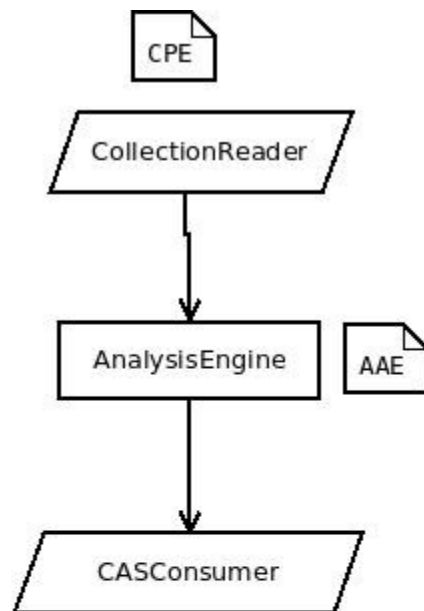
Flow Overview

The CPE consists of:-

Collection Reader

AggregateAnalysisEngines

CASConsumers



Structure of CPE

1. CollectionReader(i/p source)
2. AAE process the CAS
3. CASConsumer displays or can be used to persist the results

Components of AAE

Deployment Overview of UIMA AS

Since the UIMA AS works in a distributed environment a deployment diagram is shown below to illustrate a run time view of the system in a distributed context. Below the nodes ideally represent a machine but in reality it can be separate process space as well. Distribution enables scaling out is helpful when heavy specialized annotators are to be run. It is used especially when client does not have sufficient resources or data to run.

